Molecular Interpretation Workflow through Attentive Recursive Tree Model

Nural Ozel[1*†], Berk Atıl[2†], Asu Busra Temizer[3,4], Taha Koulani[3,4], Elif Ozkirimli[5], Nilgün Karalı[3], Kutlu O. Ulgen[1,6], Arzucan Özgür[2]

[1*]Department of Computational Science and Engineering, Faculty of Arts and Sciences, Boğaziçi University, Istanbul, 34342, Turkey.
[2]Department of Computer Engineering, Faculty of Arts and Sciences, Boğaziçi University, Istanbul, 34342, Turkey.
[3]Department of Pharmaceutical Chemistry, Faculty of Pharmacy, Istanbul University, Istanbul, 34452, Turkey.
[4]Department of Pharmaceutical Chemistry, Institute of Health Sciences, Istanbul University, Istanbul, 34126, Turkey.
[5]Data and Analytics Chapter, Pharma International Informatics, F. Hoffmann-La Roche AG, Basel, 4070, Switzerland.
[6]Department of Chemical Engineering, Faculty of Arts and Sciences, Boğaziçi University, Istanbul, 34342, Turkey.

*Corresponding author(s). E-mail(s): nural.ozel@boun.edu.tr;
Contributing authors: atilberk98@gmail.com; temizerab@gmail.com; tahakoulani8@gmail.com; elif.ozkirimli@roche.com; karalin@istanbul.edu.tr; arzucan.ozgur@boun.edu.tr; ulgenk@boun.edu.tr;
†These authors contributed equally to this work.

Abstract

Purpose: To identify and interpret the significant drug molecule fragments for the corresponding molecular property tasks (e.g., physiology, biophysics, and physical chemistry), we propose a sequence of processes named Molecular Interpretation Workflow through Attentive Recursive Tree Model (MIW-ART) by utilizing the Attention Recursive Tree (AR-Tree) model.
Methods: Using tree representations of molecules provided by AR-Tree, we identified and scored chemically valid structures, referred to as "fragments". We performed experiments on a diverse set of tasks from the MoleculeNet, comprising

five classification and four regression tasks, as benchmark tasks. The identified fragments were then interpreted from a medicinal chemistry perspective.
Results: The model outperformed the state-of-the-art models in the clinical trial toxicity (ClinTox) and the β-secretase (BACE) regression tasks.
Conclusion: The proposed workflow succeeded in finding chemically meaningful fragments for the benchmark tasks, including blood-brain barrier penetration (BBBP), ClinTox, toxicology in the 21st century (Tox21), and BACE.

Keywords: interpretation, molecule, drug, deep learning, attention

1 Introduction

Identifying key properties of novel molecules, such as partition coefficient, solubility, and toxicity, is crucial in new drug discovery. Traditional methods are time-intensive and costly, involving complex laboratory experiments. To overcome these challenges, computational methods based on deep learning (DL) have emerged as valuable alternatives [1, 2], offering promising solutions [3]. The interpretation of these methods is essential, as it enhances our understanding and facilitates the efficient identification of significant molecular features.

In this regard, various methods for determining feature importances in neural networks have been proposed, primarily focusing on interpretability. Backpropagation-based methods like DeepLIFT [4] highlight the significance of inputs by considering deviations from a reference state. Similarly, forward propagation-based methods by Zeiler and Fergus [5] and Zintgraf et al. [6] alter and evaluate layer activations. Despite their benefits, methods like those by Klöppel et al. and Ecker et al. [7, 8] have been critiqued for potential misinterpretations [9, 10]. Layer-wise Relevance Propagation (LRP) methods [11], alongside deconvolutional network-based methods [12], and gradient-based localization methods [13, 14], further contribute to the field. The integration of latent tree-based [15–18] and attention-based methods [19–23] demonstrates the evolution of interpretative strategies in neural network analysis.

Building on this foundation, our work introduces the Molecular Interpretation Workflow through Attentive Recursive Tree Model (MIW-ART). A graphical abstract of the workflow can be seen in **Figure 1**. The workflow is based on Attentive Recursive Tree (AR-Tree) model [24], consists of Tree-structured Long Short-Term Memory (Tree-LSTM) technology. This model excels in learning molecular representations and focuses on task-specific sentence embeddings, highlighting essential molecular tokens. We identify chemically valid subtrees as "fragments" for a comprehensive scoring process. This scoring encompasses token scores, loss value, and repetition count in specific tasks, leading to a systematic categorization and interpretation of significant molecular substructures. The top 20 fragments for each task are selected for further analysis, aiming to enhance the interpretability and efficiency in molecular property exploration. Our workflow signifies a notable advancement in computational drug discovery, paving the way for quicker and more cost-effective identification of drug candidates, underlining the critical role of interpretation in this evolving field.

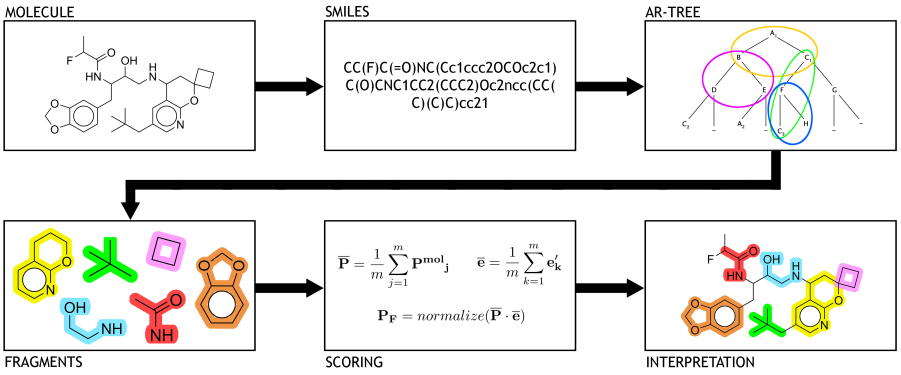

Fig. 1 Graphical abstract of the workflow.

2 Results

2.1 Model Performance on Benchmark Tasks

We tested our workflow MIW-ART on a variety of datasets such as physiology (BBBP, ClinTox, SIDER, and Tox21), biophysics (BACE) and physical chemistry (ESOL, Lipophilicity, FreeSolv) datasets. The receiver operating characteristic curve-area under the curve (ROC-AUC) scores of our model and the state-of-the-art (SOTA) models [25–27] are shown in **Table 1**. The root mean square error (RMSE) scores of our model and the SOTA models [25–27] are shown in **Table 2**. The first row is our model's results, the results for the both version of ChemBERTa retrieved from [26], the results in the last row are retrieved from [27] and the rest are retrieved from [25]. Also, "-" indicates there is no result found in the related paper. Note, the models that are in last five rows are pretrained models. For BACE Classification, BBBP, ClinTox, SIDER (Side Effect Resource), and Tox21 tasks, our model's scores are %76.2, %64.9, %99.7, %57.7, and %63.2, respectively. For BACE Regression, ESOL (Estimated Solubility), FreeSolv (Free Solvation), and Lipophilicity tasks, our model's scores are 1.02, 0.45, 0.77, and 0.71, respectively.

Our model outperformed the SOTA models in the ClinTox and the BACE Regression tasks and achieved moderate scores in the other benchmark tasks, according to the results. All of the tested parameters as well as the best hyperparameters can be seen in **Appendix E**.

2.2 Interpretations of Found Fragments

The workflow fragments obtained have been analyzed within the context of their respective tasks. The fragments were referred to as "meaningful fragments" if they were deemed significant by pharmaceutical chemistry and supported by the literature.

Table 1 Score table of classification tasks.

