Glolloc: Mixture of Global and Local Experts for Molecular Activity Prediction

Anonymous authors
Paper under double-blind review

Abstract

Quantitative structure-activity relationships (QSAR) models have been used for decades to predict the activity of small molecules, using encodings of the molecular structure, for which simple 2D descriptors of the molecular graph are still most commonly used. One of the recurrent problems of QSAR models is that relationships observed for a specific scaffold (pruned molecular skeleton) are sometimes not translatable to another, due to the 3D flexibility of molecular objects. This is also true when building multitask networks predicting the activity against several proteins at the same time - sometimes single protein models work better, and adding dissimilar proteins into the model decreases performance. In this paper, mixtures of experts (MoE) are used to combine a global network and local structures of the dataset (e.g. molecular scaffold, single protein) in the single task or multitask framework. We show that structuring the learning process with protein or chemical series information can enhance model performance and provide a built-in model introspection tool.

1 Introduction

Mixtures of experts (MoE) are a subset of ensemble learning algorithms, consisting in individual learners or experts, and a gate that assigns weights to different learners. They were relatively popular in the 90s (1; 2), and have recently regained interest in the deep learning community, notably for language and vision (3; 4). MoE can be seen as a type of meta-learning approach as it "learns to learn" from an ensemble of experts, which can be explicitly pre-defined e.g. by using clustering or subsets of input features (5; 6).

In the field of quantitative structure-activity relationships (QSAR), where the aim is to build predictive models from chemical structures (e.g. activity of molecule against a protein of interest), local models trained on subsets of molecules often show better performances than global models that are trained with the whole dataset (7).

In a drug discovery project, focused models are often constructed for several chemical series (sets of molecules with a similar skeleton or "scaffold"). However, throwing away most of the data comes with a loss of potentially useful information. Methods that are able to combine advantages from local and global models should therefore be of particular interest in the field.

Another potential application for problem partitioning would be multitask learning for ligand-protein interactions or proteochemometric modelling - i.e. predicting the activity of a small molecule against several proteins with the same model. These multi-protein models can perform better or worse than single task learning, depending on the protein (8) - subdividing the problem and training one expert per protein or protein family, in addition to a global expert trained on everything, could help to better adapt predictions to specific proteins.

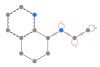
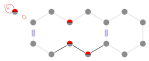
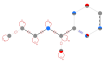
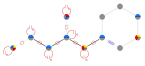
The MoE literature for quantitative structure-activity relationships (QSAR) is relatively scarce - MoEs were recently investigated (9) as a QSAR ensembling method across different feature spaces. In methods introduced in the following sections, the problem is naively partitioned based on input instances rather than input features, although combining both approaches would be an interesting follow-up. Attempting to pre-partition the input space

based on chemistry expert knowledge and then using a MoE architecture to choose between local and global experts has not yet been attempted. The main aim of the following experiments is to explore different automated ways to partition the problem in a way that is chemically relevant, and come up with a model capable to focus on a chemical series or protein of interest whilst not discarding valuable information.

2 Methods

2.1 Data and benchmark goals

Table 1: ChEMBL datasets and SMARTS pattern defining chemical compounds in validation and test set. The images were generated using SMARTSplus (10)

abbreviation	protein that we want to bind	train:valid:test	proxy chemical series
single task modelling			
akt1	RAC-alpha serine/threonine-protein kinase	642:35:35	
cp3a4	Cytochrome P450 3A4	1316:35:35	
cxcr4	C-X-C chemokine receptor type 4	145:20:20	
gcr	Glucocorticoid receptor	1118:70:70	
kif11	Kinesin-like protein KIF11	531:21:20	
protease	HIV-1 protease	1753:281:281	
revtrans	HIV-1 reverse transcriptase	1266:104:104	
multi-task modelling			
cyp	CP3A4, CP2D6, CP1A2, CP2C9, CP2CJ	2915:568:35	

In the following experiments, the 7 datasets (Table 1) correspond to 7 diverse proteins from the diverse subset of the DUD-E data collection (11). Note that while the diverse subset from DUD-E was used to choose the proteins for the targets, the ligand data associated with those proteins is not suitable for a topological QSAR experiment. Consequently, binding data was retrieved for each protein from a publicly available benchmark dataset (12) curated by Lenselink et al. (13), which compiles activity of small molecules extracted from the ChEMBL database (14).

