A. Algorithm

Algorithm 1: The overall process of MDP

Input: Molecular dataset $\mathcal{D}$, number of tasks N_c, number of sampled examples for each task η, number of molecular labeling function z

Derive multiple distance matrices $\{\mathbf{S}^1, \cdots, \mathbf{S}^z\}$ by measuring the similarity between any two molecular graphs

Utilize $\mathrm{Cluster}(\mathcal{D}, \{\mathbf{S}^1, \cdots, \mathbf{S}^z\})$ to sample $\eta \times N_c$ molecular graphs for generating labeled set $\mathcal{D}_L$

for $e = 1$ to z **do**

  Utilize labeled set $\mathcal{D}_L$ to generate Λ^e using Eq. (3), Eq. (4) and Eq. (5)

repeat

  Generate $\widetilde{\mathbf{Y}}$ using Eq. (6)

  Optimize the parameters of the label synchronizer G_ϵ using Eq. (8) via $\mathcal{L}_{\mathrm{adap}}\left(\widetilde{\mathbf{Y}}, \mathbf{Y}\right)$

  Optimize the parameters of the molecular property classifier D_θ using Eq. (7) via $\mathcal{L}_{\mathrm{s}}\left(\bar{\mathbf{Y}}, \mathbf{Y}\right)$ and $\mathcal{L}_{\mathrm{u}}\left(\bar{\mathbf{Y}}, \widetilde{\mathbf{Y}}\right)$

until reach the maximum iteration;

return: Model parameters

B. The Statistics of Datasets

The statistics of each dataset are summarized in Table 3. For all molecule datasets, we utilize the scaffold splitting procedure [35] to split molecules into training set (0.8), validation set (0.1) and testing set (0.1) according to their molecular substructure. Under the setting of weakly supervised learning, we train our proposed method and baselines with the same number of labeled data sampled from the training set. The remaining data in the training set serves as the unlabeled training data. The number of labeled data of each dataset is equal to $\eta \times N_c$. N_c represents the number of binary classification tasks of the dataset and $\eta \in \{10, 20\}$. Due to the graph classification task, we use AUC [47] as the evaluation metric following the previous molecular graph tasks. Validation set is used for early stopping and evaluate the AUC on testing set.

- **BBBP [29].** The dataset consists of binary labels of blood-brain barrier penetration (membrane permeability).
- **Tox21 [12].** The dataset contains the qualitative toxicity measurements on 12 biological targets.
- **ToxCast [40].** The dataset includes a large library of compounds based on over 600 in vitro high-throughput screenings.
- **SIDER [19].** The database of marketed drugs and adverse drug reactions of FDA approved drugs, divided into 27 system organ classes.
- **ClinTox [31].** The dataset that compares drugs approved by the FDA and drugs that have failed clinical trials for toxicity reasons.
- **MUV [10].** The subset of PubChem BioAssay contains 17 challenging tasks for around 90 thousand compounds and is specifically designed for validation of virtual screening techniques.
- **HIV [2].** The dataset contains 41,127 compounds with binary labels indicating whether the compound is active or inactive against the Human Immunodeficiency Virus (HIV).
- **BACE [45].** The dataset contains the quantitative and qualitative (binary label) binding results for a set of inhibitors of BACE-1.

Table 3. Statistics of the datasets.

Dataset	Graphs	Tasks	Avg./Max Nodes	Avg./Max Edges
HIV	41,127	1	25.5 / 222	54.9 / 502
MUV	93,087	17	24.2 / 46	52.6 / 104
ToxCast	8,576	617	18.8 / 124	38.5 / 268
Tox21	7,831	12	18.6 / 132	38.6 / 290
BBBP	2,039	1	24.1 / 132	51.9 / 290
BACE	1,513	1	34.1 / 97	73.7 / 202
ClinTox	1,477	2	26.2 / 136	55.8 / 286
SIDER	1,427	27	33.6 / 492	70.7 / 1010