Model	%BACE Clf	%BBBP	%ClinTox	%SIDER	%Tox21
AR-Tree	76.2	64.9	99.7	57.7	63.2
RF	86.7	71.4	71.3	68.4	76.9
SVM	86.2	72.9	66.9	68.2	81.8
GCN	71.6	71.8	62.5	53.6	70.9
GIN	70.1	65.8	58.0	57.3	74.0
SchNet	76.6	84.8	71.5	53.9	77.2
MGCN	73.4	85.0	63.4	55.2	70.7
D-MPNN	85.3	71.2	90.5	63.2	68.9
[28]	85.9	70.8	78.9	65.2	78.7
N-Gram	87.6	91.2	85.5	63.2	76.9
$\text{MolCLR}_{\mathbf{GCN}}$	78.8	73.8	86.7	66.9	74.7
$\text{MolCLR}_{\mathbf{GIN}}$	89.0	73.6	93.2	68.0	79.8
ChemBERTa-1	-	64.3	73.3	-	72.8
ChemBERTa-2	79.9	74.2	60.1	-	83.4
MoLFormer-XL	88.2	93.7	94.8	69.0	84.7

ROC-AUC is used as the performance metric and the binary cross-entropy (BCE) is used as the loss function. Higher values indicate better performance.

Table 2 Score table of regression tasks.

Model	BACE Reg	ESOL	FreeSolv	Lipophilicity
AR-Tree	1.02	0.45	0.77	0.71
RF	1.32[1]	1.07	2.03	0.88
SVM	-	1.50	3.14	0.82
GCN	1.65[1]	1.43	2.87	0.85
GIN	-	1.45	2.76	0.85
SchNet	-	1.05	3.22	0.91
MGCN	-	1.27	3.35	1.11
D-MPNN	2.25[1]	0.98	2.18	0.65
[28]	-	1.22	2.83	0.74
N-Gram	-	1.10	2.51	0.88
$\text{MolCLR}_{\mathbf{GCN}}$	-	1.16	2.39	0.78
$\text{MolCLR}_{\mathbf{GIN}}$	-	1.11	2.20	0.65
ChemBERTa-1	-	-	-	-
ChemBERTa-2	1.36	0.86	-	0.74
MoLFormer-XL	-	0.28	0.23	0.53

RMSE is used as both the performance metric and the loss function. Lower values indicate better performance.

[1]Retrieved from ChemBERTa-2 [26]

To determine whether the workflow provided truly meaningful fragments, literature-based SAR, STR, or physicochemical properties of small molecules were considered during tasks' interpretation.

SAR is defined as the relationship between the compounds' three-dimensional (3D) structures and their biological activities. Minor changes to the structure of the selected lead compound are made and their effects on biological activity are evaluated based

on the assumption that structurally similar compounds have similar physical and biological properties. STR refers to the relationship that exists between the structures of the compounds or its functional groups and toxicity profiles. The physicochemical parameters such as partition coefficient, lipophilicity or polarity, acidity or basicity are of great importance in determining the drug properties and the criteria for absorption, distribution, metabolism and excretion (ADME).

SAR studies from the literature were used to analyze the BACE and BBBP tasks, while STR studies were used to examine the ClinTox and Tox21 tasks. The ESOL and lipophilicity tasks were investigated by taking the molecules' physicochemical properties into account. The following subsection presents the fragments, the names of the corresponding molecules, and the results of the analyses for the BACE Classification task. The rest of all the analysis are given in **Appendix C**. The workflow determined that the colored fragments were the most meaningful fragments for the corresponding tasks.

The significant fragments identified for each benchmark task can be seen in the images in **Appendix A**. These fragments in the images were sorted in descending order from left to right based on their fragment scores.

2.2.1 BACE Classification Analysis

BACE1 (β-site amyloid precursor protein cleaving enzyme 1, β-secretase 1) is a protease enzyme that plays an important role in the formation of β-amyloid peptides in the chain of events that cause Alzheimer's disease. BACE1 levels increase in brains and cerebrospinal fluids of Alzheimer's patients. Thus, BACE1 has become an important target for drug development studies aimed at Alzheimer's disease and new BACE1 inhibitors have been developed to this end. BACE1 inhibitors interact with catalytic aspartic acid dyad residues (Asp32 and Asp228) located at the active ligand binding sites of the enzyme to inhibit the proteolytic activity of BACE1. The binding of a ligand to Asp32 and Asp228 increases the overall binding affinity to the enzyme and therefore the strength of the bond. In addition, subpockets defined as S1, S2, S3, S4, S1', S2', S3', and S4' were detected in the ligand binding region of BACE1. These additional binding pockets, which are different from the catalytic pockets, were found to contribute to the inhibitory effect and stability of the enzyme-substrate complex. Hydrophobic moieties that bind to the enzyme's hydrophobic pockets, alongside polar and charged groups that can establish hydrogen bonds and electrostatic bonds with the enzyme, are all essential chemical groups in binding to BACE1 [29–31]. Effective inhibition of BACE1 requires an inhibitor with higher affinity for the enzyme's binding site than the endogenous substrate. This can be achieved by maximizing the number of binding interactions with BACE1, primarily binding to Asp32 and Asp228. The extended active subpocket region of BACE1 has wide amino acid tolerance, but most of the central ones harbor hydrophobic side chains. This property may be advantageous in the development of BACE1 inhibitors with enhanced lipophilicity to improve membrane permeability and penetration into the blood-brain barrier (BBB). An amide bond or its many transition state bioisosteres such as hydroxyethylene, hydroxyethylamine, isophthalamide have been used as the main element in the design of BACE inhibitors [31].

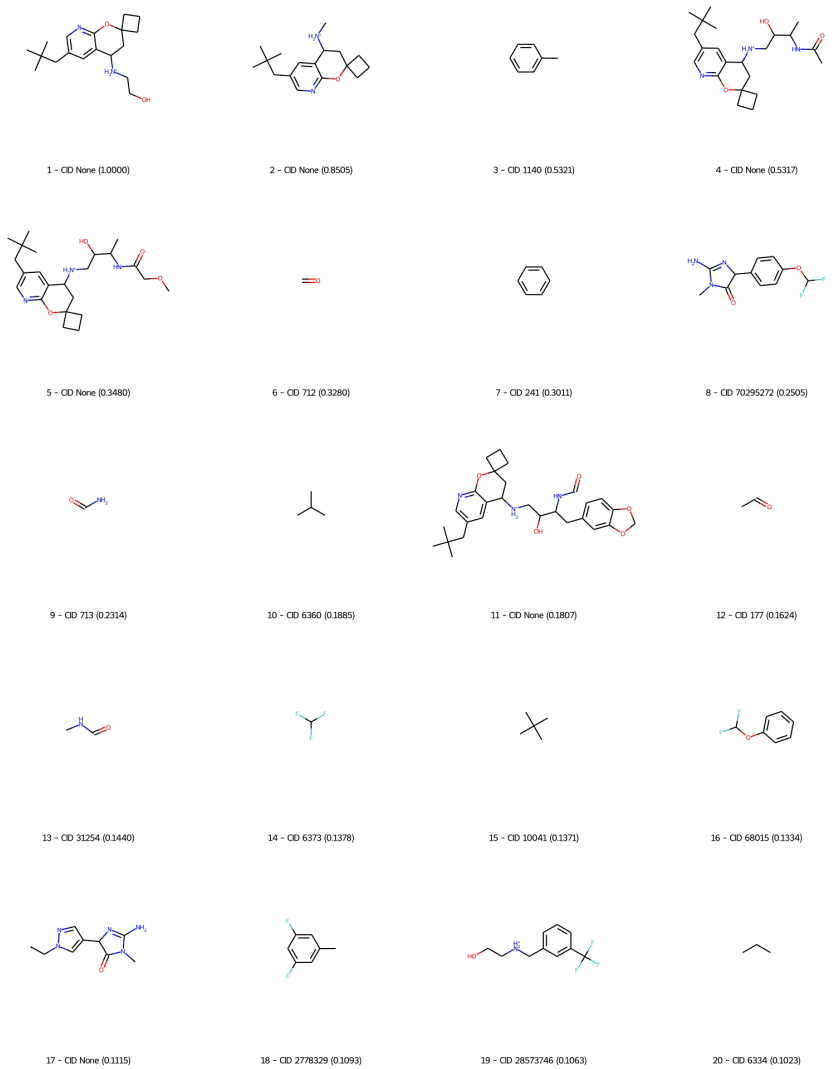

Fig. 2 Top 20 fragments for the BACE Classification task.