Instances are small molecules, and the endpoint is pChEMBL activity (-Log(molar IC50, XC50, EC50, AC50, Ki, Kd or Potency)) against each of the 7 proteins. For each of these datasets, a tree-based "autosmarts" algorithm (Algorithm 1) was used on molecules with pChEMBL value > 5.5 to define a top chemical series of interest - its corresponding SMARTS pattern visualisation is reported in Table 1.

Compounds matching that series were split into train, valid and test using time splits, and the series-specific train set was merged with the remaining compounds, resulting in a set that contains both diverse molecules and the pattern of interest. The time splits were defined using publication year, to mimic what happens in a drug discovery project, where the aim is to do better on more recent data. It should be noted that publication year is not an ideal proxy and does not reflect the actual experiment date. It is, however, the only publicly available proxy.

The cp3a4 dataset was supplemented by other cytochrome data for the multi-task challenge (CP2D6, CP1A2, CP2C9, and CP2CJ) to construct the cyp dataset - the test dataset being the same as for cp3a4.

ECFP4_2048 descriptors were used to featurise molecules: Morgan fingerprints with a radius of 2 and size 2048, as implemented in rdkit (15).

Glolloc architectures (cf. next sections) were benchmarked against a single global expert baseline and the scikit-learn (16) implementation of Random Forests (17) regression. Parameters were optimised on the valid set without cross-validation - hidden layer sizes were the only variable hyperparameter for Glolloc networks (two hidden layers with [64, 64] or [1024, 1024] units). For Random Forests, combinations of hyperparameters were systematically screened from maximum depth = {5, 25, 50, 75, 100} and number of estimators = {100, 200}.

2.2 Proposed single task and multi-task networks

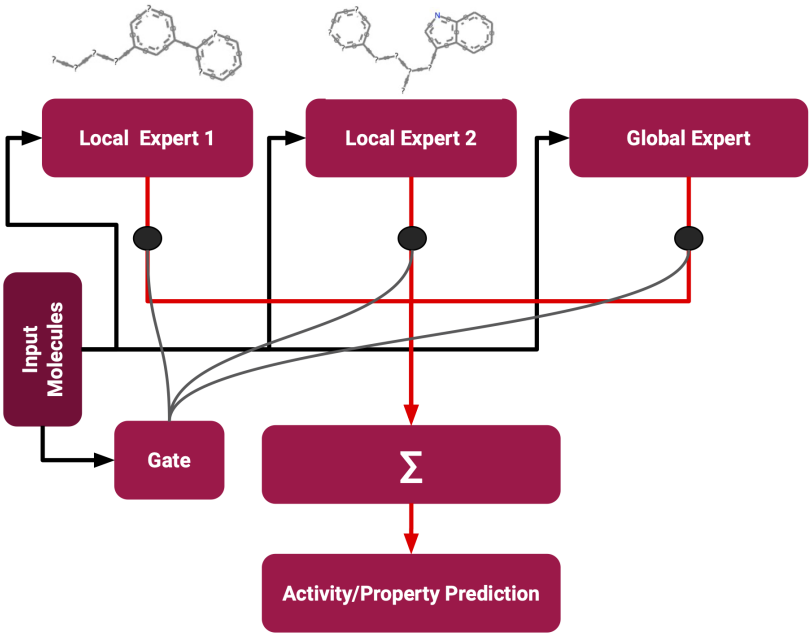


Figure 1: Mixture of Experts (MoE) for molecular activity modelling: the gate assigns weights to predictions from a global expert and individual experts specialised in specific chemical series

Two architectures were investigated: a single-task network with local experts specialising in different regions of the chemical space (Figure 1), and a multi-task network to predict the activity of multiple proteins, where local experts specialise in predicting specific proteins (Figure 2). In each case, in addition to the local networks (chemical series or protein experts), a global expert is trained on the whole data set, and is expected to perform better for molecules outside of the applicability domain of the local models.

