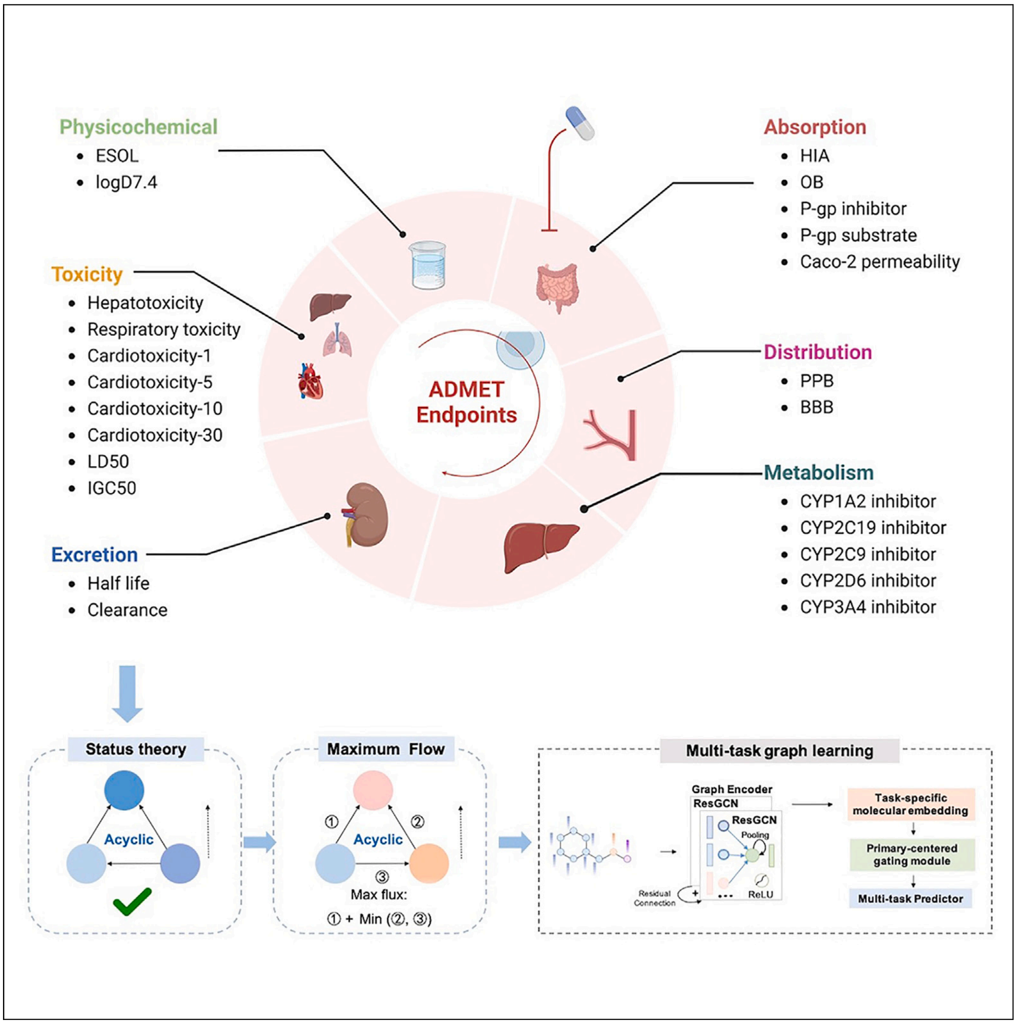
iScience

CellPress
OPEN ACCESS

Article

ADMET property prediction via multi-task graph learning under adaptive auxiliary task selection



Bing-Xue Du, Yi Xu, Siu-Ming Yiu, Hui Yu, Jian-Yu Shi

huiyu@nwpu.edu.cn (H.Y.)
jianyushi@nwpu.edu.cn (J.-Y.S.)

Highlights

We developed a multi-task graph learning model MTGL-ADMET for drug ADMET prediction

MTGL-ADMET employed status theory and maximum flow to adaptively select auxiliary

MTGL-ADMET demonstrated outstanding performance in single-task and multi-task methods

MTGL-ADMET identifies key molecular substructures related to the specific ADMET tasks

Du et al., iScience *26*, 108285
November 17, 2023 © 2023 The Author(s).
https://doi.org/10.1016/j.isci.2023.108285

iScience

CellPress
OPEN ACCESS

Article

ADMET property prediction via multi-task graph learning under adaptive auxiliary task selection

Bing-Xue Du,[1] Yi Xu,[1] Siu-Ming Yiu,[2] Hui Yu,[3,*] and Jian-Yu Shi[1,4,*]

SUMMARY

It is a critical step in lead optimization to evaluate the absorption, distribution, metabolism, excretion, and toxicity (ADMET) properties of drug-like compounds. Classical single-task learning (STL) has effectively predicted individual ADMET endpoints with abundant labels. Conversely, multi-task learning (MTL) can predict multiple ADMET endpoints with fewer labels, but ensuring task synergy and highlighting key molecular substructures remain challenges. To tackle these issues, this work elaborates a multi-task graph learning framework for predicting multiple ADMET properties of drug-like small molecules (MTGL-ADMET) by holding a new paradigm of MTL, "one primary, multiple auxiliaries." It first adeptly combines status theory with maximum flow for auxiliary task selection. The subsequent phase introduces a primary-task-centric MTL model with integrated modules. MTGL-ADMET not only outstrips existing STL and MTL methods but also offers a transparent lens into crucial molecular substructures. It is anticipated that this work can promote lead compound finding and optimization in drug discovery.

INTRODUCTION

Pharmaceutical companies usually spend approximately 10 years and 1.1 billion dollars in discovering and developing a novel drug.[1,2] One of the undesired and painful events in such a costly process is the failure of drug candidates at clinical trials. It is mainly caused by undesirable pharmacokinetic (PK) properties or unacceptable toxicities.[3] By leveraging experimental and clinical data, AI-based computational methods are promising in a rapid and low-risk manner to predict the PK properties and toxicities of drug-like molecules before performing clinical trials.[4,5] Aiming at this perspective, machine-learning-based (ML-based) methods, including classical shallow learning and modern deep learning, were popularly developed over the past years. Their typical tasks cover predicting absorption, distribution, metabolism, excretion, and toxicity (ADMET) and other physicochemical properties of small-molecule compounds.

Through building a predictor for each ADMET endpoint task, ML-based approaches can precisely infer unknown properties for newly given compounds. Their success relies on known ADMET properties (labels) of compounds and molecular representation algorithms (Figure 1A), where the latter is referred to as feature extraction in classical machine learning and as embedding in modern deep learning. For example, MoleculeNet contributes a library of machine-learning-based ADMET predictive approaches,[6] including classical machine learning (support vector machines, random forest, etc.) and modern deep learning (deep neural networks and graph neural networks [GNNs], etc.). In general, these methods belong to the paradigm of single-task learning (STL), "one model, one task," where sufficient labels are one of the crucial factors when training a good predictive model (Figure 1B). However, it is costly to acquire multiple molecular properties in most practical cases. The resulting scarce labels would cause poor molecular representations and trigger the overfitting issue, further resulting in a poor prediction of AMDET properties. To alleviate this predicament, two modern representation techniques, pre-training and fine-tuning, are utilized to achieve improved predictions in the case of scarce labels. The pre-training learns good initial molecular representations by abundant unlabeled data, whereas the fine-tuning further learns task-specific molecular representations.[7–9] These approaches can be regarded as a prototype of multi-task learning.

Multi-task learning (MTL) does not require a compound to be measured by all the ADMET properties. It solves multiple tasks at the same time while exploiting commonalities and differences across ADMET endpoint tasks. During its training, the underlying knowledge among ADMET endpoints can be transferred between them such that the issue of scarce labels can be compensated.[10] The superiority of MTL to STL is demonstrated by recent works in ADMET endpoint predictions.[11] In common, the models in these works first design a GNNs model (e.g., graph convolution network [GCN],[12] relational graph convolution network [R-GCN],[13] and graph isomorphism network[14]) to extract task-shared embeddings. Then, they leverage parallel fully connected neural networks to generate task-specific embeddings for multiple ADMET endpoints simultaneously. In short, existing MTL-based ADMET models follow the paradigm "one-model-fits-all tasks" (Figure 1C). However, such joint learning cannot guarantee that one can always achieve better performance by MTL than that by STL w.r.t. a specific task, because it

[1] School of Life Sciences, Northwestern Polytechnical University, Xi'an 710072, China
[2] Department of Computer Science, The University of Hong Kong, Hong Kong, China
[3] School of Life Sciences, Northwestern Polytechnical University, Xi'an 710072, China
[4] Lead contact
*Correspondence: huiyu@nwpu.edu.cn (H.Y.), jianyushi@nwpu.edu.cn (J.-Y.S.)
https://doi.org/10.1016/j.isci.2023.108285

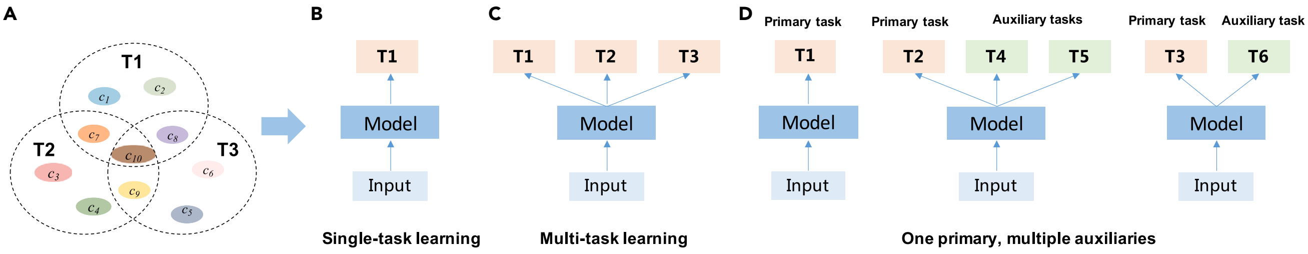


Figure 1. Learning paradigms

(A) c_1-c_{10} represents different compounds and T1-T6 represents different ADMET endpoints.