Within the top 20 fragments presented by the workflow **Figure 2**, fragments 1, 4, 5, and 11 bear the same structural core, 2-spirocyclobutyl-6-neopentyl-8-azachromane, which has been shown in studies in the literature to be a promising core for the development of innovative BACE1 inhibitors [32–34]. Some of the compounds obtained in these studies are shown in **Figure 3(a-c)**. One commonality among these studies is the presence of a hydroxyethylamine (HEA) scaffold (shown in blue **Figure 3**) across all of them that is believed to be an important moiety in terms of forming hydrogen bonds (H-bonds) with the aspartic catalytic dyad of BACE1, making it a

moiety that should be maintained to have optimal binding to the receptor. As depicted in **Figure 2**, fragments 1, 4, 5, and 11 bear the HEA scaffold; however, fragment 2 does not fully incorporate this scaffold, but still contains the methylamino part of the HEA moiety. It is also reported that the azachromane (shown in yellow) and the spirocyclobutane (shown in pink) rings that occupy the S1' subpocket along with the neopentyl group (shown in green) that fills the S2' subpocket, all increase the binding affinity to BACE1 and therefore are essential structural components of BACE1 inhibitors. Additionally, the benzodioxolane group (shown in orange) was introduced in a new series of molecules resulting in the enhancement of the oral bioavailability and metabolic stability of the previously designed compounds [32]. The workflow also performed successfully in the extraction of this fragment, which holds suitable features in terms of rational drug design. Furthermore, fragments 9 and 13 can be assigned to the terminal acetamide group (shown in red), representing the formamide (`NC=O`) and N-methylformamide (`CNC=O`) structures, respectively. This acetamide group was designed by shortening a larger amide group to optimize the molecular weight, the binding efficiency, and the central nervous system (CNS) permeability of previously-synthesized undesired compounds [34]. Fragment 15 is also excluded by the workflow as an important part of the neopentyl group (shown in green).

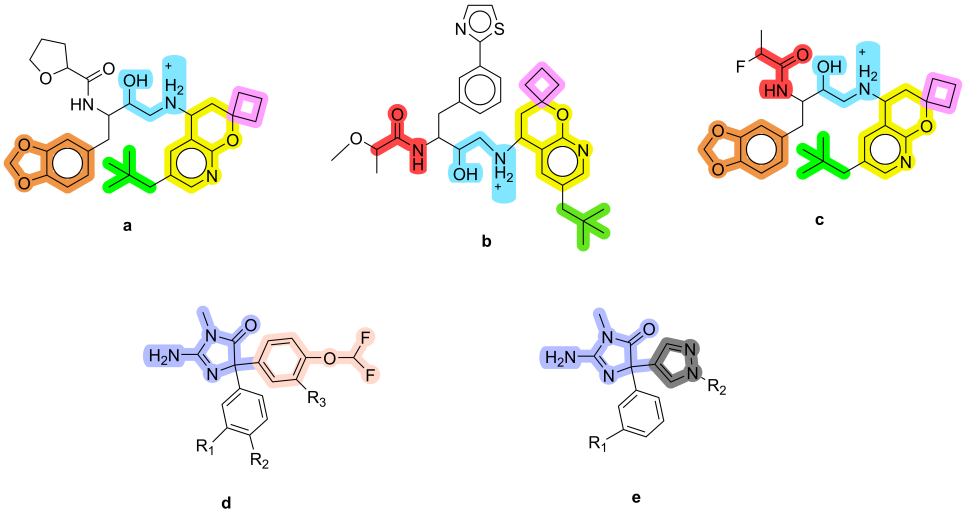

Fig. 3 2D structure of 6'-neopentyl-3',4'-dihydrospiro[cyclobutane-1,2'-pyrano[2,3-b]pyridin]-4'-aminium derivatives **(a-c)**, 2D structure of substituted 2-amino-5-(3-substituted-4-(difluoromethoxy)phenyl)-3-methyl-5-(1,2-disubstituted)phenyl-3,5-dihydro-4*H*-imidazol-4-one **(d)**, substituted 2-amino-3-methyl-5-(3-substituted-phenyl)-5-(1-substituted)pyrazol-4-yl)-3,5-dihydro-4*H*-imidazol-4-one **(e)**. The fragments found by the workflow are marked on the compounds.

Moreover, BACE1 inhibiting compounds containing 2-amino-5-substituted-3-methyl-3,5-dihydro-4*H*-imidazol-4-one moiety (aminohydantoins and iminohydantoin tautomers) can be found widely throughout the literature either as their imidazole-4-one (aminohydantoin) form (shown in mild purple in **Figure 3(d, e)**) [35, 36] or

as their respective iminohydantoin tautomeric form (shown in mustard yellow and orange in **Figure C11**) [37]. Tautomerism is a chemical phenomenon, in which two structural isomers interconvert to each other readily by the migration of a hydrogen atom inside the molecule. The amidine moiety (shown in orange in **Figure C11**) of the iminohydantoin compounds takes place in an H-bond complex with the catalytic residues Asp32 and Asp228 located at the active site of BACE1. According to the mentioned studies, analogs containing either aminohydantoin or iminohydantoin moieties have been shown to be more brain-penetrable and have higher oral bioavailability and BACE1 selectivity. The fragments 8 and 17 found by the workflow represent 4-difluoromethoxyphenyl (shown in salmon) and pyrazolyl (shown in gray) derivative structures substituted to aminohydantoin, respectively. Also, fragment 1 denoting the difluoromethoxy phenyl moiety within studies is stated to have been introduced to aminohydantoin derivatives to afford more potent compounds. It is determined that this moiety occupies the S2' region of BACE1 [35].

3 Discussion

MoleculeNet [38] has four main titles for its datasets: quantum mechanics, physical chemistry, biophysics, and physiology. Quantum mechanics datasets could not be used in this work due to input incompatibility (3D-coordinates). In the context of the remaining tasks' complexity, it is reasonable to assume that learning the physical chemistry tasks (i.e., ESOL, FreeSolv, Lipophilicity) is easier than the rest because the tasks include only information about the chemicals themselves and no information about any external variables. On the other hand, it can be seen that the biophysics tasks (e.g., BACE) are becoming more complex, as the tasks now include not only information about the chemicals themselves, but also information about at least one external variable (e.g., proteins and interactions with proteins). But the most complex ones are undoubtedly the physiology tasks (e.g., BBBP, Tox21, SIDER), which are also regression tasks because there is now much more information due to consecutive systematic interactions at the cell-scale. Overall, it is expected that physical chemistry task scores will be higher than biophysics task scores, and that biophysics task scores will be higher than physiology task scores. Indeed, the obtained benchmark results are consistent with the previously stated expectations. The model ranks only in the top three of all the SOTA models for physical chemistry tasks. The only tested task for biophysics is BACE, and it is the only classification task with a relatively good rank, i.e., fourth to last among all the SOTA models. Finally, with the exception of the ClinTox task, the model is either last or second last in the physiology tasks. We consider that the high success observed in the ClinTox results might be related to a certain type of bias inherent in the ClinTox dataset.

The model demonstrates a clear superiority in regression tasks over classification tasks, ranking third among all the SOTA models. A plausible explanation for this might be the nature of the target values. In regression tasks, the target is a continuous variable, such as a numerical value, with the objective being to predict this value as precisely as possible. On the other hand, classification tasks involve a discrete target variable, like a category or binary label, where the goal is to accurately predict the

correct category or label for each input instance. The continuity of variables in regression tasks allows for a broader range of values, facilitating more accurate predictions. Conversely, classification tasks, with their limited range of potential target values, present a greater challenge in achieving high accuracy [39, 40]. Another contributing factor could be the imbalance in datasets for classification tasks [41, 42], as highlighted in the **Appendix D**, where certain categories or labels are underrepresented, adding complexity to the task.