Only fully connected layers were used, with fixed parameters for dropout (0.25 for experts, 0.025 for the gate), early stopping patience (15), batch size (32), learning rate (0.0005), and temperature (1) for the gate softmax - for the datasets in this paper, which are relatively small, the networks are easily trainable on a single CPU within a few minutes.

The gate is a fully connected network with a softmax function, outputing a gate weight matrix of dimension [batch size, number of tasks, number of experts], the softmax function being applied on the last dimension, so that weights across experts sum to 1 for each task.

The final predicted activity for a molecule n and task t is the weighted sum of predictions from different experts and gate weights (Equation 1).

$$\hat{y}_{nt} = \sum_{i \in Experts} G_{nti} \hat{y}_{nti} \tag{1}$$

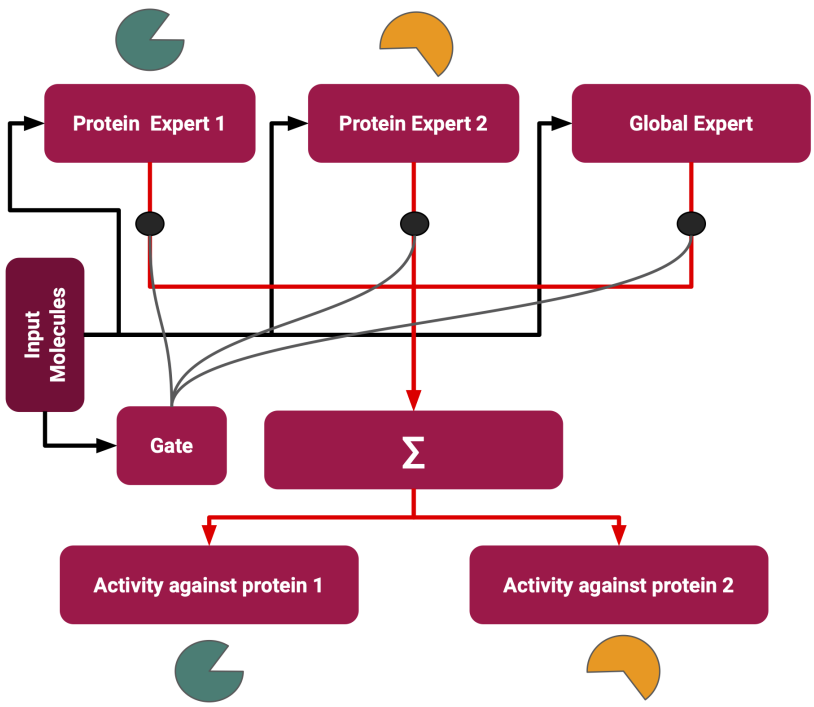


Figure 2: Mixture of Experts (MoE) for multitask ligand-protein modelling: the gate assigns weights to predictions from a global expert and individual protein-specific experts

2.3 Training procedure

Experts are defined using masks, so that the losses for irrelevant data points are not taken into account during the training process: e.g. if an expert focuses on pyrazole-containing molecular compounds, all losses for molecules not containing a pyrazole substructure will be zeroed out.

Experts and gate are trained at the same time, although each expert is kept independent by using expert-specific optimisers and losses (Equation 2). The gate is also assigned its own optimiser, but is made "aware" of expert predictions in the gate-specific loss (Equation 3), which is used to update the gate parameters.