(B) "One model, one task" in the single-task learning framework.

(C) "One-model-fits-all tasks" in the popular multi-task learning framework.

(D) "One primary, multiple auxiliaries" in our new multi-task learning framework.

assumes that all the tasks have the same ranking or assume a certain learning trade-off among tasks (e.g., task-specific weights in the loss function).[10] In other words, one model cannot fit all tasks in general. It depends on whether appropriate auxiliary tasks can be selected, which is still an ongoing issue.[15]

We hold two assumptions in mind. We first believe that the utilization of task associations can boost the selection of appropriate auxiliary tasks. For example, cytochrome P450 enzymes (CYP450s) are hemoproteins that participate in the metabolism of compounds. Their inhibition can increase the plasma concentration (i.e., an endpoint of Distribution), reduce the Clearance (an endpoint of Excretion), and prolong the half-life (another endpoint of Excretion) of therapeutic agents.[16] The associations between tasks could help select auxiliary endpoint tasks. Moreover, we believe that the presence of certain substructures of a compound is strongly related to its PK endpoint properties. For instance, the absorption endpoint, human intestinal absorption (HIA), is usually related to hydrophilicity functional groups.[17] Similarly, another case of excretion endpoint, Clearance, is usually related to lipophilicity functional groups.[18] The capture of important functional groups (i.e., task-specific crucial substructures) could provide insights into the underlying mechanism of different ADMET properties for a compound/drug.

Based on these ideas, we propose a new paradigm of MTL, "one primary, multiple auxiliaries," where appropriate auxiliary tasks are adaptively selected to boost the primary task even with their own degradation (Figure 1D). We first build a task association network by training individual and pairwise tasks. Then, we jointly leverage the status theory and the maximum flow in complex network science to adaptively collect appropriate tasks (i.e., auxiliary tasks) for each specific task (i.e., the primary task). For each group of "primary-auxiliaries" tasks, we construct a multi-task graph learning framework for predicting multiple ADMET properties of drug-like small molecules (MTGL-ADMET). The model technically includes a task-shared atom embedding module, a task-specific molecular embedding module, a primary task-centered gating module, and a multi-task predictor. In brief, our contributions are as follows: (1) holding the paradigm "one primary, multiple auxiliaries," we design a novel adaptive task selection algorithm, which utilizes the status theory to determine friendly auxiliaries of a specific task and the maximum flow to estimate the increments of MTL w.r.t. STL to obtain optimal auxiliary tasks. (2) We provide a novel model of MTL-based ADMET prediction, where atom embeddings are shared by multiple tasks and they are aggregated further into generating task-specific molecular embeddings. (3) By the aggregation weights of atoms, the proposed model provides an interpretable manner to indicate crucial compound substructures significantly associated with each ADMET task. The multi-task graph learning framework enhanced by status theory and maximum flow (MTGL-ADMET) is shown in Figure 2.

RESULTS

Comparisons with state-of-the-art

We assessed our MTGL-ADMET by the comparison with three state-of-the-art multi-task learning models, which commonly use GNN but different MTL architectures under the paradigm "one-model-fits-all tasks." They are briefly summarized as follows:

MT-GCN[19]: it extends GCNs into the MTL architecture. It utilizes a two-layer GCN module with mean pooling and maximum pooling to generate task-shared molecular embeddings and a group of task-specific four-layer fully connected neural networks to generate task-specific molecular embeddings. Its default values of parameters were used to train the model in the following experiments. Also, we denote its single-task form as ST-GCN.

MT-GCNAtt: we designed an extension of MT-GCN, MT-GCNAtt, by adding an extra attention block for each task after its GCN module. The parameters in the attention blocks are the same as those in our model, whereas other parameters are directly taken from the original MT-GCN.

MGA[13]: adopting a similar architecture to MT-GCN, it uses two-layer R-GCNs[20] (an extension of regular GCNs) to generate task-shared molecular embeddings and builds a task-specific attention layer followed by a three-layer fully connected neural network for each task. We used all the default values of parameters to train the MGA in the following experiments. Meanwhile, we adopted its single-task form as ST-MGA.

For a fair comparison, we utilized 10 independent experiments for all the methods. In each round of the experiments, the dataset was randomly split into a training set, a validation set, and a testing set by the ratio of 8:1:1 in terms of sample number. The validation set was

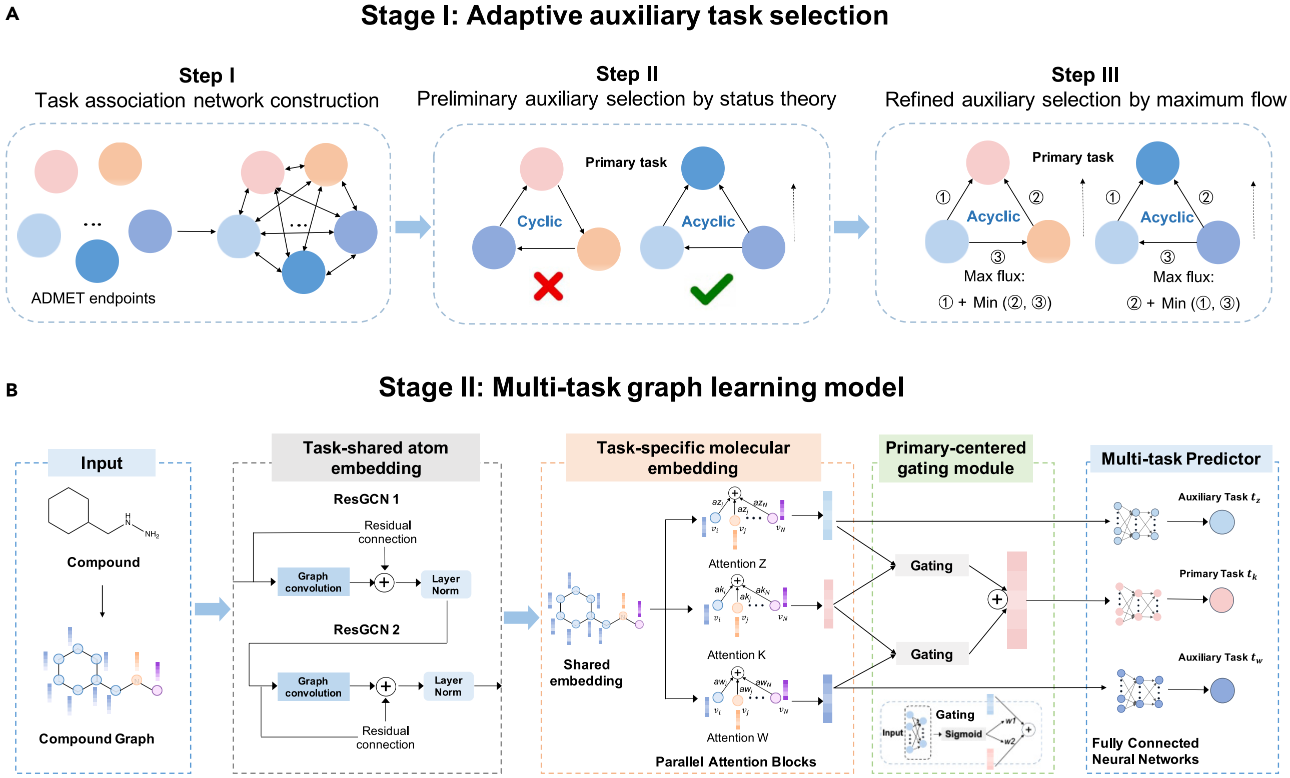


Figure 2. Overview of MTGL-ADMET

(A) The adaptive auxiliary task selection contains three sub-steps.

(B) The novel multi-task learning model for both a primary task and its selected auxiliaries. It is an end-to-end model, which contains a task-shared atom feature module, a task-specific molecular feature module, a primary-task-centered gating module, and a multi-task predictor module from left to right.

used to select tasks. We run 10 repetitions under different random seeds and measured their performance by the average area under the receiver operating characteristic curve (AUC) for classification tasks and the square determination coefficient (R^2) for regression tasks, respectively (Table 1). The greater, the better. We highlighted the result of best in bold and the second best in underline, and the numbers in brackets mean serial numbers of auxiliary tasks w.r.t. each primary task in our model.

The results show that MT-GCN has the worse performance, MT-GCNAtt and MGA exhibit similar performances, and our method achieves the best performance on average with significant improvements over these five approaches by 4.6%, 2.4%, 6.8%, 4.1%, and 4.1%, respectively. Meanwhile, the performance of ST-GCN and ST-MGA is better than their MTL forms, MT-GCN and MGA on average. In terms of individual endpoints, ST-GCN achieves the second best over 4 tasks only, ST-MGA wins the best over 1 task and the second best over 10 tasks, MT-GCN achieves the worst on all tasks, and MT-GCNAtt achieves the second best over 3 tasks, whereas MGA wins the best over 3 tasks and the second best over 3 tasks. In contrast, our MTGL-ADMET wins the best over 20 tasks and the second best over the remaining 4 tasks. Therefore, the comparison demonstrates the superiority of our MTGL-ADMET.

Ablation studies

We evaluated the proposed selection of primary-specific auxiliary tasks by the comparison with three selection strategies, including an individual-task strategy and two designated multiple-task strategies. The individual-task strategy, denoted as Single, trained predictors independently for individual ADMET endpoint tasks. The second strategy (denoted as Rand-5) randomly selected five auxiliaries for a specific task t_k. The third one (denoted as Top-5) selected at most top 5 positive auxiliary tasks of t_k according to the descending order of $\{\widehat{Z}_{i\rightarrow k}\}$ and $\widehat{Z}_{i\rightarrow k} > 0$ (i.e., the fluence of auxiliary t_i on t_k). In contrast, our adaptive selection algorithm results in different numbers of auxiliary endpoints w.r.t. a specific endpoint. See the last column in Table 1.