Interestingly, some fragments within the top 20, particularly in the Tox21 task, show a trend of increasing branch sizes (notably, fragments 4, 6, 9, and 15). The workflow appears to favor larger branches, as evidenced by their higher rankings. Additionally, the recurrence of certain fragments across different tasks is notable. This repetition is attributed to the frequent presence of some small molecules (e.g., benzene, ethene, and propane) in most datasets. Since the workflow takes fragment frequency into account, these molecules consistently appear in the top 20 for various tasks. A noteworthy observation in the Tox21 task is the higher toxicity of the fragment 7 compared to that of fragment 4, indicating that the workflow might not always yield a precise ranking. Furthermore, an analysis could not be conducted for the FreeSolv and SIDER tasks as the fragments identified were predominantly small structures. The small size of the FreeSolv dataset and the complex nature of the SIDER task, focusing on side effects, might have contributed to this outcome. Despite these limitations, the proposed workflow, MIW-ART, is considered effective in emphasizing chemically significant fragments, while acknowledging the subjective nature of "significance".

4 Methods

4.1 Molecular Property Benchmark Tasks

As benchmark tasks, we used five different classification and four different regression datasets from MoleculeNet [38]. According to [38], the physiology datasets are BBBP, ClinTox, SIDER, and Tox21, while the BACE dataset is classified as biophysics. Physical chemistry datasets include ESOL and Lipophilicity, as well as FreeSolv. The total number of molecules and tasks for all datasets can be found in **Appendix D**, which is also contains balances in the distributions of the classification and regression datasets.

4.1.1 BACE Task

β-secretase (BACE) dataset contains molecules that are inhibitors of human β-site amyloid precursor protein cleaving enzyme 1 (BACE1). The dataset contains regression (the half maximal inhibitory concentration (IC_{50}) and classification binding labels of the molecules. Both the regression and the classification datasets consist of 1513 same molecules.

4.1.2 BBBP Task

Blood-brain barrier penetration (BBBP) dataset contains molecules and their classification labels based on their ability to penetrate the blood-brain barrier (BBB). The BBB is a membrane that separates circulating blood from brain extracellular fluid,

inhibits the majority of medications, hormones, and neurotransmitters. As a result, the penetration of the barrier has long been a problem in the development of medications that target the central nervous system. The dataset consists of 2039 molecules.

4.1.3 ClinTox Task

Clinical trial toxicity (ClinTox) dataset contains molecules that have not been approved and approved by the Food and Drug Administration (FDA) due to toxicity. The dataset contains 2 classification tasks. In the experiments, the clinical toxicity (CT_TOX) task [38] was used. The dataset consists of 1478 molecules.

4.1.4 SIDER Task

The side effect resource (SIDER) dataset contains molecules that both are marketed and their adverse drug reactions (ADR). The dataset contains 27 classification tasks. In the experiments, the hepatobiliary disorders task [38] was used. The dataset consists of 1427 molecules.

4.1.5 Tox21 Task

Toxicology in the 21st century (Tox21) dataset contains information on the toxicity of molecules according to different toxicity criteria. The dataset contains 12 classification tasks. In the experiments, the p53 stress-response pathway activation (SR_p53) [38] task was used to determine the toxicity labels of the molecules. The dataset consists of 7831 molecules.

4.1.6 ESOL Task

Estimated solubility (ESOL) dataset contains solubility abilities of the molecules. The dataset consists of 1128 molecules.

4.1.7 FreeSolv Task

The Free Solvation (FreeSolv) dataset contains experimental and calculated hydration free energies of molecules in water. The obtained numbers were obtained through molecular dynamics simulations of alchemical free energy calculations. The dataset consists of 642 molecules.

4.1.8 Lipophilicity Task

Lipophilicity dataset contains experimental results of $n - octanol/water$ distribution coefficient ($\log D$ at pH 7.4) of molecules which affects both membrane permeability and solubility. The dataset consists of 4200 molecules.

4.2 Tokenization of SMILES Representations

Using the Simplified Molecular Input Line Entry System (SMILES) representations, we model molecules as documents derived from a chemical language [43]. For character-based segmentation of SMILES representations, the algorithm from [44] is used. Every

atom except those in salt structures within molecules, every covalent bond except single bonds, every salt structure within molecules, and every molecular branching are represented as different molecular fragments (tokens) in this algorithm.

The fragment dictionary is then obtained by repeating this procedure through all of the benchmark tasks. SMILES representations are encoded into a vector that contains only integers by assigning integers to each fragment in this dictionary, which includes 194 distinct fragments, for later use in building tree structures. **Figure F20(a, b)** in **Appendix F** show a 2D-structured chemical representation and a tree-structured representation of the molecule known as N-methylformamide. It should be noted that the molecule's SMILES representation is `CNC=O` and "-" denotes the absence of a node.

4.3 Experimental Setup

Python3 software [45] is used for all programming tasks, and the PyTorch library [46] is used for all artificial intelligence tasks in this work. First, the aforementioned datasets are downloaded as "scaffold" splits using the DeepChem [47] library. Pandas library [48] is used to read and process datasets saved in comma-separated values (CSV) format.

The AR-Tree model scripts were obtained from [24] and modified to meet the requirements of the changes due to input differences, as it is no longer a normal sentence but a SMILES representation. The AR-Tree model is divided into two sub-models: single-input and double-input versions. Because the only inputs in this work are SMILE representations, the single-input version is chosen. The AR-Tree model also provides a choice between RL and STG, with STG being preferred due to some inconsistencies when summing the main loss with the RL loss to calculate the total loss.

Various hyperparameter combinations are tested during training sessions to determine the best one. To begin, all hyperparameters are tested individually with different values to determine which ones affect the benchmark results. The hyperparameters that affect the benchmark results are then combined, and the resulting combinations are tested. Finally, the best combination of them is used for the model, which is trained from scratch one more time to obtain the best possible checkpoints. The maximum number of epochs is set to 500, and all trainings use early-stopping. The model's structure is explained in detail in the following section (**Section 4.4**). **Figure 4** depicts an abstraction of the model.

The best model checkpoints obtained during the training session are used to evaluate the models' benchmarks. As the training objective for the classification tasks, the binary cross-entropy (BCE) loss function is used. We used the Area Under the Receiver Operating Characteristic Curve (ROC-AUC) as a metric to assess the model's performance. For the regression tasks, the root mean square error (RMSE) loss function is used as both the training objective and the performance metric.

As with the evaluation of the models' benchmarks, the best model checkpoints obtained during the training session are used for the formation and scoring of fragments, as explained in detail in **Sections 4.5** and **4.6**. The PubChemPy library [49] is used to identify Chemical Identification Number (CID) numbers from the PubChem

database [50], and the RDKit library [51] is used to visualize the top 20 fragments out of all the scored fragments.

4.4 Attentive Recursive Tree Model

An input sentence S of N words is represented as $\{\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_N}\}$, where $\mathbf{x_i}$ is a word embedding vector in the D_x-dimensions. An *Attentive Recursive Tree* (AR-Tree) is constructed basicly as a binary tree for each phrase, with R and T standing for the root and the tree's itself, respectively. Each node $t \in T$ has two children marked by $t.left \in T$ and $t.right \in T$ (nil for missing cases) and one word denoted by $t.index$ ($t.index = i$ means the i-th word of input sentence). The $in-order$ traversal of T corresponding to S (i.e., the index of each node in the left subtree of t must be smaller than $t.index$) is ensured to preserve the crucial sequential information. The AR-Tree's most notable characteristic is that words with more task-specific information are located closer to the root.