This expert independence makes training the network challenging, as validation sets need to be carefully defined and loss curves monitored for each expert - local experts are in effect trained on subsets of the training data, and they are prone to overfitting. This is only partially addressed by adding learning rate decay for each optimiser.

$$expert_loss_i = \frac{1}{NT} \sum_{n \in expert_i}^{N} \sum_{t}^{T} y_{nt} - \hat{y}_{nti} \tag{2}$$

$$gate_loss = \frac{1}{NT} \sum_{n}^{N} \sum_{t}^{T} y_{nt} - \hat{y}_{nt} \tag{3}$$

2.4 Automated methods to define chemical series

In a drug discovery project, chemical series are typically identified by medicinal chemists as structural patterns. Chemoinformaticians convert the patterns into the appropriate SMARTS depictions - SMARTS being a type of regex language for chemical structures which allow to query molecules containing specific substructures.

However, for benchmarks on public datasets, an automated way to partition the chemical space is needed. Note that in a real-world scenario, project chemists would help in the definition of the series to add their domain knowledge of the model structure. Glolloc implements 3 methods to generate chemical series experts from a chemical dataset: cluster-based, tree-based, and neighbourhood-based.

The cluster-based method is self-explanatory: a k-means clusterer using scikit-learn's implementation (16) is trained on the training set and cluster labels are used to construct cluster-specific experts.

The tree-based and neighbourhood-based methods both rely on finding fuzzy maximum substructures (MCS) in groups of molecules, defined by their hierarchical clustering or neighbourhood membership, respectively, and using the RDKit (15) implementation of fuzzy MCS - with a minimum of 20 molecules per MCS. The advantage of MCS-based methods is that we obtain SMARTS patterns that can be visualised and interpreted easily - this type of approach is coined "autosmarts" in this paper.

The tree-based method (Algorithm 1) is similar to the one described recently by Kruger al (18): building fuzzy maximum common substructures (MCS) from nodes in a tree obtained using an agglomerative clustering algorithm, going from the root to the leaves: when a series (MCS) is found in a parent node, children are discarded. The difference between the Glolloc implementation and the one by Kruger et al. is the criterion to validate the MCS: they use a probability that a random molecule matches the scaffold, whereas Glolloc sets a minimum number of atoms for the scaffold, to guarantee a large enough scaffold (default value: 11 atoms). Strict MCS ring matching rules and loose rules for atom and bond matching were arbitrarily used - the impact of different MCS algorithm settings is not explored in this paper.

Algorithm 1 Tree-based autosmarts algorithm: find fuzzy maximum common substructures (MCS) in different regions of the chemical space using hierarchical clustering

```
min_atoms ← user parameter                    ▷ minimum number of atoms in MCS
min_node_size ← user parameter                ▷ minimum number of molecules to construct an MCS from
T ← dendrogram(D)                             ▷ construct dendrogram from the molecule data matrix D
N ← node_dict(T)                              ▷ dictionary mapping tree nodes to molecules
N′ ← sort(N)                                  ▷ sort nodes from root to leaves
MCS ← ∅                                       ▷ the MCS set stores the maximum common substructures
for node, molecules_in_node in N′ do
    if count(molecules_in_node) ≥ min_node_size then
        mcs ← fuzzy_maximum_substructure(molecules_in_node)
        ▷ if a large enough MCS is found in parent node
        if mcs exists and mcs.n_atoms ≥ min_atoms then
            MCS ← MCS ∪ {mcs}
            set children of node in N′ to ∅       ▷ discard children nodes
        end if
    end if
end for
```

The neighbourhood-based method implemented in Glolloc (Algorithm 2) is described in the Appendix, as it was not used for benchmarks in this paper.

3 Results

3.1 Single task: partition problem by chemical series

This section compares Random Forests and Glolloc architectures using different partitions of the chemical space to define experts.