Overall, our adaptive selection strategy outperforms these selection strategies, and our MTGL-ADMET wins the best over all the tasks (Figure 3). Our method achieves the best performance on average with significant improvements over Single, Rand-5, and Top-5 by 3.16%, 3.60%, and 1.85%, respectively. Especially, our selection algorithm remarkably improves the prediction on two classification tasks ("Hepatotoxicity," Cardiotoxicity-30) with 4.4% and 4.1% increments than the second best. More details can be found in Table S1.

In detail, there are 2 tasks, "P-gp inhibitors" and "respiratory toxicity", having no appropriate auxiliary, 7 tasks having only one auxiliary, 9 tasks having only two auxiliaries, 5 tasks having three auxiliaries, and 1 task having four auxiliaries ("CYP2C9 inhibitor"). These results

Table 1. Performance comparisons of MTGL-ADMET and baselines on 24 ADMET endpoint

No.	Endpoint	Metric	ST-GCN	ST-MGA	MT-GCN	MT-GCNAtt	MGA	MTGL-ADMET[a]
1	HIA	AUC	0.916 ± 0.054	0.972 ± 0.014	0.899 ± 0.057	0.953 ± 0.019	0.911 ± 0.034	**0.981 ± 0.011** (18)
2	OB	AUC	0.716 ± 0.035	0.710 ± 0.035	0.728 ± 0.031	0.726 ± 0.027	0.745 ± 0.029	**0.749 ± 0.022** (14,24)
3	P-gp inhibitors	AUC	0.916 ± 0.012	0.917 ± 0.006	0.895 ± 0.014	0.907 ± 0.009	0.901 ± 0.010	**0.928 ± 0.008** (None)
4	P-gp substrates	AUC	0.775 ± 0.034	0.755 ± 0.014	0.733 ± 0.044	0.730 ± 0.034	0.719 ± 0.035	**0.801 ± 0.031**(18,21)
5	Caco-2 permeability	R^2	0.451 ± 0.033	0.519 ± 0.014	0.374 ± 0.022	0.404 ± 0.017	0.385 ± 0.031	**0.523 ± 0.025** (24)
6	PPB	R^2	0.577 ± 0.028	0.585 ± 0.004	0.589 ± 0.036	0.619 ± 0.025	0.568 ± 0.038	**0.626 ± 0.029** (9)
7	BBB	AUC	0.956 ± 0.008	0.959 ± 0.004	0.945 ± 0.007	0.955 ± 0.009	0.956 ± 0.010	**0.973 ± 0.005** (23)
8	CYP1A2 inhibitor	AUC	0.932 ± 0.007	0.931 ± 0.013	0.914 ± 0.009	0.941 ± 0.008	0.940 ± 0.006	**0.952 ± 0.005** (9)
9	CYP2C19 inhibitor	AUC	0.774 ± 0.012	0.781 ± 0.008	0.775 ± 0.011	0.782 ± 0.011	0.795 ± 0.019	**0.804 ± 0.015** (12,16)
10	CYP2C9 inhibitor	AUC	0.746 ± 0.016	0.764 ± 0.017	0.771 ± 0.016	0.782 ± 0.011	**0.798 ± 0.019**	0.794 ± 0.013 (5,6,11,16)
11	CYP2D6 inhibitor	AUC	0.848 ± 0.016	0.841 ± 0.022	0.839 ± 0.015	0.845 ± 0.015	**0.877 ± 0.017**	0.869 ± 0.016 (5,6)
12	CYP3A4 inhibitor	AUC	0.892 ± 0.006	0.915 ± 0.006	0.865 ± 0.007	0.896 ± 0.011	0.875 ± 0.006	**0.916 ± 0.007** (11)
13	Half-life	AUC	0.725 ± 0.011	0.708 ± 0.024	0.688 ± 0.035	0.699 ± 0.028	0.707 ± 0.017	**0.727 ± 0.022** (14)
14	Clearance	AUC	0.723 ± 0.030	0.710 ± 0.015	0.686 ± 0.031	0.755 ± 0.014	0.740 ± 0.027	**0.779 ± 0.027**(12,22, 24)
15	Hepatotoxicity	AUC	0.653 ± 0.040	0.669 ± 0.022	0.612 ± 0.039	0.640 ± 0.068	**0.713 ± 0.053**	0.701 ± 0.036(8,10, 17)
16	Respiratory toxicity	AUC	0.842 ± 0.018	**0.872 ± 0.013**	0.810 ± 0.014	0.828 ± 0.015	0.828 ± 0.021	0.859 ± 0.010 (None)
17	Cardiotoxicity-1	AUC	0.707 ± 0.026	0.703 ± 0.020	0.683 ± 0.028	0.696 ± 0.028	0.684 ± 0.023	**0.740 ± 0.023** (2,16,23)
18	Cardiotoxicity-5	AUC	0.620 ± 0.015	0.637 ± 0.010	0.626 ± 0.027	0.619 ± 0.015	0.623 ± 0.014	**0.641 ± 0.014** (9,14,19)
19	Cardiotoxicity-10	AUC	0.627 ± 0.013	0.611 ± 0.015	0.609 ± 0.022	0.613 ± 0.021	0.603 ± 0.026	**0.654 ± 0.010** (9,10)
20	Cardiotoxicity-30	AUC	0.664 ± 0.036	0.653 ± 0.036	0.645 ± 0.036	0.687 ± 0.059	0.709 ± 0.035	**0.723 ± 0.029** (6,11, 23)
21	LD50	R^2	0.588 ± 0.018	0.617 ± 0.018	0.503 ± 0.017	0.502 ± 0.023	0.492 ± 0.029	**0.635 ± 0.015** (16,22)
22	IGC50	R^2	0.703 ± 0.055	0.818 ± 0.021	0.618 ± 0.027	0.744 ± 0.032	0.772 ± 0.021	**0.819 ± 0.008** (5,19)
23	ESOL	R^2	0.814 ± 0.030	0.896 ± 0.013	0.824 ± 0.030	0.872 ± 0.018	0.866 ± 0.020	**0.931 ± 0.038** (21,22)
24	logD7.4	R^2	0.759 ± 0.056	0.904 ± 0.008	0.770 ± 0.019	0.838 ± 0.016	0.838 ± 0.018	**0.915 ± 0.008** (1,16)
	Average		0.747 ± 0.025	0.769 ± 0.015	0.725 ± 0.006	0.752 ± 0.022	0.752 ± 0.023	**0.793 ± 0.018**

[a] The numbers in the parenthesis means the auxiliary task numbers for each primary task in our model.

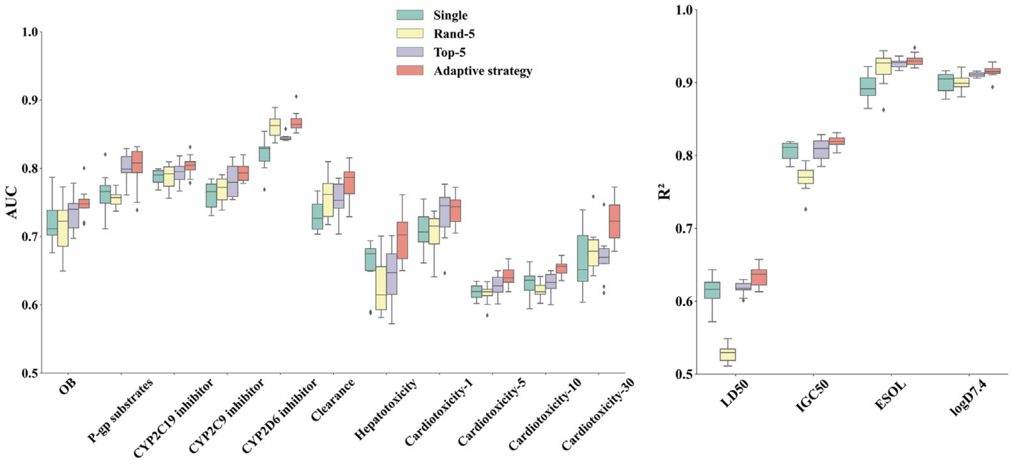


Figure 3. Comparison with auxiliary selection strategies: the Single, Rand-5, and Top-5

The left panel is for classification tasks, whereas the right panel is for regression tasks. Specifically, the Single strategy represents an individual-task approach, the Rand-5 strategy involves randomly selecting five auxiliary tasks for a specific primary task, and the Top-5 strategy entails selecting at most top-5 positive auxiliary tasks.

demonstrate that our selection algorithm relying on the status theory and the maximum flow can select approximate tasks adaptively. It significantly outperforms these strategies over all the endpoint tasks, no matter whether it is a classification task or a regression task.

To investigate why our task selection algorithm strategy is effective, we first calculated the maximum flux over all the primary-specific task groups. Then, we measure the correlation between them and the incremental performance of the multi-task learning to the single-task learning on the testing dataset, i.e., $Z_{k|w,z}^{(m)} - Z_k^{(s)}$. We found a significant Spearman correlation ($\gamma = 0.9562, \rho = 2.6089e - 08$) between the maximum flux and the increment (Figure 4). The results demonstrate that the maximum flow on status-shaped triads can be a good indicator to select appropriate auxiliary tasks for a specific task. More results can be found in Figure S1.

In addition, we investigated how the main components of MTGL-ADMET contribute to the prediction by ablation studies. We designed two variants of MTGL-ADMET (Figure 5). The first one (denoted as w/o Att) eliminates the parallel attention block in the task-specific molecular embedding module. The second one (denoted as w/o Gate) lacks gate networks in the primary-task-centered gate module. MTGL-ADMET significantly outperforms these variants in both classification and regression tasks. Specifically, compared with w/o Att and w/o Gate, MTGL-ADMET improves on average value by 3.1% and 1.1%. In detail, MTGL-ADMET improves the AUC value by 2.45% and 1.16% in classification tasks and the R^2 value by 4.9% and 1.03% in regression tasks. Therefore, the results indicate that the parallel attention blocks and the gate networks play critical roles in predicting ADMET endpoints.