In order to accomplish the property, a scoring function that evaluates the relative significance of words and top-down recursively selects the word with the highest score is created. A modified Tree-LSTM is used to embed the nodes bottom-up, or from leaf to root, in order to produce the sentence embedding. The downstream tasks use the resulting sentence embedding. Abstract of the model can be seen in **Figure 4**. Pseudo-code of the model architecture can be seen in **Appendix B**.

4.4.1 Top-Down AR-Tree Formation

A bidirectional LSTM is used to process the input phrase and produce a context-sensitive hidden vector for each word. The word hidden state ($\overrightarrow{\mathbf{h_i}}$) and the word cell state ($\overrightarrow{\mathbf{c_i}}$) of the right-directional LSTM are expressed as

$$\overrightarrow{\mathbf{h_i}}, \overrightarrow{\mathbf{c_i}} = \overrightarrow{\mathbf{LSTM}}(\mathbf{x_i}, \overrightarrow{\mathbf{h_{i-1}}}, \overrightarrow{\mathbf{c_{i-1}}}), \tag{1}$$

where $\overrightarrow{\mathbf{h_{i-1}}}$, $\overrightarrow{\mathbf{c_{i-1}}}$, $\mathbf{x_i}$, and $\overrightarrow{\mathbf{LSTM}}$ indicate the previous word hidden state of the right-directional LSTM layer, the previous word cell state of the right-directional LSTM layer, the word embedding vector, and the right-directional LSTM layer, respectively. The word hidden state ($\overleftarrow{\mathbf{h_i}}$) and the word cell state ($\overleftarrow{\mathbf{c_i}}$) of the left-directional LSTM are expressed as

$$\overleftarrow{\mathbf{h_i}}, \overleftarrow{\mathbf{c_i}} = \overleftarrow{\mathbf{LSTM}}(\mathbf{x_i}, \overleftarrow{\mathbf{h_{i+1}}}\overleftarrow{\mathbf{c_{i+1}}}), \tag{2}$$

where $\overleftarrow{\mathbf{h_{i+1}}}$, $\overleftarrow{\mathbf{c_{i+1}}}$, $\mathbf{x_i}$, and $\overleftarrow{\mathbf{LSTM}}$ indicate the next word hidden state of the left-directional LSTM layer, the next word cell state of the left-directional LSTM layer, the word embedding vector, and the left-directional LSTM layer, respectively. The hidden state of the bidirectional LSTM ($\mathbf{h_i}$) is expressed as

$$\mathbf{h_i} = [\overrightarrow{\mathbf{h_i}}; \overleftarrow{\mathbf{h_i}}]. \tag{3}$$

The cell state of the bidirectional LSTM ($\mathbf{c_i}$) is expressed as

$$\mathbf{c_i} = [\overrightarrow{\mathbf{c_i}}; \overleftarrow{\mathbf{c_i}}]. \tag{4}$$

$\mathbf{h}$ is used to score and leave $S = \{\mathbf{h_1}, \mathbf{h_2}, \ldots, \mathbf{h_N}\}$ alone. A trainable scoring function is created based on these context-aware word embeddings to account for the significance of each word and this scoring function is expressed as

$$Score(\mathbf{h_i}) = \mathbf{MLP}(\mathbf{h_i}; \theta), \tag{5}$$

where **MLP** is any multi-layer perceptron that has been parameterized by θ. A 2-layer MLP with 128 hidden units and ReLU activation are employed in particular. Traditional Term Frequency - Inverse Document Frequency (TF-IDF) is a straightforward and obvious way to express the value of words, but it is not intended for certain jobs. It will serve as the starting point.

To build the AR-Tree, a recursive top-down attention-first method is used. Given an input phrase S and the scores for each word, The word with the highest score is chosen as the root R. Then, using recursion, the two subsequences that come before and after the R is used to get the two offspring of the root. The general algorithm for creating the AR-Tree for the sequence $S[b:e] = \{\mathbf{h_b}, \mathbf{h_{b+1}}, \ldots, \mathbf{h_e}\}$ is provided in **Figure B10**. By invoking $R = \text{BUILD}(S, 1, N)$, the whole sentence's AR-Tree and T can be gotten by traversing all of the nodes. Each node in the parsed AR-Tree is the most insightful among its rooted subtree. Because of the fact that any additional data is not utilized in the creation, AR-Tree is applicable to any tasks requiring sentence embedding.

4.4.2 Bottom-Up Tree-LSTM Embedding

After building the AR-Tree, Tree-LSTM [52, 53] is utilized as the composition function to calculate the parent representation from its children and corresponding word in a bottom-up fashion in **Figure F21**. Tree-LSTM inserts cell state into Tree-RNNs to promote improved information flow. Tree-LSTM units may use both the sequential and the structural information to compose semantics since the original word sequence is maintained throughout the in-order traversal of the AR-Tree.

The whole Tree-LSTM composition function is expressed as

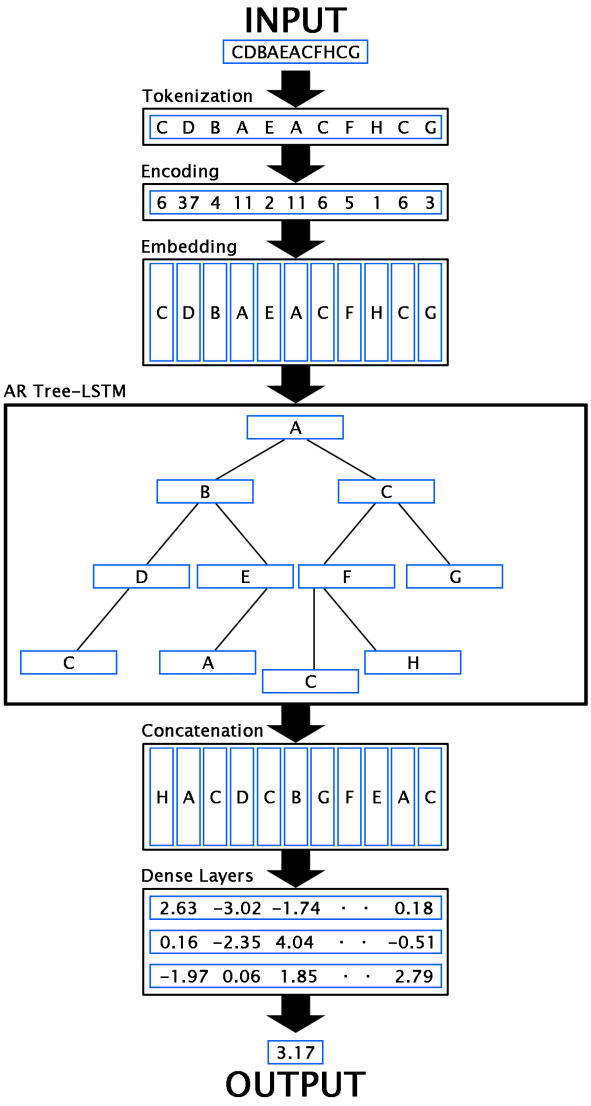

Fig. 4 Abstract of the Attentive Recursive Tree.