Table 2: (a) Mean squared error (MSE) and (b) Spearman's rank correlation on single test set (using best model determined from validation set) for a single global expert, and different global + local expert setups: global expert with 6 random experts (global+6xrandom), 6 clusters (global+6clusters) or an automatically determined number of chemical series (global+autosmarts)

(a) MSE	global	global+6xrandom	global+6xclusters	global+autosmarts
akt1	1.048	1.036	**0.985**	1.109
cp3a4	0.142	**0.107**	0.117	0.118
cxcr4	0.148	**0.145**	0.153	0.148
gcr	1.442	1.472	1.348	**1.264**
kif11	0.533	0.603	0.534	**0.530**
protease	1.025	1.002	0.980	**0.894**
revtrans	0.833	0.854	**0.788**	0.885
(b) Spearman's ρ	global	global+6xrandom	global+6xclusters	global+autosmarts
akt1	0.634	0.624	**0.688**	0.603
cp3a4	0.624	0.657	**0.692**	0.630
cxcr4	0.208	0.150	**0.338**	0.190
gcr	0.320	-0.147	0.349	**0.361**
kif11	**0.703**	0.615	0.650	0.664
protease	0.440	0.443	0.457	**0.499**
revtrans	0.281	0.253	**0.353**	0.213

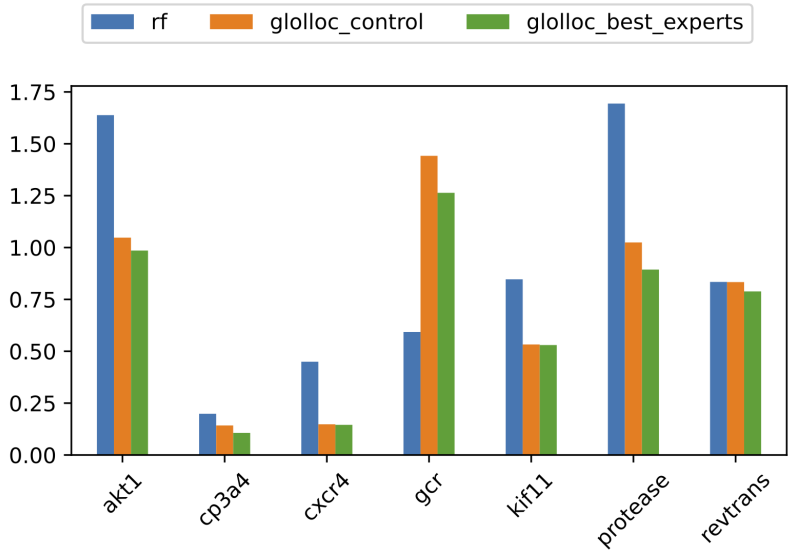


Figure 3: Test set mean squared error (MSE) for Random Forests (rf), a single global expert (glolloc_control) and best combination of experts (glolloc_best_experts).

Two control experiments were designed for the Glolloc framework: a model using a single global expert, and a model using one global expert and 6 random experts. The random experts divide the chemical space into 6 random sections.

The control experiments are run against two structured Glolloc models: one with the global expert and 6 cluster experts, the other with the global expert and 6 autosmarts (Algorithm 1) experts.

The more structured experts (based on clustering or autosmarts algorithms) yield lower mean squared errors (MSE) for 5 out of 7 datasets, and higher Spearman rank correlation in 6 out of 7 (Table 2).

When comparing the best glolloc expert combination (whether random or structured) against a single expert or Random Forests (Figure 3), Random Forests wins only for the gcr dataset and has the worst performance in all other cases.

3.2 Multi-task: partition problem by protein

Two control experiments were run on the cyp dataset in a multitask setting, with the goal of predicting CP3A4 activity in the test set. The first control consists in a single task model with only one global expert, the second in a multi-task model with also only one global expert. These controls are compared to a multi-task model with both a global expert and 5 protein experts, one for each of the CYPs in the dataset (CP2D6, CP1A2, CP2C9, CP2CJ, CP3A4). Results are reported in Figure 4.