Case study: Interpretability of MTGL-ADMET

Although deep learning is known as a black-box model, it is essential to understand how MTGL-ADMET makes a prediction and whether it can guide lead compound optimization in drug discovery. Because the task-specific molecular embedding module can learn task-specific atom

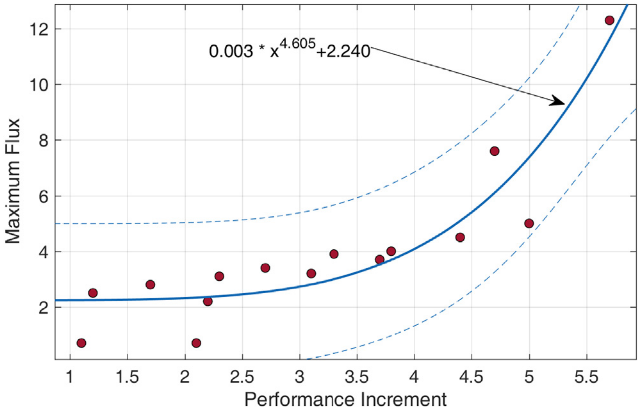


Figure 4. Correlation between the maximum fluxes and the incremental performance on optimal combinations for each endpoint

Each point represents a task group, which contains at least 2 auxiliary tasks. The maximum flux is calculated by the Ford-Fulkerson algorithm. The solid curve indicates a fitting, whereas two dotted curves denote its 95% confidence bounds.

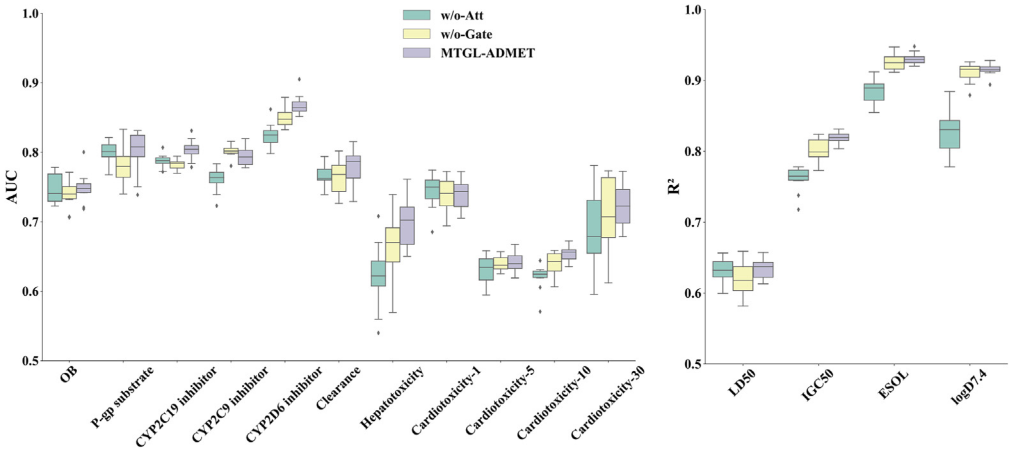


Figure 5. Ablation experiments of each component

The left panel is for classification tasks, whereas the right panel is for regression tasks. Compared with the full MTGL-ADMET, w/o Att eliminates the parallel attention block in the task-specific molecule embedding module, and w/o Gate lacks gating networks in the primary-task-centered gating module.

importance by its task-specific attention layers, we decided that two bonding atoms are regarded as a crucial structure fragment if both of them have high attention weights. The weight of a bond is the average of the weights of its constituent atoms (highlighted in Figure 6). One or more fragments form a crucial substructure, which is specific to endpoint tasks.

We selected eight endpoints as the case study, including HIA (A), OB (A), BBB (D), CYP3A4 and CYP2D6 inhibitors (M), clearance (E), hepatotoxicity (T), and cardiotoxicity-1 (T), where A is for an absorption endpoint, D is for a distribution endpoint, M is for a metabolism endpoint, E is for an excretion endpoint, and T is for a toxicity endpoint, respectively.

First, we picked up two compounds (*Maprotiline* and *Acetohexamide*) having good HIAs and another two compounds (*Paroxetine cation* and *Zonisamide*) having good OBs. As shown in Figure 6A, their crucial substructures indicated by our MTGL-ADMET involve *hydroxyl* and *amino*, of which all are commonly hydrophilic.[21] The solubilization aspect of the absorption process is greatly influenced by the presence of these hydrophilic functional groups.[22,23] However, purely hydrophilic drugs may not be favorable for subsequent permeation.[24] Thus, it is essential to consider the lipophilic functional groups within the molecule when evaluating permeability.

Then, four compounds having good BBBs were investigated (Figure 6B). Similarly, their highlighted substructures involve lipophilic functional groups, (i.e., *phenyl ring*, *morpholinyl*, and *piperidine*), which are helpful to pass the blood-brain barrier.[25–27]

After that, we investigated an important enzyme inhibitor, *Ethinylestradiol*, which belongs to the family of CYP3A4 inhibitors. Two compounds having high affinities with CYP enzymes were focused on. After an extra docking simulation (Autodock), we found that their highlighted substructures involve aromatic rings and hydrophobic fragments (Figure 6C, upper panel), which are the key to the binding site in the pocket by contributing to non-covalent bonds (e.g., H-bonds and Pi-Pi bonds).[28] Also, the result is consistent with the domain knowledge that CYP3A4 inhibitors usually have *furan ring*, *tertiary amine*, or *acetylene* substructures.[28]

More importantly, to investigate whether a compound shows task-specific crucial substructures, we picked up a compound (*1-Allyl-1,5-anhydro-2,3-dideoxy-4-O-(4-fluoro-2-methylphenyl)hex-2-enitol*) inhibiting two kinds of CYP enzymes (CYP 2D6 and CYP 3A4) as the case study (Figure 6C, lower panel). The results validate that its highlighted substructures are specific to two endpoints.

Furthermore, four compounds having good Clearances were selected (Figure 6D). Their highlighted substructures, including *alkyl* and *halogen*, are lipophilic. The results are consistent with the knowledge that compounds having high lipophilicity tend to have high clearance.[18,21] Last, we paid attention to two toxicity endpoints, hepatotoxicity, and cardiotoxicity-1, which represent serious concerns in drug development and are the main reasons for a drug being withdrawn from the market.[29] *Acetohexamide* and *Amodiaquine* have highlighted substructures, *sulfonamide moiety* and *halogen atom* (Figure 6E), whereas *Lidoflazine* and *Bromperidol* have highlighted substructures, a basic nitrogen center flanked by aromatic or hydrophobic groups (Figure 6F). These two groups of substructures agree with the finding in[30] and,[31] respectively.

In summary, the consistency of our findings with domain knowledge and literature demonstrates that MTGL-ADMET is an interpretable model, which can indicate compound substructures (or functional groups) significantly associated with ADMET endpoints. It would help reveal why a compound shows a specific ADMET property of interest.

DISCUSSION

In this paper, holding a new paradigm of MTL, "one primary, multiple auxiliaries," we have proposed a multi-task graph learning framework for predicting various ADMET endpoints of drug-like small molecules (MTGL-ADMET). It contains two stages: adaptive auxiliary task selection and primary-centered multi-task learning. The former stage builds a task association network by training individual and pairwise tasks and leverages both the status theory and the maximum flow in complex network science to adaptively collect appropriate tasks. The latter stage constructs a novel primary-centered multi-task graph learning model to train the primary task and its auxiliary tasks together. The model

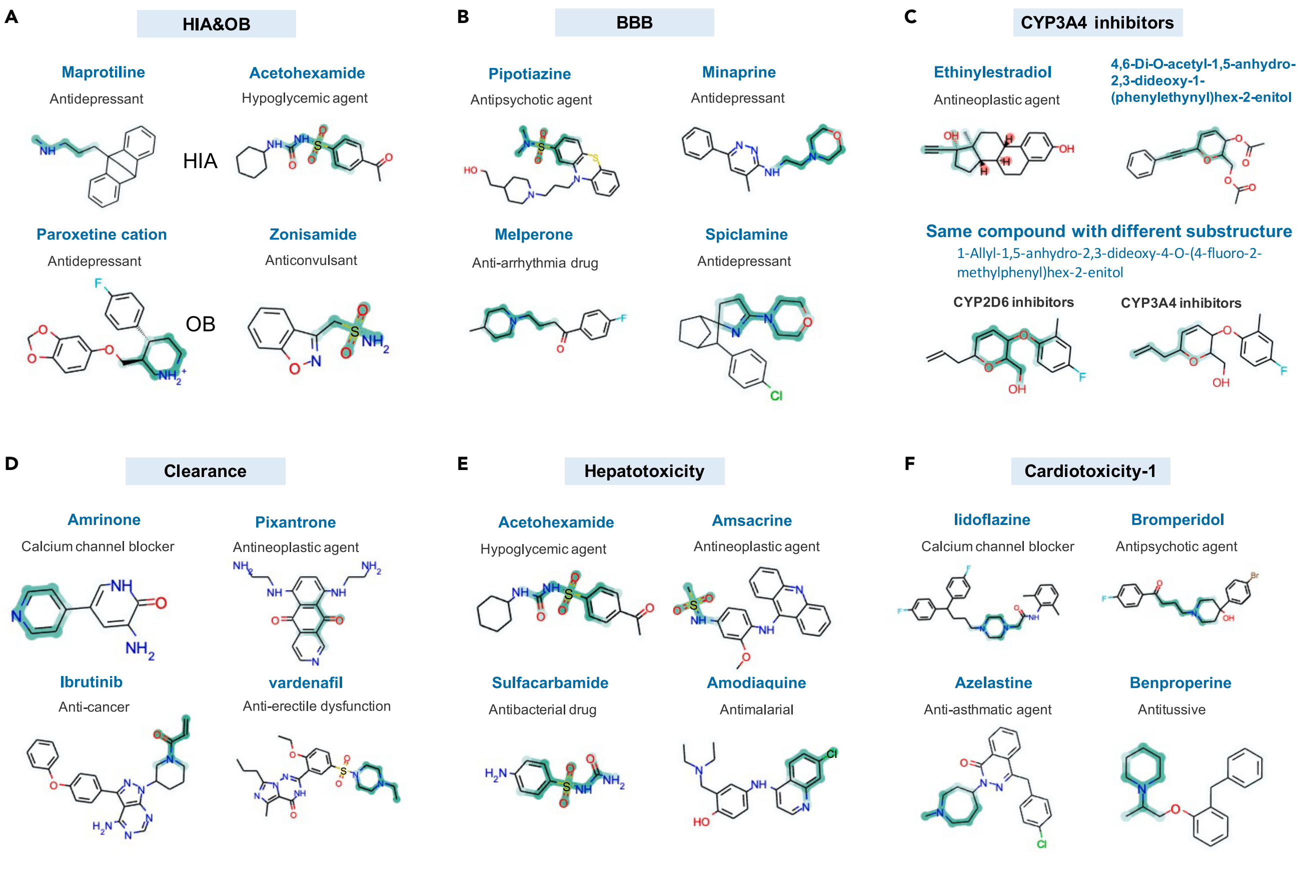


Figure 6. Cases study of crucial substructures

(A) Two compounds of HIA and two of OB.