$$\begin{bmatrix} \mathbf{ig} \\ \mathbf{f_L} \\ \mathbf{f_R} \\ \mathbf{f_i} \\ \mathbf{o} \\ \mathbf{g} \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ \sigma \\ \sigma \\ tanh \end{bmatrix} \left(\mathbf{W_c} \begin{bmatrix} \mathbf{h_L} \\ \mathbf{h_R} \\ \mathbf{h_i} \end{bmatrix} + \mathbf{b_c} \right), \tag{6}$$

where $\mathbf{ig}$, $\mathbf{f_L}$, $\mathbf{f_R}$, $\mathbf{f_i}$, $\mathbf{o}$, $\mathbf{g}$, σ, and $tanh$ indicate the input gate, the gate of the left child node, the gate of the right child node, the gate of the parent node, the output gate, the candidate vector, the sigmoid function, and the hyperbolic tangent function, respectively. The cell state of the parent node ($\mathbf{c}$) is expressed as

$$\mathbf{c} = (\mathbf{f_L} \odot \mathbf{c_L}) + (\mathbf{f_R} \odot \mathbf{c_R}) + (\mathbf{f_i} \odot \mathbf{c_i}) + (\mathbf{i} \odot \mathbf{g}), \tag{7}$$

where $\mathbf{c_L}$, $\mathbf{c_R}$, and $\mathbf{c_i}$ indicate the cell state of the left child node, the cell state of the right child node, and the cell state of the current word, respectively. The hidden state of the parent node ($\mathbf{h}$) is expressed as

$$\mathbf{h} = \mathbf{o} \odot tanh(\mathbf{c}). \tag{8}$$

While $\mathbf{f_L}$ and $\mathbf{f_R}$ gates control the cell state of the left and right child nodes, $\mathbf{f_i}$ is responsible for adding new information to the current node's cell state. $\mathbf{ig}$ determines the extent to which the cell state will be updated when adding new information. $\mathbf{o}$ converts the embedded representation obtained from the cell state into an output representation. $\mathbf{g}$ is used to update the cell state. σ squeezes values between 0 and 1 which is an activation function commonly used in neural networks. And $tanh$ squeezes values between -1 and 1 which is also another activation function commonly used in neural networks.

To create the node embedding ($\mathbf{h}, \mathbf{c}$), the Tree-LSTM unit combines the semantics of the current word ($\mathbf{h_i}, \mathbf{c_i}$), the right child ($\mathbf{h_R}, \mathbf{c_R}$), and the left child ($\mathbf{h_L}, \mathbf{c_L}$). Zeros are substituted for the missing inputs for nodes that lack certain inputs, such as leaf nodes or nodes with just one child.

Finally, the phrase S is fed onto tasks farther down the line using the embedding $\mathbf{h}$ of the R. Because they are closer to the R and their semantics is naturally highlighted, the sentence embedding will concentrate on those informative terms.

4.5 Creating Dynamic Fragment Dictionary

To begin, all possible subtree formations (fragments) should be discovered in all molecule trees using the corresponding datasets. It is critical to find relatively large-sized fragments rather than small fragments (those with only 2 or 3 atoms except hydrogen atoms). Because larger fragments can be more interpretable and chemically meaningful.

The only way to accomplish this is to form the fragments from leaf to root or bottom-up. Although the root is more discriminative than the other nodes, atoms of the subtree structures formed from the root to the leafs cannot be found side by side in SMILES representations of the corresponding molecules, as shown in **Figure F23**'s tree. This tree is identical to the tree in **Figure F22a** and represents the sentence $\texttt{C}_2\texttt{DBA}_2\texttt{EA}_1\texttt{C}_3\texttt{FHC}_1\texttt{G}$ or, in this case, the SMILES representation. Unfortunately, only the subtree shown in the blue ellipse can form a valid chemical fragment; the others are not. So, in essence, these structures are chemically invalid because the atoms of

the chemical fragments presented as nodes are not connected to one another. This also applies to subtree structures formed between leafs and roots. As a result, forming fragments from leaf to root is the only way to obtain chemically valid subtree structures (fragments). **Figure F22** explains the formation procedure in detail. Subindexes are used to demonstrate that the same tokens can exist at different nodes.

A dynamic fragment dictionary (DFD) is created using the obtained fragments, which is specifically dependent on the corresponding dataset. Then, depending on whether the selected criteria are met, some of the fragments are eliminated. In addition, following the first criteria, all found fragments are canonicalized to obtain a more standardized representation for fragments. The fragment elimination criteria are shown in **Figure G25** in **Appendix G**.

Finally, remaining fragments in the DFD are scored using the scoring procedure described in the following section (**Section 4.6**).

4.6 Scoring Procedure for Molecule Fragments

To evaluate the model's interpretability, all of the fragments (subtrees) found in the previous section (**Section 4.5**) should be searched in all of the molecules in the dataset DS and rated using a scoring procedure.

In this scoring procedure, each leaf is assigned 1 point, and each node is assigned 1 point node-by-node from the leaves to the R in the molecule tree T, as shown in **Figure F24**, where "level" indicates point level. As a result, the R receives the maximum point in the T.

The points ($\mathbf{P_i}$) of each node are then divided by the distance ($\mathbf{d_{max}}$) between the farthest leaf from the R and the R in the T. As a result, all of the outcomes are between 0 and 1. The total score of the fragment F in the T ($\mathbf{P^{mol}}$) is expressed as

$$\mathbf{P^{mol}} = \sum_{i=1}^{n} \frac{\mathbf{P_i}}{\mathbf{d_{max}}}, \tag{9}$$

where $\mathbf{n}$ represents the total number of repeats of the F in the T. While scoring the fragments, not only the node positions of the F in the T, but also the total repeat count of the F in the DS and the test loss of the T in the DS ($\mathbf{e_k}$) should be taken into account. Scores ($\mathbf{P^{mol}_j}$) of the F are added up among the compounds in the DS. The more frequent fragments have a higher total score when added together. The total score of the F in the DS is divided by $\mathbf{m}$ after summarization. As a result, once again, all of the outcomes fall between 0 and 1. The F's average score in the DS ($\overline{\mathbf{P}}$) is expressed as

$$\overline{\mathbf{P}} = \frac{1}{m} \sum_{j=1}^{m} \mathbf{P^{mol}_j}, \tag{10}$$

where $\mathbf{m}$ represents the F's total repeat count in the DS. The average test loss of the F in the dataset DS ($\overline{\mathbf{e}}$) is expressed as

$$\mathbf{e'_k} = \begin{cases} 1 - \mathbf{e_k}, & regression \\ \mathbf{e_k}, & classification, \end{cases} \tag{11}$$

$$\overline{\mathbf{e}} = \frac{1}{m} \sum_{k=1}^{m} \mathbf{e'_k}, \tag{12}$$

where $\mathbf{e'_k}$ denotes a modified version of $\mathbf{e_k}$ parameter. **Equations 11 and 12** are used to include the $\mathbf{e_k}$ parameter into the general equation. After scoring all of the fragments in the DFD, the set FS of all fragment scores is normalized. The min-max normalization formula [54] is expressed as

$$normalize(\mathbf{x}) = \frac{\mathbf{x} - \mathbf{x_{min}}}{\mathbf{x_{max}} - \mathbf{x_{min}}}, \tag{13}$$

where $\mathbf{x}$, $\mathbf{x_{max}}$, and $\mathbf{x_{min}}$ denote the unnormalized FS value, the maximum FS value, and the minimum FS value. **Equation 13** is used for normalization, which scales values in a set between 0 and 1. The final score of the F in the FS ($\mathbf{P_F}$) is expressed as

$$\mathbf{P_F} = normalize(\overline{\mathbf{P}} \cdot \overline{\mathbf{e}}). \tag{14}$$

Finally, all fragments were sorted in descending order. Because fragment scores after the first 20 are significantly lower, it was decided to only analyze the top 20 fragments across all datasets.