C. Implementation

We use a 5-layer Graph Isomorphism Network (GIN) [54] as the representation model, due to its expressive architecture for prediction tasks on graphs. We set the embedding dimension of a GIN layer is 300 and a mean pooling layer for the readout function. The Adam is selected as the optimizer with initial learning rate of 1×10^{-3}. The batch size is 32 across all scenarios. We set dropout ratio as 0.5 for GIN layers and default settings for baselines. The running epoch is fixed to 100. We implement the proposed model using Pytorch [32] and run it on Titan RTX GPUs. We run each experiment 5 times and report the mean values with standard deviation.

D. Additional Experiments

D.1. Additional Diversity Measures

In Figure 5, we present a heatmap showing the pairwise diversity of BBBP.

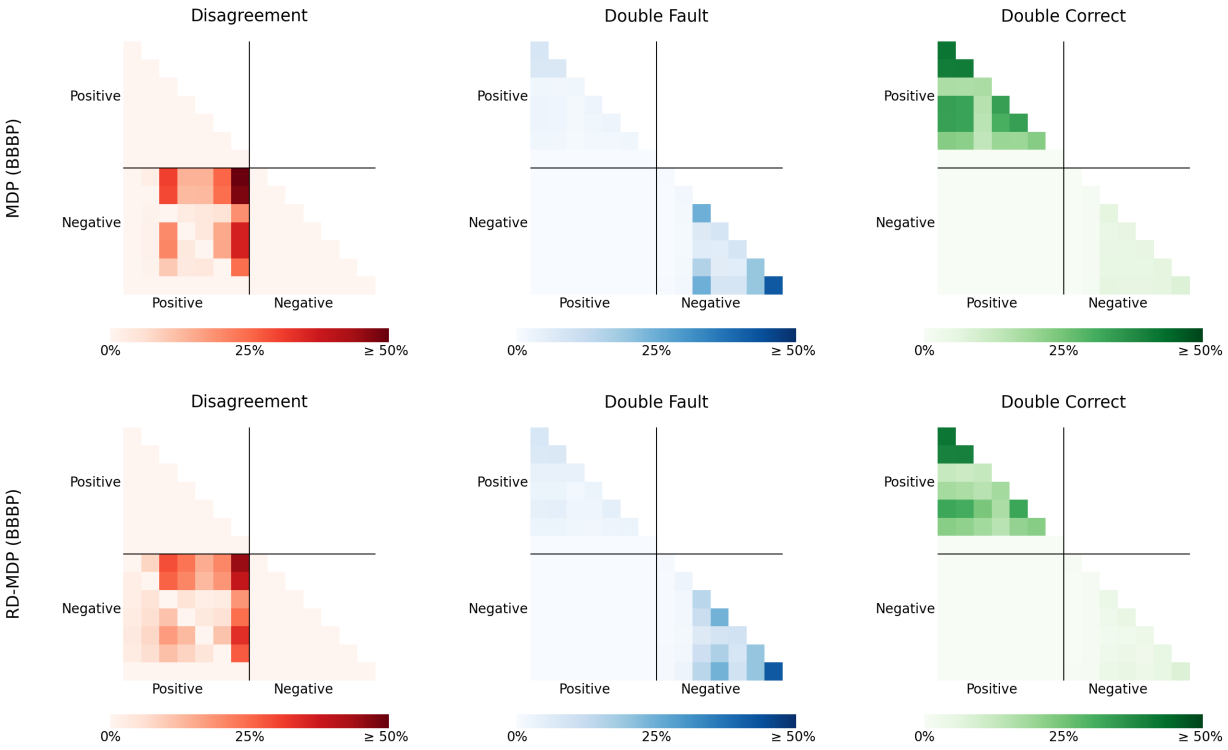

Figure 5. Diversity metrics among labeling functions on BBBP: Disagreement (left), Double Fault (center), Double Correct (right). Each cell in the matrix denotes the coverage of training instances, marked by color intensity, where both molecular labeling functions i and j label the same example.