(B) Four compounds of BBB.

(C) Two CYP3A4 inhibitors and one inhibitor for CYP2D6 and CYP3A4.

(D) Four compounds of clearance.

(E) Four compounds of hepatotoxicity.

(F) Four compounds of cardiotoxicity-1. The atoms and bonds of endpoint-specific critical substructures are highlighted in green.

technically includes a task-shared atom embedding module, a task-specific molecular embedding module, a primary-task-centered gating module, and a multi-task predictor. MTGL-ADMET can address two existing issues, including auxiliary selection and task-specific molecular substructure finding.

The comparison with state-of-the-art MTL-based models demonstrates the superiority of our MTGL-ADMET in terms of prediction performance. More elaborate experiments validate its contributions. First, it improves the selection algorithm of appropriate auxiliary tasks in the MTL by calculating the maximum flux of status-theory-satisfied task triads as the initial estimator. Secondly, by the gating networks, it uncovers the contributions of auxiliary tasks to the primary task, which helps understand ADMET endpoint associations in a quantity manner. Thirdly, by task-shared atom embeddings and task-specific attention scores, it obtains task-specific molecular embeddings with the highlight of crucial compound substructures specific to ADMET endpoints.

In summary, we believe that our study provides new insights into ADMET endpoint prediction and also can be borrowed for other multi-task learning problems (e.g., compound physical-chemical property prediction in drug discovery, object detection, autonomous vehicles, and recommendation systems). In the coming future, the integration of status theory and maximum flow techniques into the architecture of neural networks (e.g., to embed the task association network) would improve the finding of optimal auxiliary tasks.

Reflecting on recent ADMET prediction studies, Zhang et al.[32] emphasize the limitations of traditional ADMET systems due to scarce labeled data. Their work with HelixADMET (H-ADMET) introduces the advantages of self-supervised learning and its potential for knowledge transfer between ADMET endpoints. This aligns with our focus on optimal auxiliary tasks, hinting at the promise of self-supervised learning for our methodology. Additionally, Fang et al.[33] advocate for the integration of fundamental domain knowledge in deep learning through their knowledge-graph-enhanced molecular contrastive learning with functional prompt methodology, which uses a chemical-element-oriented knowledge graph. Their emphasis on interpretability and chemically sound predictions underscores the gaps in purely data-driven models and signals the significance of domain knowledge and interpretability in advancing our approach.

Given these considerations, we posit that future research in this domain can benefit immensely from a synergy of domain-specific knowledge, advanced machine-learning models, and methodologies that prioritize interpretability and robustness. We are optimistic that such integrations will pave the way for more precise, interpretable, and reliable ADMET predictions, advancing drug discovery processes, and multi-task learning applications.

Limitations of the study

An additional layer of complexity in ADMET predictions pertains to the intricate balance between hydrophilicity and lipophilicity in the PK/PD profiles of organic compounds. As noted in feedback from the scientific community, functional groups such as hydroxyl and amino, while increasing hydrophilicity, play indispensable roles in drug-receptor interactions or in metabolic pathways like CYP-mediated reactions. Such nuances underscore the importance of considering the totality of molecular features, rather than isolating specific functional groups. The compensatory addition of non-polar groups to modulate hydrophilic effects further exemplifies the multifaceted nature of these interactions. It is a testament to the fact that although deep learning models provide powerful tools for predictive analytics, they must be employed judiciously, taking into account the intricate interplay of molecular characteristics. This intricate balance and its implications for drug development and ADMET predictions warrant deeper exploration and are areas we aim to further investigate in our subsequent research endeavors.

Despite the powerful insights offered by attention mechanisms in deep learning for ADMET predictions, they are not without limitations in interpretability. Marrying domain-specific knowledge with these models can compensate for such constraints. Our work with MTGL-ADMET underscores the need for a synergistic approach, ensuring that predictions are both accurate and meaningfully interpretable for chemists and biologists.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- RESOURCE AVAILABILITY
  - Lead contact
  - Materials availability
  - Data and code availability
- METHOD DETAILS
  - Dataset and setup
  - Problem formulation
  - Overview of framework
  - Adaptive auxiliary task selection
  - Multi-task graph learning model

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.isci.2023.108285.

ACKNOWLEDGMENTS

The authors would like to thank anonymous reviewers and Associate Professor Meiying Songin for suggestions that improved the paper and graphical abstract and Figure S2 was created with BioRender.com. This work is supported in part by the National Nature Science Foundation of China (Grant number 62372375), Shaanxi Province Key R&D Program (Grant number 2023-YBSF-114), and CAAI-Huawei MindSpore Open Fund (Grant Number CAAIXSJLJJ-2022-035A).

AUTHOR CONTRIBUTIONS

B.D. and J.S. contributed to the study concept and design. B.D., Y.X., and J.S. contributed to data collection, analysis, and interpretation. S.Y. and H.Y. provided important advice and assistance for manuscript drafting. H.Y. and J.S. supervised the study. All authors read and approved the final manuscript and had final responsibility for the decision to submit it for publication.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: August 2, 2023
Revised: September 18, 2023
Accepted: October 18, 2023
Published: October 24, 2023