References

[1] Huang, B., Von Lilienfeld, O.A.: Communication: Understanding molecular representations in machine learning: The role of uniqueness and target similarity. The Journal of Chemical Physics **145**(16), 161102 (2016)

[2] David, L., Thakkar, A., Mercado, R., Engkvist, O.: Molecular representations in ai-driven drug discovery: A review and practical guide. Journal of Cheminformatics **12**(1), 1–22 (2020)

[3] Kim, J., Park, S., Min, D., Kim, W.: Comprehensive survey of recent drug discovery using deep learning. International Journal of Molecular Sciences **22**(18), 9983 (2021)

[4] Shrikumar, A., Greenside, P., Kundaje, A.: Learning important features through propagating activation differences. In: Proceedings of the 34th International Conference on Machine Learning, vol. 70, pp. 3145–3153 (2017)

[5] Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Computer Vision - ECCV, vol. 8689, pp. 818–833 (2014)

[6] Zintgraf, L.M., Cohen, T.S., Adel, T., Welling, M.: Visualizing deep neural network decisions: Prediction difference analysis. In: International Conference on Learning Representations (2017)

[7] Klöppel, S., Stonnington, C., Chu, C., Draganski, B., Scahill, R., Rohrer, J., Fox, N., Jack, C., Ashburner, J., Frackowiak, R.: Automatic classification of mr scans in alzheimer's disease. Brain: A Journal of Neurology **131**(3), 681–689 (2008)

[8] Ecker, C., Marquand, A., Mourão-Miranda, J., Johnston, P., Daly, E., Brammer, M., Maltezos, S., Murphy, C., Robertson, D., Williams, S., Murphy, D.: Describing the brain in autism in five dimensions-magnetic resonance imaging-assisted diagnosis of autism spectrum disorder using a multiparameter classification approach. The Journal of Neuroscience: The Official Journal of the Society for Neuroscience **30**(32), 10612–10623 (2010)

[9] Gaonkar, B., Davatzikos, C.: Analytic estimation of statistical significance maps for support vector machine-based multivariate image analysis and classification. NeuroImage **78**(9), 270–283 (2013)

[10] Haufe, S., Meinecke, F., Görgen, K., Dähne, S., Haynes, J.-D., Blankertz, B., Biessmann, F.: On the interpretation of weight vectors of linear models in multivariate neuroimaging. NeuroImage **87**(10), 96–110 (2013)

[11] Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., Samek, W.: On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. Public Library of Science One **10**(7), 1–46 (2015)

[12] Simonyan, K., Vedaldi, A., Zisserman, A.: Deep inside convolutional networks: Visualising image classification models and saliency maps. In: Workshop at International Conference on Learning Representations (2014)

[13] Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. International Journal of Computer Vision **128**(2), 336–359 (2019)

[14] Sundararajan, M., Taly, A., Yan, Q.: Gradients of Counterfactuals. arXiv 1611.02639 (2016)

[15] Bowman, S.R., Gauthier, J., Rastogi, A., Gupta, R., Manning, C.D., Potts, C.: A fast unified model for parsing and sentence understanding. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, vol. 1, pp. 1466–1477 (2016)

[16] Yogatama, D., Blunsom, P., Dyer, C., Grefenstette, E., Ling, W.: Learning

to compose words into sentences with reinforcement learning. In: International Conference on Learning Representations (2017)

[17] Maillard, J., Clark, S., Yogatama, D.: Jointly learning sentence embeddings and syntax with unsupervised tree-lstms. Natural Language Engineering **25**(4), 433–449 (2019)

[18] Choi, J., Yoo, K.M., Lee, S.-g.: Learning to compose task-specific tree structures. In: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence, pp. 5094–5101 (2018)

[19] Santos, C.D., Tan, M., Xiang, B., Zhou, B.: Attentive Pooling Networks. arXiv 1602.03609 (2016)

[20] Munkhdalai, T., Yu, H.: Neural tree indexers for text understanding. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, vol. 1, pp. 11–21 (2017)

[21] Arora, S., Liang, Y., Ma, T.: A simple but tough-to-beat baseline for sentence embeddings. In: International Conference on Learning Representations (2017)

[22] Lin, Z., Feng, M., Santos, C.N., Yu, M., Xiang, B., Zhou, B., Bengio, Y.: A structured self-attentive sentence embedding. In: International Conference on Learning Representations (2017)

[23] Kim, Y., Denton, C., Hoang, L., Rush, A.M.: Structured attention networks. In: International Conference on Learning Representations (2017)

[24] Shi, J., Hou, L., Li, J., Liu, Z., Zhang, H.: Learning to embed sentences using attentive recursive trees. Proceedings of the Association for the Advancement of Artificial Intelligence Conference on Artificial Intelligence **33**(1), 6991–6998 (2019)

[25] Wang, Y., Wang, J., Cao, Z., Barati Farimani, A.: Molecular contrastive learning of representations via graph neural networks. Nature Machine Intelligence **4**(3), 279–287 (2022)

[26] Ahmad, W., Simon, E., Chithrananda, S., Grand, G., Ramsundar, B.: ChemBERTa-2: Towards Chemical Foundation Models. arXiv 2209.01712 (2022)

[27] Ross, J., Belgodere, B., Chenthamarakshan, V., Padhi, I., Mroueh, Y., Das, P.: Large-scale chemical language representations capture molecular structure and properties. Nature Machine Intelligence **4**(12), 1256–1264 (2022)

[28] Hu, W., Liu, B., Gomes, J., Zitnik, M., Liang, P., Pande, V., Leskovec, J.:

Strategies for pretraining graph neural networks. In: International Conference on Learning Representations (2020)

[29] Hamada, Y.: Drug discovery of β-secretase inhibitors based on quantum chemical interactions for the treatment of alzheimer's disease. Symbiosis Open Access Journals Pharmacy & Pharmaceutical Sciences **1**(3), 1–8 (2014)

[30] Mouchlis, V.D., Melagraki, G., Zacharia, L.C., Afantitis, A.: Computer-aided drug design of β-secretase, γ-secretase and anti-tau inhibitors for the discovery of novel alzheimer's therapeutics. International Journal of Molecular Sciences **21**(3), 703 (2020)

[31] Moussa-Pacha, N.M., Abdin, S.M., Omar, H.A., Alniss, H., Al-Tel, T.H.: Bace1 inhibitors: Current status and future directions in treating alzheimer's disease. Medicinal Research Reviews **40**(1), 339–384 (2020)

[32] Weiss, M.M., Williamson, T., Babu-Khan, S., Bartberger, M.D., Brown, J., Chen, K., Cheng, Y., Citron, M., Croghan, M.D., Dineen, T.A., *et al.*: Design and preparation of a potent series of hydroxyethylamine containing β-secretase inhibitors that demonstrate robust reduction of central β-amyloid. Journal of Medicinal Chemistry **55**(21), 9009–9024 (2012)

[33] Dineen, T.A., Weiss, M.M., Williamson, T., Acton, P., Babu-Khan, S., Bartberger, M.D., Brown, J., Chen, K., Cheng, Y., Citron, M., *et al.*: Design and synthesis of potent, orally efficacious hydroxyethylamine derived β-site amyloid precursor protein cleaving enzyme (bace1) inhibitors. Journal of Medicinal Chemistry **55**(21), 9025–9044 (2012)

[34] Kaller, M.R., Harried, S.S., Albrecht, B., Amarante, P., Babu-Khan, S., Bartberger, M.D., Brown, J., Brown, R., Chen, K., Cheng, Y., *et al.*: A potent and orally efficacious, hydroxyethylamine-based inhibitor of β-secretase. American Chemical Society Medicinal Chemistry Letters **3**(11), 886–891 (2012)

[35] Malamas, M.S., Robichaud, A., Erdei, J., Quagliato, D., Solvibile, W., Zhou, P., Turner, J., Wagner, E., Fan, K., Olland, A., *et al.*: Design and synthesis of aminohydantoins as potent and selective human β-secretase (bace1) inhibitors with enhanced brain permeability. Bioorganic & Medicinal Chemistry Letters **20**(22), 6597–6605 (2010)

[36] Malamas, M.S., Erdei, J., Gunawan, I., Barnes, K., Hui, Y., Johnson, M., Robichaud, A., Zhou, P., Yan, Y., Solvibile, W., *et al.*: New pyrazolyl and thienyl aminohydantoins as potent bace1 inhibitors: Exploring the s2' region. Bioorganic & Medicinal Chemistry Letters **21**(18), 5164–5170 (2011)