D.2. Additional Experiment with Fewer Labeled Examples

In Figure 6, we plot test auc curves for the other datasets in the molecular property prediction experiments.

D.3. Influence of the Labeling Ratio

Q: Whether our proposed framework MDP delivers the superior accuracy compared with baseline models on the different labeling ratio? Yes, we examine it below.

▷ **Comparison with baseline models**. We vary the labeling ratio of training data on BBBP, BACE, SIDER, ClinTox and Tox21 to evaluate the effectiveness of MDP and collect the comparison results in Table 4. Different labeling ratio influences the number of labeled molecule data sampled from the training set. As seen from the Table 4, ❶ *the performance of all models generally improves when the number of labeling ratio increases*, which illustrates that utilizing more available labeled data can efficiently boost the performance. ❷ *Compared with all baselines, MDP consistently achieves the best results across all datasets.* The experimental results demonstrate that increasing the amount of manually labeled molecule data does not compromise the performance of MDP when compared to other baselines.

D.4. Labeling Function Type Ablation.

Q: Whether it is beneficial to generate multiple weak supervision signals combing various domain knowledge? We test the predictive performance using the four molecule datasets to identify the impact of different types of labeling functions:

- **Graph Kernel (GK)**: Assigning pseudo-labels for unlabeled graphs according to the similarity computed by graph kernel method.
- **Molecular Fingerprint (MF)**: Assigning pseudo-labels for unlabeled graphs according to the molecular fingerprint similarity.
- **Structure-based (SB)**: Assigning pseudo-labels for unlabeled graphs according to the difference of statistical structural properties.

We show the experimental results in Table 5. We see that ❶ *combining different types of labeling functions can always outperform using only one type of labeling function.* The

Table 4. Test AUC performance of different methods on five molecular prediction benchmarks with various amounts of labeled data. The best results are in bold and the second best results are underlined.

Method	BBBP		BACE		SIDER		ClinTox		Tox21	
	1%	5%	1%	5%	1%	5%	1%	5%	1%	5%
GCN	0.503±0.019	0.582±0.015	0.453±0.019	0.547±0.007	0.498±0.003	0.517±0.004	0.485±0.006	0.526±0.007	0.544±0.012	0.609±0.005
GAT	0.506±0.005	0.592±0.009	0.460±0.017	0.539±0.021	0.523±0.001	0.544±0.006	0.489±0.007	0.514±0.030	0.553±0.004	0.616±0.006
GraphSAGE	0.511±0.009	0.566±0.013	0.464±0.009	0.531±0.009	0.520±0.003	0.552±0.006	0.473±0.022	0.541±0.008	0.539±0.002	0.607±0.008
GIN	0.514±0.007	0.557±0.025	0.499±0.027	0.576±0.032	0.505±0.005	0.559±0.006	0.457±0.015	0.503±0.021	0.554±0.008	0.613±0.017
Pseudo-labeling	0.547±0.013	0.596±0.031	0.478±0.011	0.601±0.014	0.514±0.007	0.566±0.018	0.519±0.024	0.584±0.030	0.593±0.015	0.627±0.020
Self-training	0.539±0.014	0.602±0.023	0.527±0.010	0.619±0.019	0.508±0.006	0.553±0.010	0.528±0.018	0.594±0.019	0.588±0.017	0.637±0.024
infomax	0.540±0.037	0.603±0.009	0.515±0.015	0.574±0.044	0.529±0.006	0.530±0.004	0.542±0.025	0.576±0.034	0.570±0.008	0.629±0.004
contextpred	0.443±0.070	0.600±0.018	0.544±0.048	0.577±0.017	0.513±0.003	0.530±0.003	0.473±0.030	0.502±0.009	0.585±0.037	0.644±0.006
masking	0.464±0.006	0.595±0.006	0.554±0.015	0.608±0.026	0.488±0.012	0.527±0.007	0.532±0.016	0.539±0.023	0.592±0.009	0.653±0.004
edgepred	0.573±0.019	0.580±0.009	0.566±0.032	0.603±0.021	0.497±0.010	0.529±0.007	0.510±0.041	0.530±0.014	0.579±0.014	0.647±0.007
GraphLog	0.522±0.020	0.580±0.011	0.470±0.015	0.612±0.021	0.524±0.007	0.567±0.005	0.516±0.019	0.578±0.036	0.589±0.007	0.640±0.005
perturb_edge	0.564±0.013	0.571±0.017	0.514±0.030	0.594±0.014	0.520±0.008	0.542±0.007	0.561±0.006	0.556±0.037	0.597±0.011	0.625±0.003
drop_node	0.525±0.015	0.564±0.013	0.526±0.015	0.568±0.017	0.510±0.012	0.497±0.008	0.530±0.022	0.525±0.020	0.572±0.009	0.608±0.007
subgraph	0.539±0.018	0.525±0.023	0.535±0.010	0.582±0.019	0.498±0.009	0.520±0.006	0.445±0.015	0.505±0.043	0.589±0.009	0.652±0.013
Our	**0.585±0.027**	**0.619±0.022**	**0.602±0.021**	**0.636±0.008**	**0.532±0.032**	**0.570±0.023**	**0.573±0.030**	**0.615±0.025**	**0.649±0.009**	**0.671±0.008**