REFERENCES

1. Schneider, G. (2018). Automating drug discovery. Nat. Rev. Drug Discov. *17*, 97–113.
2. Wouters, O.J., McKee, M., and Luyten, J. (2020). Estimated research and development investment needed to bring a new medicine to market, 2009-2018. JAMA *323*, 844–853.
3. Waring, M.J., Arrowsmith, J., Leach, A.R., Leeson, P.D., Mandrell, S., Owen, R.M., Pairaudeau, G., Pennie, W.D., Pickett, S.D., Wang, J., et al. (2015). An analysis of the attrition of drug candidates from four major pharmaceutical companies. Nat. Rev. Drug Discov. *14*, 475–486.
4. Schneider, P., Walters, W.P., Plowright, A.T., Sieroka, N., Listgarten, J., Goodnow, R.A., Jr., Fisher, J., Jansen, J.M., Duca, J.S., Rush, T.S., et al. (2020). Rethinking drug design in the artificial intelligence era. Nat. Rev. Drug Discov. *19*, 353–364.
5. Jia, C.-Y., Li, J.-Y., Hao, G.-F., and Yang, G.-F. (2020). A drug-likeness toolbox facilitates ADMET study in drug discovery. Drug Discov. Today *25*, 248–258.
6. Wu, Z., Ramsundar, B., Feinberg, E.N., Gomes, J., Geniesse, C., Pappu, A.S., Leswing, K., and Pande, V. (2018). MoleculeNet: a benchmark for molecular machine learning. Chem. Sci. *9*, 513–530.
7. Shen, W.X., Zeng, X., Zhu, F., Wang, Y.L., Qin, C., Tan, Y., Jiang, Y.Y., and Chen, Y.Z. (2021). Out-of-the-box deep learning prediction of pharmaceutical properties by broadly learned knowledge-based molecular representations. Nat. Mach. Intell. *3*, 334–343.
8. Rong, Y., Bian, Y., Xu, T., Xie, W., Wei, Y., Huang, W., and Huang, J. (2020). Self-supervised graph transformer on large-scale molecular data. Adv. Neural Inf. Process. Syst. *33*, 12559–12571.
9. Chen, D., Gao, K., Nguyen, D.D., Chen, X., Jiang, Y., Wei, G.-W., and Pan, F. (2021). Algebraic graph-assisted bidirectional transformers for molecular property prediction. Nat. Commun. *12*, 3521.
10. Ruder, S. (2017). An Overview of Multi-Task Learning in Deep Neural Networks. Preprint at arXiv. https://doi.org/10.48550/arXiv.1706.05098.
11. Bhhatarai, B., Walters, W.P., Hop, C.E.C.A., Lanza, G., and Ekins, S. (2019). Opportunities and challenges using artificial intelligence in ADME/Tox. Nat. Mater. *18*, 418–422.
12. Feinberg, E.N., Joshi, E., Pande, V.S., and Cheng, A.C. (2020). Improvement in ADMET prediction with multitask deep featurization. J. Med. Chem. *63*, 8835–8848.
13. Xiong, G., Wu, Z., Yi, J., Fu, L., Yang, Z., Hsieh, C., Yin, M., Zeng, X., Wu, C., Lu, A., et al. (2021). ADMETlab 2.0: an integrated online platform for accurate and comprehensive predictions of ADMET properties. Nucleic Acids Res. *49*, W5–W14.
14. Peng, Y., Lin, Y., Jing, X.Y., Zhang, H., Huang, Y., and Luo, G.S. (2020). Enhanced Graph Isomorphism Network for Molecular ADMET Properties Prediction. IEEE Access *8*, 168344–168360.
15. Fifty, C., Amid, E., Zhao, Z., Yu, T., Anil, R., and Finn, C. (2021). Efficiently identifying task groupings for multi-task learning. Adv. Neural Inf. Process. Syst. *34*, 27503–27516.
16. Dong, J., Li, S., and Liu, G. (2021). Binimetinib Is a Potent Reversible and Time-Dependent Inhibitor of Cytochrome P450 1A2. Chem. Res. Toxicol. *34*, 1169–1174.
17. Khadka, P., Ro, J., Kim, H., Kim, I., Kim, J.T., Kim, H., Cho, J.M., Yun, G., and Lee, J. (2014). Pharmaceutical particle technologies: An approach to improve drug solubility, dissolution and bioavailability. Asian J. Pharm. Sci. *9*, 304–316.
18. Johnson, T.W., Gallego, R.A., and Edwards, M.P. (2018). Lipophilic Efficiency as an Important Metric in Drug Design. J. Med. Chem. *61*, 6401–6420.
19. Montanari, F., Kuhnke, L., Ter Laak, A., and Clevert, D.-A. (2019). Modeling physico-chemical ADMET endpoints with multitask graph convolutional networks. Molecules *25*, 44.
20. Schlichtkrull, M., Kipf, T.N., Bloem, P., van den Berg, R., Titov, I., and Welling, M. (2018). Modeling Relational Data with Graph Convolutional Networks (Springer). 10843, 593–607.
21. Li, Y., Hsieh, C.-Y., Lu, R., Gong, X., Wang, X., Li, P., Liu, S., Tian, Y., Jiang, D., Yan, J., et al. (2022). An adaptive graph learning method for automated molecular interactions and properties predictions. Nat. Mach. Intell. *4*, 645–651.
22. Wang, Y., Xing, J., Xu, Y., Zhou, N., Peng, J., Xiong, Z., Liu, X., Luo, X., Luo, C., Chen, K., et al. (2015). In silico ADME/T modelling for rational drug design. Q. Rev. Biophys. *48*, 488–515.
23. Stegemann, S., Leveiller, F., Franchi, D., de Jong, H., and Lindén, H. (2007). When poor solubility becomes an issue: from early stage to proof of concept. Eur. J. Pharmaceut. Sci. *31*, 249–261.
24. Cano-Cebrián, M.J., Zornoza, T., Granero, L., and Polache, A. (2005). Intestinal absorption enhancement via the paracellular route by fatty acids, chitosans and others: a target for drug delivery. Curr. Drug Deliv. *2*, 9–22.
25. Zhu, C., Li, X., Zhao, B., Peng, W., Li, W., and Fu, W. (2020). Discovery of aryl-piperidine derivatives as potential antipsychotic agents using molecular hybridization strategy. Eur. J. Med. Chem. *193*, 112214.
26. Ritchie, T.J., and Macdonald, S.J.F. (2009). The impact of aromatic ring count on compound developability–are too many aromatic rings a liability in drug design? Drug Discov. Today *14*, 1011–1020.
27. Khaldan, A., Bouamrane, S., En-Nahli, F., El-mernissi, R., El khatabi, K., Hmamouchi, R., Maghat, H., Ajana, M.A., Sbai, A., Bouachrine, M., and Lakhlifi, T. (2021). Prediction of potential inhibitors of SARS-CoV-2 using 3D-QSAR, molecular docking modeling and ADMET properties. Heliyon *7*, e06603.
28. Beck, T.C., Beck, K.R., Morningstar, J., Benjamin, M.M., and Norris, R.A. (2021). Descriptors of cytochrome inhibitors and useful machine learning based methods for the design of safer drugs. Pharmaceuticals *14*, 472.
29. Onakpoya, I.J., Heneghan, C.J., and Aronson, J.K. (2016). Post-marketing withdrawal of 462 medicinal products because of adverse drug reactions: a systematic review of the world literature. BMC Med. *14*, 10.
30. Liu, R., Yu, X., and Wallqvist, A. (2015). Data-driven identification of structural alerts for mitigating the risk of drug-induced human liver injuries. J. Cheminf. *7*, 4.
31. Cavalluzzi, M.M., Imbrici, P., Gualdani, R., Stefanachi, A., Mangiatordi, G.F., Lentini, G., and Nicolotti, O. (2020). Human ether-à-go-go-related potassium channel: Exploring SAR to improve drug design. Drug Discov. Today *25*, 344–366.
32. Zhang, S., Yan, Z., Huang, Y., Liu, L., He, D., Wang, W., Fang, X., Zhang, X., Wang, F., Wu, H., and Wang, H. (2022). HelixADMET: a robust and endpoint extensible ADMET system incorporating self-supervised knowledge transfer. Bioinformatics *38*, 3444–3453.
33. Fang, Y., Zhang, Q., Zhang, N., Chen, Z., Zhuang, X., Shao, X., Fan, X., and Chen, H. (2023). Knowledge graph-enhanced molecular contrastive learning with functional prompt. Nat. Mach. Intell. *5*, 542–553.
34. Yang, M., Chen, J., Xu, L., Shi, X., Zhou, X., Xi, Z., An, R., and Wang, X. (2018). A novel adaptive ensemble classification framework for ADME prediction. RSC Adv. *8*, 11661–11683.
35. Wang, X., Liu, M., Zhang, L., Wang, Y., Li, Y., and Lu, T. (2020). Optimizing Pharmacokinetic Property Prediction Based on Integrated Datasets and a Deep Learning Approach. J. Chem. Inf. Model. *60*, 4603–4613.
36. Wang, N.N., Deng, Z.K., Huang, C., Dong, J., Zhu, M.F., Yao, Z.J., Chen, A.F., Lu, A.P., Mi, Q., and Cao, D.S. (2017). ADME properties evaluation in drug discovery: Prediction of plasma protein binding using NSGA-II combining PLS and consensus modeling. Chemometr. Intell. Lab. Syst. *170*, 84–95.
37. Alsenan, S., Al-Turaiki, I., and Hafez, A. (2021). A deep learning approach to predict blood-brain barrier permeability. PeerJ. Comput. Sci. *7*, e515.
38. Wu, Z., Jiang, D., Wang, J., Hsieh, C.Y., Cao, D., and Hou, T. (2021). Mining Toxicity Information from Large Amounts of Toxicity Data. J. Med. Chem. *64*, 6924–6936.
39. Lombardo, F., Berellini, G., and Obach, R.S. (2018). Trend Analysis of a Database of Intravenous Pharmacokinetic Parameters in Humans for 1352 Drug Compounds. Drug Metab. Dispos. *46*, 1466–1477.
40. Wang, J.-B., Cao, D.-S., Zhu, M.-F., Yun, Y.-H., Xiao, N., and Liang, Y.-Z. (2015). In silico evaluation of logD7.4 and comparison with other prediction methods. J. Chemometr. *29*, 389–398.
41. Delaney, J.S. (2004). ESOL: Estimating Aqueous Solubility Directly from Molecular

Structure. J. Chem. Inf. Comput. Sci. *44*, 1000–1005.

42. Tang, J., Chang, Y., Aggarwal, C., and Liu, H. (2016). A survey of signed network mining in social media. ACM Comput. Surv. *49*, 1–37.

43. Ford, L.R., and Fulkerson, D.R. (1956). Maximal flow through a network. Can. J. Math. *8*, 399–404.

44. Nguyen, T., Le, H., Quinn, T.P., Nguyen, T., Le, T.D., and Venkatesh, S. (2021). GraphDTA: predicting drug–target binding affinity with graph neural networks. Bioinformatics *37*, 1140–1147.

45. Kipf, T.N., and Welling, M. (2016). Semi-supervised classification with graph convolutional networks. Preprint at arXiv. https://doi.org/10.48550/arXiv.1609.02907.

46. Tang, H., Liu, J., Zhao, M., and Gong, X. (2020). Progressive layered extraction (ple): A novel multi-task learning (mtl) model for personalized recommendations. In 14th ACM Conference on Recommender Systems, pp. 269–278.

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited data		
source code	This paper	https://github.com/dubingxue/MTGL-ADMET
Software and algorithms		
Python (version 3.8)	Python software	https://www.python.org/

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources should be directed to the lead contact, Jian-Yu Shi (jianyushi@nwpu.edu.cn).

Materials availability

This study did not generate any new unique materials.

Data and code availability

- Deidentified final results supporting this study are available for research purposes upon reasonable written request to the corresponding author. Access to such data is available from the date of publication and requires a Data Access Agreement, which is examined and approved by the ethics committees who approved this research.
- The codes and the data underlying this article are freely available at https://github.com/dubingxue/MTGL-ADMET.
- Any additional information required to reanalyze the data reported in this paper is available from the lead contact upon request.

METHOD DETAILS

Dataset and setup

To evaluate our MTGL-ADMET, we built a dataset covering 24 endpoints (18 for classification and 6 for regression) from 8 publications. The dataset contains five Absorption,[34,35] two Distribution,[36,37] five Metabolism,[38] two Excretion,[39] eight Toxicity[38] and two ADMET-related Physicochemical properties.[40,41] (See Supplementary ADMET Dataset). There are 43,291 drug-like compounds across 24 endpoint tasks in total, including 28,153 compounds in classification tasks and 16,545 in regression tasks, where a compound may have one or more endpoint labels (Figure S2). More details about the dataset spilt of ADMET endpoints can be found in Table S2.