[37] Cumming, J.N., Smith, E.M., Wang, L., Misiaszek, J., Durkin, J., Pan, J., Iserloh, U., Wu, Y., Zhu, Z., Strickland, C., *et al.*: Structure based design of iminohydantoin bace1 inhibitors: Identification of an orally available, centrally active bace1

inhibitor. Bioorganic & Medicinal Chemistry Letters **22**(7), 2444–2449 (2012)

[38] Wu, Z., Ramsundar, B., Feinberg, E.N., Gomes, J., Geniesse, C., Pappu, A.S., Leswing, K., Pande, V.: Moleculenet: A benchmark for molecular machine learning. Chemical Science **9**(2), 513–530 (2018)

[39] Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd edn. Springer, New York, USA (2009)

[40] Kuhn, M., Johnson, K.: Applied Predictive Modeling. Springer, New York, USA (2013)

[41] Bishop, C.: Pattern Recognition and Machine Learning. Springer, New York, USA (2006)

[42] Geron, A.: Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, 2nd edn. O'Reilly Media, Sebastopol, USA (2019)

[43] Weininger, D.: Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. Journal of Chemical Information and Computer Sciences **28**(1), 31–36 (1988)

[44] Schwaller, P., Laino, T., Gaudin, T., Bolgar, P., Hunter, C.A., Bekas, C., Lee, A.A.: Molecular transformer: A model for uncertainty-calibrated chemical reaction prediction. American Chemical Society Central Science **5**(9), 1572–1583 (2019)

[45] Van Rossum, G., Drake, F.L.: Python 3 Reference Manual. Scotts Valley, CA (2009). accessed on 15 August, 2023

[46] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: PyTorch: An Imperative Style, High-Performance Deep Learning Library. Curran Associates, New York, USA (2019)

[47] Ramsundar, B., Eastman, P., Walters, P., Pande, V., Leswing, K., Wu, Z.: Deep Learning for the Life Sciences. O'Reilly Media, Sebastopol, USA (2019)

[48] McKinney, W., *et al.*: Data structures for statistical computing in python. In: Proceedings of the 9th Python in Science Conference, vol. 445, pp. 51–56 (2010)

[49] Swain, M., Sjögren, R., zachcp, Hsiao, Y., Lazzaro, L., Dahlgren, B.: PubChemPy: A Way to Interact with PubChem in Python. https://pubchempy.readthedocs.io. accessed on August 15, 2023 (2017)

[50] Kim, S., Chen, J., Cheng, T., Gindulyte, A., He, J., He, S., Li, Q., Shoemaker, B.A., Thiessen, P.A., Yu, B., Zaslavsky, L., Zhang, J., Bolton, E.E.: PubChem 2023 Update. Nucleic Acids Research **51**(D1), 1373–1380 (2022)

[51] Landrum, G.: RDKit: Open-Source Cheminformatics Software. https://www.rdkit.org. accessed on August 15, 2023 (2016)

[52] Zhu, X., Sobhani, P., Guo, H.: Long Short-Term Memory Over Tree Structures. arXiv 1503.04881 (2015)

[53] Tai, K.S., Socher, R., Manning, C.D.: Improved semantic representations from tree-structured long short-term memory networks. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, vol. 1, pp. 1556–1566 (2015)

[54] Patro, S.G.K., Sahu, K.K.: Normalization: A preprocessing stage. International Advanced Research Journal in Science, Engineering and Technology **2**(3) (2015)

[55] Landry, M.L., Crawford, J.J.: Log *d* contributions of substituents commonly used in medicinal chemistry. American Chemical Society Medicinal Chemistry Letters **11**(1), 72–76 (2019)

[56] Nepali, K., Lee, H.-Y., Liou, J.-P.: Nitro-group-containing drugs. Journal of Medicinal Chemistry **62**(6), 2851–2893 (2018)

[57] Kovacic, P., Somanathan, R.: Nitroaromatic compounds: Environmental toxicity, carcinogenicity, mutagenicity, therapy and mechanism. Journal of Applied Toxicology **34**(8), 810–824 (2014)

[58] Gross, P., Smith, R.P.: Biologic activity of hydroxylamine: A review. Chemical Rubber Company Critical Reviews in Toxicology **14**(1), 87–99 (1985)

[59] Nunes, J.H., Nakahata, D.H., Lustri, W.R., Corbi, P.P., Paiva, R.E.: The nitro-reduced metabolite of nimesulide: Crystal structure, spectroscopic characterization, esi-qtof mass spectrometric analysis and antibacterial evaluation. Journal of Molecular Structure **1157**, 469–475 (2018)

[60] James, L.P., Mayeux, P.R., Hinson, J.A.: Acetaminophen-induced hepatotoxicity. Drug Metabolism and Disposition **31**(12), 1499–1506 (2003)

[61] Ahmed Laskar, A., Younus, H.: Aldehyde toxicity and metabolism: The role of aldehyde dehydrogenases in detoxification, drug resistance and carcinogenesis. Drug Metabolism Reviews **51**(1), 42–64 (2019)

[62] LoPachin, R.M., Gavin, T.: Molecular mechanisms of aldehyde toxicity: A chemical perspective. Chemical Research in Toxicology **27**(7), 1081–1091 (2014)

[63] Mezey, E.: Ethanol metabolism and ethanol-drug interactions. Biochemical Pharmacology **25**(8), 869–875 (1976)

[64] Blanchet, B., Billemont, B., Barete, S., Garrigue, H., Cabanes, L., Coriat, R., Francès, C., Knebelmann, B., Goldwasser, F.: Toxicity of sorafenib: Clinical and molecular aspects. Expert Opinion on Drug Safety **9**(2), 275–287 (2010)

[65] Gordon, G.B., Spielberg, S.P., Blake, D.A., Balasubramanian, V.: Thalidomide teratogenesis: Evidence for a toxic arene oxide metabolite. Proceedings of the National Academy of Sciences **78**(4), 2545–2548 (1981)

[66] Brouwer, A., Ahlborg, U.G., Berg, M., Birnbaum, L.S., Boersma, E.R., Bosveld, B., Denison, M.S., Gray, L.E., Hagmar, L., Holene, E., *et al.*: Functional aspects of developmental toxicity of polyhalogenated aromatic hydrocarbons in experimental animals and human infants. European Journal of Pharmacology: Environmental Toxicology and Pharmacology **293**(1), 1–40 (1995)

[67] Boelsterli, U.A.: Mechanisms of nsaid-induced hepatotoxicity: Focus on nimesulide. Drug Safety **25**, 633–648 (2002)

[68] Richardson, K., Cooper, K., Marriott, M., Tarbit, M., Troke, P., Whittle, P.: Design and evaluation of a systemically active agent, fluconazole. Annals of the New York Academy of Sciences **544**(1), 4–11 (1988)

[69] Richardson, K., Cooper, K., Marriott, M., Tarbit, M., Troke, F., Whittle, P.: Discovery of fluconazole, a novel antifungal agent. Reviews of Infectious Diseases **12**(3), 267–271 (1990)

[70] Boy, K.M., Guernon, J.M., Wu, Y.-J., Zhang, Y., Shi, J., Zhai, W., Zhu, S., Gerritz, S.W., Toyn, J.H., Meredith, J.E., *et al.*: Macrocyclic prolinyl acyl guanidines as inhibitors of β-secretase (bace). Bioorganic & Medicinal Chemistry Letters **25**(22), 5040–5047 (2015)

[71] Gerritz, S.W., Zhai, W., Shi, S., Zhu, S., Toyn, J.H., Meredith Jr, J.E., Iben, L.G., Burton, C.R., Albright, C.F., Good, A.C., *et al.*: Acyl guanidine inhibitors of β-secretase (bace-1): Optimization of a micromolar hit to a nanomolar lead via iterative solid-and solution-phase library synthesis. Journal of Medicinal Chemistry **55**(21), 9208–9223 (2012)

[72] Lipinski, C.A., Lombardo, F., Dominy, B.W., Feeney, P.J.: Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. Advanced Drug Delivery Reviews **23**(1-3), 3–25 (1997)