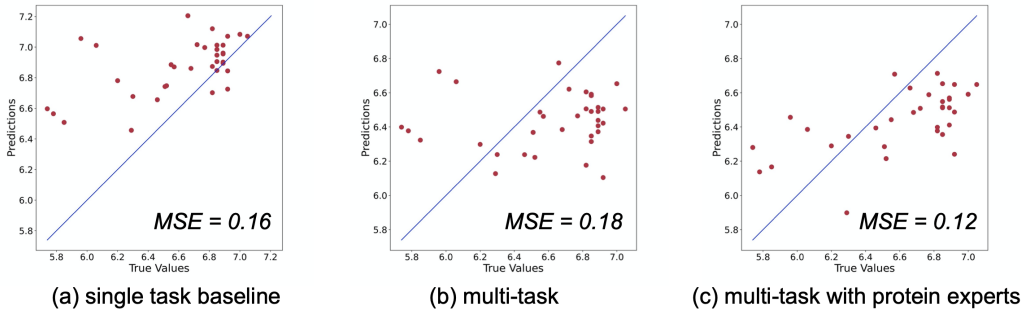
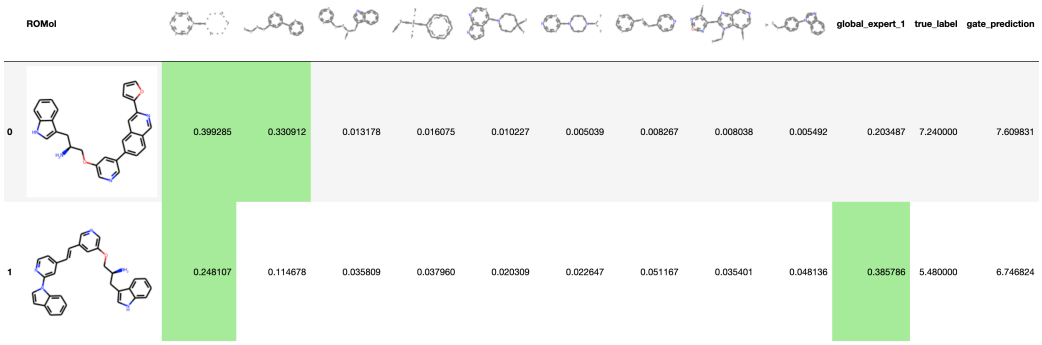


Figure 4: Test set mean squared error (MSE) on the cyp test set for (a) a single task global expert, (b) multi-task global expert and (c) multi-task with both global expert and 5 protein experts - one for each of the proteins in the cyp training set.

Predictions from the single task model are generally higher than from the multi-task model, and lower-activity molecules are not well predicted. The multi-task baseline pushes those predictions down but the range of predicted values stays narrow; adding protein experts improves the MSE in this particular instance.

3.3 Model introspection

The main interest of partitioning a QSAR problem is for downstream model introspection. In Figure5, the weights assigned to different local experts are shown for two molecules in the akt1 validation set. If the model was trained correctly, most of the gate weights should go to experts specialised in the relevant chemical series and the global expert. Molecules belonging to chemical series which are not covered by local experts are expected to be mostly assigned to the global expert.



	ROMol										global_expert_1	true_label	gate_prediction
0		0.399285	0.330912	0.013178	0.016075	0.010227	0.005039	0.008267	0.008038	0.005492	0.203487	7.240000	7.609831
1		0.248107	0.114678	0.035809	0.037960	0.020309	0.022647	0.051167	0.035401	0.048136	0.385786	5.480000	6.746824

Figure 5: Gate weights visualisation for two molecules in the akt1 validation set: columns with SMARTS patterns symbolise local experts specialised in chemical series. For each molecule, the two experts with the largest weight are highlighted.

The same type of visualization can be produced for the multi-task, multi-protein Glolloc architecture (Figure 2): in Figure 6, weights assigned to different protein experts are shown for two molecules in the cyp validation set. Only the gate weights for CP3A4 activity prediction are shown in the figure, as different weights would be assigned if another task was queried (e.g. CP2D6, CP1A2, CP2C9, or CP2CJ). For the CP3A4 task, most predictions are dominated by the CP3A4 expert and the global expert.