Each node of the input drug/compound was initially represented by a 40-dimensional (40-d) binary atom feature vector, including atom symbol (16-d), degree (7-d), formal charge (1-d), radical electrons (1-d), hybridization (6-d), aromaticity (1-d), hydrogens (5-d), chirality (1-d) and chirality type (2-d), as suggested in[38]. Furthermore, the two-layer ResGCN encodes each compound into a 64-dimensional embedding vector. After that, through parallel attention blocks on atoms of each task, each compound is represented as a 64-dimensional embedding vector. In the meanwhile, the input feature of FC in the gating network is the task-shared average molecular representation, which is also a 64-dimensional embedding vector and the output feature is a 2-dimensional embedding vector. In the *n*-th task predictor module, of which the input layer, the two-hidden layer, and the output layer contain 64, 128, 128, and 1 neurons respectively. In addition, we adopted an empirical setting for the parameters in the training, which assigns the batch size of 128, the epoch number of 200, the learning rate of 0.001, and selects Adam as the optimizer. Furthermore, MTGL-ADMET is implemented in Python 3.8 and PyTorch 1.8.0, along with functions from dgl 0.4.3, Scikit-learn 0.24.2, Numpy 1.20.2, Pandas 1.20.0 and RDKit 2019.09.3.

Problem formulation

Given a set of n ADMET endpoint tasks $\mathcal{T} = \{t_1, t_2, \ldots, t_n\}$, we construct n multi-task neural networks (MTNN) $\mathcal{M} = \{m_1, m_2, \ldots, m_n\}$ accordingly. Each MTNN m_k adopts a paradigm, 'one primary, multiple auxiliaries', which enhances the primary task by its auxiliary tasks (shown in Figure 1D). Suppose that a specific primary task t_k and its auxiliary tasks $\mathcal{T}_k = \{t_1^{(k)}, t_2^{(k)} \ldots\} \subseteq \mathcal{T}$, where t_k's auxiliary task $t_i^{(k)} \in \mathcal{T}$. One of our goals is to determine $\mathcal{T}_k$ among $\mathcal{T}$ for t_k for further training m_k.

Given M compounds $\{c_i, i = 1, 2, \ldots, M\}$ and their ADMET properties $y = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_M\}$ w.r.t. $\mathcal{T}$, where $\mathbf{y}_i \in R^n$, $\mathbf{y}_i(j) \in \{1, 0, -\}$ or $\mathbf{y}_i(j) \in \mathcal{R}$, $j \in \{1, 2, \ldots n\}$, where '$-$' has no property measured. The former type of $\mathbf{y}_i(j)$ indicates the binary classification problem, which determines whether a molecule has an ADMET endpoint of interest or not. The latter accounts for the regression problem, which reflects how well its endpoint is. For example, *Phenobarbital*, an anticonvulsant drug, has significant Hepatotoxicity (i.e., $\mathbf{y}_i(j) = 1$) but none of the four kinds of Cardiotoxicities (i.e., $\mathbf{y}_i(j) = 0$) while exhibiting the value 3.16 of Lethal Dose 50 % (LD50), which is a measure of Acute oral toxicity (i.e., $\mathbf{y}_i(j) = 3.16$), but has no other measured property (i.e., $\mathbf{y}_i(j) = ' - '$). One of our goals is to predict n ADMET properties $\mathbf{y}_x$ of a new coming compound c_x by the well-trained MTNNs $\{m_k\}$.

Overview of framework

As shown in Figure 2, our MTGL-ADMET model consists of two stages. One is the adaptive auxiliary tasks selection (Figure 2A) while another is a novel multi-task learning model for both a primary task and its selected auxiliaries (Figure 2B). The adaptive auxiliary task selection contains three steps. **Step I**: We leverage the differences between the performances of individual tasks and those of pairwise tasks to calculate inter-task associations. **Step II**: we perform a preliminary selection of auxiliary tasks for each task by the status theory.[42] **Step III**: we run a further selection of optimal auxiliary tasks by the maximum flow calculation.[43] The novel multi-task learning model is designed for each task group under the paradigm 'one primary task, multiple auxiliary tasks'. It contains four components, including a task-shared atom embedding module, a task-specific molecular embedding module, a primary task-centered gating module, and a multi-task predictor.

Adaptive auxiliary task selection

Task association network construction

To investigate how well a task can boost another one, we measure the differences between the performances of individual ADMET tasks and their corresponding pairwise tasks. These differences are regarded as inter-task associations. Suppose that t_w and t_k are two tasks, $\mathcal{S}_w$ and $\mathcal{S}_k$ are two single-task learning models accounting for t_w and t_k individually, and $\mathcal{D}_{w,k}$ is a dual-task learning model training the tasks simultaneously (Figure S3). These learning models contain a two-layer ResGCN, an attention block, and fully connected neural networks. We design an index $\widehat{Z}_{w\to k}$ to reflect the fluence of t_w on t_k as follows,

$$\widehat{Z}_{w\to k} = Z_{k|w}^{(d)} - Z_k^{(s)}, \tag{Equation 1}$$

where $Z_k^{(s)}$ is the performance of t_k given by $\mathcal{S}_k$, and $Z_{k|w}^{(d)}$ is the performance of t_k given by $\mathcal{D}_{w,k}$. Similarly, we can define $\widehat{Z}_{k\to w} = (Z_{w|k}^{(d)} - Z_w^{(s)})$ to reflect how t_k fluences t_w, where $Z_{w|k}^{(d)}$ is the performance of t_w given by $\mathcal{D}_{w,k}$. In general, $\widehat{Z}_{t_w\to t_k} \neq \widehat{Z}_{t_k\to t_w}$, since the influence between two tasks is asymmetric. Moreover, such an influence could be either positive or negative. If $\widehat{Z}_{w\to k} > 0$, t_w boosts t_k; otherwise, t_w depresses t_k. Finally, regarding these differences as inter-task associations among all the tasks $\mathcal{T}$, we organize them into a bi-directed and signed network, where nodes are ADMET tasks and edges are inter-task associations. The adjacent matrix of the task association network is illustrated in Figure S4. Based on the task association network, a preliminary selection of primary task-auxiliary tasks shall be performed by leveraging complex network analysis.

Preliminary auxiliary selection by status theory

Inspired by status theory in social network,[42] we perform a preliminary selection of auxiliaries for each task. The status theory states the rule of 'the person respected by me should have higher status than me' where social status represents the prestige of nodes (persons) in a social network. Analogously, the status represents the ranking of tasks in the AMDET task association network. Holding the paradigm 'one primary, multiple auxiliaries', we attempt to construct a primary task-specific pool containing a high-ranking primary and its low-ranking auxiliaries. The status theory helps find the lower-ranking auxiliaries. In terms of $\widehat{Z}_{w\to k}$, we empirically selected ≤ 5 auxiliary tasks in descending order with considering the high complexity of status theory-satisfied triads in the case of multiple auxiliaries.

Given a primary task t_k, our task is to select a set of auxiliary tasks $\overline{T_k} = \{t_i\} \subset \mathcal{T}$, $i \neq k$ and $i = 1,2\ldots,n$. We consider two tasks $t_w \in \overline{T_k}$ and $t_z \in \overline{T_k}$ accompanying with t_k to form a triad. There are two types of triads in directed networks, which correspond to acyclic and cyclic triads (Figure 2A). If the triad is acyclic, then the triad satisfies the status theory.[42] In this case, the status theory implies that both t_w and t_z have higher status than t_k if there are positive associations from them to t_k or negative associations from t_k to them respectively. However, we cannot determine the result directly because the inter-task associations are signed and bi-directed. Thus, we first turn to bi-directed associations among t_k, t_w and t_z into mono-directed associations as follows,

$$e_{i,j} = \widehat{Z}_{i\to j} - \widehat{Z}_{j\to i}, i,j \in \{w,z,k\}, \tag{Equation 2}$$

where $e_{i,j}$ is the mono-directed association between t_i and t_j. Then, we reverse the directions of all negative $e_{i,j}$ and flip their signs to positive (i.e., $e_{j,i} = -e_{i,j}$, if $e_{i,j} < 0$). So far, the task association network can be treated as a pure directed network when omitting positive signs. Sequentially, it can examine whether a given triad satisfies the status theory (acyclic) or not (cyclic). In this manner, acyclic triads w.r.t. the primary task t_k could be found to expand the auxiliary group.

Refined auxiliary selection by maximum flow

Holding the paradigm 'one primary, multiple auxiliaries', we suppose that t_k is the primary task, and two other tasks t_w and t_z are its auxiliaries. Let $M_{k|w,z}$ be a multi-task learning model accounting for them, $Z_{k|w,z}^{(m)}$ be the performance of t_k obtained by the model. The status theory guarantees that mono-directed associations flow from all the auxiliaries to the primary task, but the corresponding bi-directed inter-task associations could be positive or negative. In details, each mono-directional association to t_k ($\widehat{Z}_{i\to k} - \widehat{Z}_{k\to i} > 0$) accounts for three scenarios of bi-directional inter-task associations, including (1) ($\widehat{Z}_{i\to k} > 0$ & $\widehat{Z}_{k\to i} < 0$), (2) ($\widehat{Z}_{i\to k} > 0$ & $\widehat{Z}_{k\to i} > 0$), (3) ($\widehat{Z}_{i\to k} < 0$ & $\widehat{Z}_{k\to i} < 0$), where $i \in \{w,z\}$.

Naturally, we desire positive inter-task associations from the auxiliaries to the primary, which reflects that the auxiliary task t_i boosts the primary task t_k. Thus, we select the first two scenarios with the top priority to determine two auxiliaries of a specific primary task t_k.

More importantly, since we also expect that $Z^{(m)}_{k|w,z} \geq Max(Z^{(d)}_{k|w}, Z^{(d)}_{k|z})$ at the same time, we calculate the maximum flow for the triad based on Ford-Fulkerson algorithm,[43] where t_k is the sink node, one of its auxiliaries is the source node (e.g., t_i) and another is the intermedia node (e.g., t_j), $i,j \in \{w,z\}$. We directly regard mono-directed associations $\{e_{i,j}\}$ as the flux between two tasks. Specifically, $e_{i,k}$ is the flux from t_i to t_k, and $e_{j,k}$ is the flux from t_j to t_k. Thus, we define the maximum input flux (e.g., $f^{\max}_{k}(i,j)$) of t_k given the source t_i and the sink t_j as follows,

$$f^{\max}_{k|i,j} = e_{i,k} + \min\left(e_{i,j}, e_{j,k}\right). \qquad \text{(Equation 3)}$$

In the case that t_k has more than 2 auxiliaries, we apply the Ford-Fulkerson algorithm[43] to calculate the maximum flux. Finally, according to the value of $f^{\max}_{k}(i,j)$ of multiple combinations from 2 to 5 auxiliaries in descending order, we determine the auxiliaries w.r.t. t_k.