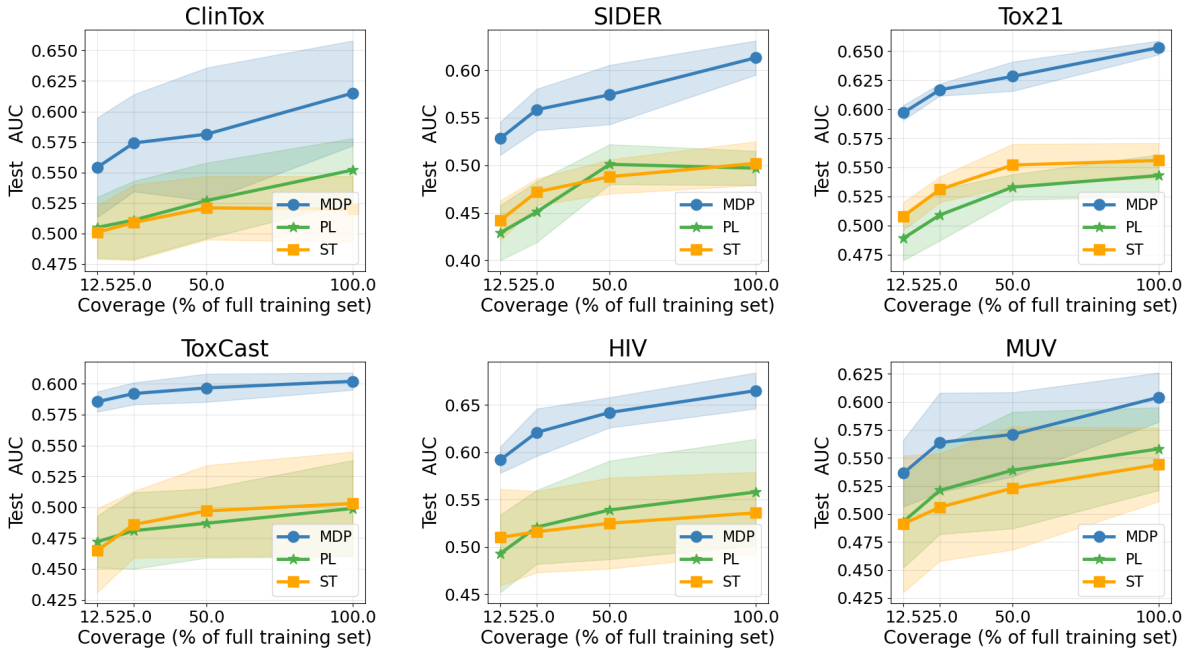

Figure 6. Plot of increasing labeling sample coverage (x-axis), v.s., accuracy (y-axis) using test curves.

reason behind this is that different types of labeling functions complement each other, effectively generating high-quality pseudo-labels without requiring human labeling effort.