	ROMol	CP2D6	CP1A2	CP2C9	CP2CJ	CP3A4	global_expert_1	true_label	gate_prediction
0		0.000054	0.000008	0.000198	0.000055	0.450989	0.548695	7.660000	7.561827
1		0.000360	0.000085	0.001001	0.000370	0.466637	0.531546	6.850000	7.231433

Figure 6: Gate weights visualisation for two molecules in the multitask cytochrome dataset, querying predictions for CP3A4 activity. CP2D6, CP1A2, CP2C9, CP2CJ, and CP3A4 are protein-specific experts. For each molecule, the two experts with the highest gate weight are highlighted.

4 Conclusion and next steps

Glolloc adds local structure to QSAR models, which can be used for downstream model introspection by visualising how local or global structures contribute to the final prediction.

In the preliminary investigations shown in this paper, models adding more structure, based on chemical series, clusters, or single proteins, can often perform better than controls (RF/single experts/random experts). However, these models need more careful optimisation, and more

datasets and splits should be investigated in follow-up experiments to build more confidence in performance improvement.

There is still progress to be made on the side of expert training, as currently the masked architecture is not ideal and needs for the user to monitor a series of losses. Arguably, the discrete, masked multi-expert setting could be replaced by a more continuous approach. Alternatively, an attention mechanism could be used instead of experts, to give more attention to predefined dataset structures.

Different 2D and 3D depictions of molecules could also be used to define additional experts, as well as potential groupings of related tasks, such as protein families in the multi-task setting. Glolloc-like architectures could also be used to model different endpoint types or experimental settings.

5 Appendix

5.1 Other automated methods to define chemical series

Other methods of detecting chemical series in a data set can be used beyond tree-based, e.g. varying the number of molecules that can match an MCS in a data set, fragment-based methods, scaffold tree methods, or nearest neighbours algorithms. The neighbourhood-based autosmarts method for series identification also implemented in Glolloc is based on the algorithm described by Hattori et al. (19) and adapted for MCS series definition. Hattori et al.'s algorithm is based on iteratively finding representative molecules with the largest neighbourhood and removing them and their neighbours from the pool of molecules. This method is dependent on a similarity threshold parameter to define the neighbours. For Glolloc, neighbourhoods are also iteratively defined, but at each iteration a fuzzy MCS from the top neighbourhood is defined and all molecules matching that MCS are removed from the pool used at the next iteration. The neighbourhood-based approach is not used in this paper for benchmarks as it seemed to provide similar results to the tree-based method when varying the similarity threshold for our data sets.

Algorithm 2 Neighbourhood autosmarts algorithm: find fuzzy maximum common substructures (MCS) in different regions of the chemical space using nearest neighbours

```
min_similarity ← user parameter                          ▷ similarity cutoff
min_molecules ← user parameter                ▷ minimum number of molecules per MCS
bag_of_molecules ← set of all molecules indices
S ← similarity_matrix(D)                ▷ similarity matrix from molecule data matrix D
MCS ← ∅                          ▷ the MCS set stores the maximum common substructures
while bag_of_molecule is not empty do
    ▷ find neighbourhood of molecule with max number of neighbours
    top_neighbourhood ← f(S, min_similarity)
    if count(top_neighbourhood) ≥ min_molecules then
        mcs ← fuzzy_maximum_substructure(top_neighbourhood)
        if mcs exists then                          ▷ if MCS is found in neighbourhood
            MCS ← MCS ∪ {mcs}
            mcs_matches ← matches(bag_of_molecule, mcs)
            ▷ discard molecules matching MCS for next iteration
            bag_of_molecule ← bag_of_molecule \ mcs_matches
            S[mcs_matches] ← 0.                ▷ set similarities of discarded molecules to 0
        end if
    end if
end while
```

References

[1] M.I. Jordan and R.A. Jacobs. Hierarchical mixtures of experts and the EM algorithm. In *Proceedings of 1993 International Conference on Neural Networks (IJCNN-93-Nagoya,*

Japan). IEEE.