Multi-task graph learning model

The multi-task graph learning model contains a task-shared atom embedding module, a task-specific molecular embedding module, a primary task-centered gating module, and a multi-task predictor module (Figure 2B).

Task-shared atom embedding module

By adopting two-layer residual GCN (ResGCN), the task-shared atom embedding module learns atom embedding representation by aggregating neighboring atom features on molecule graphs, which are shared by both the primary task and its auxiliaries. According to chemical structure, each compound c is represented as a molecule graph $\mathcal{G} = (\mathcal{V},\mathcal{E})$, where $\mathcal{V}$ is a set of atoms and $\mathcal{E}$ is a set of chemical bonds. Let $\mathbf{A} \in R^{N\times N}(N = |\mathcal{V}|)$ be its adjacency matrix, in which $a_{ij} = 1$ indicates the occurring bond ($e_{ij} \in \mathcal{E}$) between atom $i(v_i)$ and atom $j(v_j)$, and $a_{ij} = 0$ indicates no bond. Here, each node v_i (atom) is initially represented by a q-dimensional binary feature vector $\mathbf{h}_i \in R^q$. As suggested in[44], the initial node features typically include the atom symbol, the number of adjacent atoms, the number of adjacent hydrogens, the implicit value of the atom, and the atom occurrence in an aromatic structure. For each atom v_i in the molecule graph $\mathcal{G}$ of compound c, each layer of the ResGCN updates its features $\mathbf{h}^c_i \in R^{1\times d}$ by aggregating the embeddings of its neighboring atoms and adding a residual connection from the previous layer.[45] The aggregation update rule for atom embedding on the kth layer in the matrix form is defined as follows:

$$\mathbf{H}^{(k+1)} = \sigma\left(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{H}^{(k)}\mathbf{W}^{(k)}\right) + \sigma\left(\mathbf{H}^{(k)}\right), k = 0,1, \qquad \text{(Equation 4)}$$

where $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}_\mathbf{N}$, $\mathbf{I}_\mathbf{N}$ is the identity matrix, $\mathbf{D} \in R^{N\times N}$ is the degree matrix, in which diagonal elements are the degrees of each vertex and $\tilde{\mathbf{D}}_{ii} = \sum_j \tilde{\mathbf{A}}_{ij}$, $\mathbf{W}^{(k)}$ is the layer-wise trainable weight matrix, and $\sigma(\cdot)$ denotes an activation function.

Note that the atom embedding vector $\mathbf{h}^c_i$ (i.e., each row of $\mathbf{H}$) is shared by all the tasks. In classical single-task learning, the molecular embedding $\overline{\mathbf{h}}$ can be obtained by a popular readout, which is simply generated by averaging the task-shared atom embeddings of compound c. In contrast, the current multi-task learning requires task-specific molecular embedding, which shall be obtained in the next section.

Task-specific molecular embedding module

The task-specific molecular embedding module leverages parallel attention layers to learn task-specific molecular representations ($\mathbf{h}_k$,$\mathbf{h}^{(k)}_i$) of compound c. The idea seems like a multi-head attention (i.e., a concatenation of parallel attentions). However, considering a different manner, we don't concatenate atom embeddings weighted by parallel task-specific attention layers, but take the weights on the same set of task-shared atom embeddings to differentiate atom embeddings w.r.t. tasks. Then, the task-specific molecular representations can be obtained by a classical readout on them.

Suppose that compound c contains N atoms, of which each is encoded into p-dimensional vectors $\mathbf{h}^c_i \in \mathrm{R}^{p\times 1}$ by the task-shared atom embedding module. With regard to a specific task t_z, its attention weights are implemented by a forward-feed neural network as follows:

$$\mathbf{a}_z = Sigmoid(\mathbf{W}_z \cdot \mathbf{H} + \mathbf{b}_z), \sum\nolimits_{i=1}^{N} a_z(i) = 1, \qquad \text{(Equation 5)}$$

where $\mathbf{W}_z \in \mathrm{R}^{1\times p}$ is the learnable weight matrix and $\mathbf{b}_z \in \mathrm{R}^{1\times N}$ is the bias vector, $\mathbf{H} \in \mathrm{R}^{p\times N}$ is the task-shared atom embedding matrix (stacked by $\{\mathbf{h}^c_i \in \mathrm{R}^{p\times 1}\}$ column by column, a transposed form of that in the last section), $\mathbf{a}_z \in \mathrm{R}^{1\times N}$ is the atom weight vector, of which each element $a_z(i)$ accounts for the i-th atom in the compound. Thus, the task-specific molecular embedding w.r.t. t_z (denoted as $\mathbf{h}_z$) can be calculated in the following form:

$$\mathbf{h}_z = \sum\nolimits_{i=1}^{N} a_z(i)\mathbf{h}^c_i. \qquad \text{(Equation 6)}$$

More importantly, task-specific atom weights facilitate finding crucial substructures w.r.t. tasks, where two bonding atoms are regarded as a crucial structure fragment if both of them have high weights.

Primary task-centered gating module

Under the paradigm ‘one primary, multiple auxiliaries’, the primary task-centered gating module learns how each auxiliary task contributes to the primary task and how these contributions are combined by a set of gating networks. Inspired by,[46] each gating network G is simply composed of a single-layer feed-forward network and a Sigmoid activation function. It takes the task-shared average molecular embedding $\overline{\mathbf{h}}$ of compound c as the input and outputs two scalar weights, of which one is for the primary task and another is for an auxiliary. The weighted embedding of the auxiliaries with regard to the primary task is taken as its contribution to the primary task. Sequentially, all the contributed embeddings of the primary task are summed up as their final embeddings (Figure 2B).

Let G_i be a gating network containing a fully connected layer FC_G^i, which accounts for each primary-center task pair be $(t_k, t_i^{(k)})$ where t_k is the primary task, $t_i^{(k)} \in \mathcal{T}$ is its i-th auxiliary, and $i = 1, 2, \ldots, \left|\{t_i^{(k)}\}\right|$. With regard to t_k and $t_i^{(k)}$, we suppose that $w_k^{(i)}, w_i^{(k)}$ are their weights and $\mathbf{h}_k, \mathbf{h}_i^{(k)}$ are their task-specific embeddings respectively. In addition, let $\overline{\mathbf{h}} \in \mathrm{R}^{p \times 1}$ be the task-shared molecular embedding, which is simply generated by averaging the task-shared atom embeddings of compound c (i.e., a popular readout). Formally, the weights of the primary task and its auxiliary $t_i^{(k)}$ are defined by a neural network as follows,

$$\left[w_k^{(i)}, w_i^{(k)}\right] = Softmax\left(FC_G^i(\overline{\mathbf{h}})\right), \qquad \text{(Equation 7)}$$

Furthermore, with the contribution of $t_i^{(k)}$ to t_k, its embedding $\mathbf{h}_{i \to k}$ is defined as

$$\mathbf{h}_{i \to k} = w_k^{(i)} \mathbf{h}_k + w_i^{(k)} \mathbf{h}_i^{(k)}, w_k^{(i)} + w_i^{(k)} = 1. \qquad \text{(Equation 8)}$$

Thus, the final embedding of the primary task passing through all the gating networks $\{G_i\}$ is as follows:

$$\mathbf{h}_k^* = \sum_{i=1} \mathbf{h}_{i \to k}. \qquad \text{(Equation 9)}$$

Once the primary task-centered gating module is done, $\mathbf{h}_k^*$ and $\{\mathbf{h}_i^*\}$ are passed into task-specific towers (e.g., implemented by fully-connected neural networks) to predict task labels.

Multi-task predictor

Either the primary task or its auxiliaries have individual predictors, which contain task-specific fully-connected neural networks to learn better task-specific nonlinear representation. We implement these neural networks (NNs) with the same architecture, which contains an input layer, a hidden layer, and an output layer. The predictor concerning the primary task t_k maps the contributed embeddings $\mathbf{h}_k^*$ into the predicted primary task label $\widehat{y}_k$ (i.e., $\widehat{y}_k = NN_k(\mathbf{h}_k^*)$, while those predictors accounting for the auxiliaries $\{t_i^{(k)}\}$ maps $\{\mathbf{h}_i^{(k)}\}$ into corresponding auxiliary task labels (i. e., $\{\widehat{y}_i^{(k)} = NN_i^{(k)}(\mathbf{h}_i^{(k)})\}$). Furthermore, considering these tasks are of classification or regression, we use the Cross-Entropy loss for classification tasks and the Mean Squared Error loss for regression tasks when training multiple tasks together. When training MTGL-ADMET, the loss of multi-task learning is defined as follows,

$$loss = \sum_{c=1}^{C} \sum_{n=1}^{M_c} \left(- \left[p_c y_c \cdot \log \sigma(\widehat{y}_c) + (1 - y_c) \cdot \log(1 - \sigma(\widehat{y}_c)) \right] \right) + \sum_{r=1}^{R} \sum_{n=1}^{M_r} (\widehat{y}_r - y_r)^2, \qquad \text{(Equation 10)}$$

where y_c and $\widehat{y}_c$ are the true label and the predictor value of compound c_n w.r.t. classification task t_c respectively, y_r is the true property value of c_n w.r.t. regression task t_r, $\widehat{y}_r$ is the corresponding predicted value, C is the number of classification tasks and M_c is the number of compounds in classification tasks. R is the number of regression tasks and M_r is the number of compounds in regression tasks. Inspired by,[38] to alleviate the imbalance of positive and negative samples in classification tasks, we utilize a weight p_c in the loss function, which is defined as the ratio of the number of negative samples to that of positive samples w.r.t. classification task t_c.