D.5. Adapting MDP to Different Backbones.

Q: Whether MDP can still perform well when changing the backbones? We equip several state-of-the-art GNN models, i.e., GCN, GAT, GraphSAGE and GIN with MDP on five datasets (BACE, BBBP, ClinTox, SIDER, Tox21) for illustrating that MDP can be adapted to different back-

Table 5. The effect of different labeling function types on molecular property prediction tasks.

LF Type	BACE	BBBP	ClinTox	SIDER
GK	0.576±0.026	0.565±0.035	0.513±0.072	0.526±0.018
MF	0.618±0.022	0.581±0.035	0.584±0.029	0.577±0.008
SB	0.575±0.035	0.526±0.018	0.456±0.042	0.501±0.028
GK+MF	0.625±0.028	0.604±0.021	0.611±0.019	0.609±0.014
GK+SB	0.608±0.039	0.589±0.037	0.577±0.054	0.523±0.025
MF+SB	0.615±0.034	0.597±0.025	0.591±0.045	0.599±0.027
GK+MF+SB	0.632±0.023	0.618±0.011	0.615±0.043	0.613±0.018

Table 6. Test ROC-AUC performance of MDP on five molecular prediction benchmarks with different backbones.

Methods	BACE	BBBP	ClinTox	SIDER	Tox21
GCN	0.488	0.509	0.513	0.455	0.525
MDP_{GCN}	0.616	0.621	0.607	0.582	0.631
GAT	0.532	0.519	0.514	0.450	0.533
MDP_{GAT}	0.611	0.606	0.598	0.587	0.636
GSE	0.504	0.522	0.517	0.440	0.520
MDP_{GSE}	0.609	0.614	0.587	0.591	0.625
GIN	0.497	0.517	0.571	0.433	0.486
MDP_{GIN}	0.632	0.618	0.613	0.615	0.653

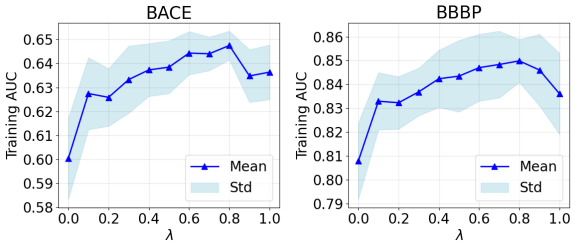

Figure 7. Hyper-parameter sensitivity analysis of MDP.

bones. In Table 6, we make the following observations: MDP adapts well to the four widely-used GNNs and improves them by a large margin on five datasets. For example, the performance that MDP_{GCN} achieves are 0.616, 0.621, 0.607, 0.582, 0.631 in AUC on five datasets, which are 0.128, 0.112, 0.094, 0.127, 0.106 higher than the GCN model, respectively. Experimental results indicate that MDP does not rely on any specific architecture, and it serves as an effective plug-in module for different GNN models.

D.6. Sensitivity Analysis

Q: How is the sensitivity of MDP to the tuning parameter λ? We examine the sensitivity of the hyper-parameter λ, which controls the weight between labeled and unlabeled loss. Figure 7 illustrates the performance change under different value of λ of MDP with BACE and BBBP datasets. Notably, the performance on the BACE dataset shows continuous improvement in validation as the parameter λ increases, affirming the efficacy of incorporating unlabeled data. However, excessive emphasis on the unlabeled loss, as indicated by further increases in the parameter λ, can introduce noise into the probabilistic labels, which has a negative impact on the performance of the classification model. Similar trends are observed with the BBBP dataset.