[2] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3:79–87, 1991.

[3] Carlos Riquelme Ruiz, Joan Puigcerver, Basil Mustafa, Maxim Neumann, Rodolphe Jenatton, André Susano Pinto, Daniel Keysers, and Neil Houlsby. Scaling vision with sparse mixture of experts. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

[4] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. 2017.

[5] S. Gutta, J.R.J. Huang, P. Jonathon, and H. Wechsler. Mixture of experts for classification of gender, ethnic origin, and pose of human faces. *IEEE Transactions on Neural Networks*, 11(4):948–960, July 2000.

[6] Bin Tang, M.I. Heywood, and M. Shepherd. Input partitioning to mixture of experts. In *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No.02CH37290)*. IEEE.

[7] Hua Yuan, Yongyan Wang, and Yiyu Cheng. Local and global quantitative structure-activity relationship modeling and prediction for the baseline toxicity. *Journal of Chemical Information and Modeling*, 47(1):159–169, December 2006.

[8] Héléna Alexandra Gaspar, Mohamed Ahmed, Thomas Edlich, Benedek Fabian, Zsolt Varszegi, Marwin Segler, Joshua Meyers, and Marco Fiscato. Proteochemometric models using multiple sequence alignments and a subword segmented masked language model. May 2021.

[9] Gyula Dörgő, Omar Péter Hamadi, Tamás Varga, and János Abonyi. Mixtures of QSAR models: Learning application domains of pka predictors. *Journal of Chemometrics*, 34(4), April 2020.

[10] Christiane Ehrt, Bennet Krause, Robert Schmidt, Emanuel S. R. Ehmki, and Matthias Rarey. SMARTS.plus – a toolbox for chemical pattern design. *Molecular Informatics*, 39(12):2000216, October 2020.

[11] Michael M. Mysinger, Michael Carchia, John. J. Irwin, and Brian K. Shoichet. Directory of useful decoys, enhanced (DUD-e): Better ligands and decoys for better benchmarking. *Journal of Medicinal Chemistry*, 55(14):6582–6594, July 2012.

[12] Eelke Bart Lenselink, Niels ten Dijke, Brandon Bongers, George Papadatos, Herman W. T. van Vlijmen, Wotjek Kowalczyk, Adriaan P. IJzerman, and Gerard J. P. Van westen. Beyond the hype: Deep neural networks outperform established methods using a chembl bioactivity benchmark set [version 1]. 4TU.ResearchData https://doi.org/10.4121/uuid:547e8014-d662-4852-9840-c1ef065d03ef, Jul 2017.

[13] Eelke B. Lenselink, Niels ten Dijke, Brandon Bongers, George Papadatos, Herman W. T. van Vlijmen, Wojtek Kowalczyk, Adriaan P. IJzerman, and Gerard J. P. van Westen. Beyond the hype: deep neural networks outperform established methods using a ChEMBL bioactivity benchmark set. *Journal of Cheminformatics*, 9(1), August 2017.

[14] Anne Hersey. ChEMBL database release 13. Technical report, September 2013.

[15] RDKit: Open-source cheminformatics. `http://www.rdkit.org`. [v. 2021.3.3].

[16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[17] Leo Breiman. *Machine Learning*, 45(1):5–32, 2001.

[18] Franziska Kruger, Nikolas Fechner, and Nikolaus Stiefl. Automated identification of chemical series: Classifying like a medicinal chemist. *Journal of Chemical Information and Modeling*, 60(6):2888–2902, May 2020.

[19] Kazunari Hattori, Hiroaki Wakabayashi, and Kenta Tamaki. Predicting key example compounds in competitors' patent applications using structural information alone. *Journal of Chemical Information and Modeling*, 48(1):135–142, January 2008.