Computers and Chemical Engineering 201 (2025) 109253

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/cace

Exploring data augmentation: Multi-task methods for molecular property prediction

Muhammad bin Javaid [a], Timo Gervens [a], Alexander Mitsos [d,b,c], Martin Grohe [a,d], Jan G. Rittig [b,*]

[a] Lehrstuhl Informatik 7, RWTH Aachen University, Aachen, 52074, Germany
[b] Process Systems Engineering (AVT.SVT), RWTH Aachen University, Aachen, 52074, Germany
[c] Forschungszentrum Jülich GmbH, Institute of Climate and Energy Systems ICE-1 – Energy Systems Engineering, 52425, Jülich, Germany
[d] JARA Center for Simulation and Data Science (CSD), 52056, Aachen, Germany

ARTICLE INFO

Dataset link: https://git.rwth-aachen.de/muhammad.javaid/exploring-data-augmentation

Keywords:
Multi-task learning
Molecular property prediction
Graph neural networks
Missing data
Machine learning

ABSTRACT

The effectiveness of machine learning (ML) for molecular property prediction is often limited by scarce and incomplete experimental datasets. A particular promising approach to facilitate training ML models in low-data regimes is multi-task learning. We investigate how additional molecular data – even potentially sparse or weakly related – can be augmented through multi-task learning to enhance prediction quality. Through controlled experiments on progressively larger subsets of the QM9 dataset [Ruddigkeit et al. (2012), J. Chem. Inf. Model; Ramakrishnan et al. (2014), Sci. Data], we evaluate the conditions under which multi-task learning outperforms single-task models. We extend these insights to a practical real-world dataset of fuel ignition properties that is small and inherently sparse, offering recommendations for augmenting auxiliary data to improve predictive accuracy. This work provides a systematic framework for data augmentation in molecular property prediction, with implications for data-constrained applications.

1. Introduction

Machine learning (ML) has emerged as a powerful approach for molecular property prediction, accelerating the discovery of new materials, pharmaceuticals, and industrial chemicals (Alshehri et al., 2020; Atz et al., 2021; Reiser et al., 2022; Stokes et al., 2020; Alshehri and You, 2022; Koscher et al., 2023). However, ML models require sufficient data for training. In general, in the chemical domain, the amount of data is quite limited because of costly experiments. While carefully-designed ML architectures and training procedures lead to adequate prediction accuracy for some properties of interest (Schweidtmann et al., 2020; Gao et al., 2024; Li et al., 2024), the data is insufficient for others, cf. Korolev et al. (2019) and Pappu and Paige (2020). Finding ways to utilize ML for small property data sets – in the order of a few hundred property data points – is therefore highly relevant to catalyze molecular discovery in low data regimes. ML model training in low data regimes can be facilitated by hybrid modeling and data augmentation (Karniadakis et al., 2021; Shorten and Khoshgoftaar, 2019; Vermeire and Green, 2021).

Hybrid modeling, also referred to as physics-informed ML, builds on the idea of incorporating mechanistic insights about the property of interest into the ML model architecture, see, e.g., overviews by Karniadakis et al. (2021), Masi et al. (2021), Jirasek and Hasse (2023) and Rittig et al. (2023). For example, recent works have combined ML with thermodynamics, leading to increased consistency of the predictions and decreased data demands for training, see, e.g., works by Rosenberger et al. (2022), Felton et al. (2024), Winter et al. (2023a,b), Specht et al. (2024), Rittig et al. (2023), Rittig and Mitsos (2024), Chaparro and Müller (2023) and Chaparro and Müller (2024). However, for many properties of interest, mechanistic insights and first principles are still lacking (Gertig et al., 2020), requiring alternative ways to facilitate model training such as data augmentation.

Data augmentation aims to increase the data that can be used in training. Since additional experimental data for the property of interest can often be collected with impractical effort only, the idea of data augmentation is to *alternate* or *utilize related* readily available data, cf. overview in Shorten and Khoshgoftaar (2019).

Alternating available data is common in domains like computer vision (Shorten and Khoshgoftaar, 2019) and natural language processing (Wei and Zou, 2019). For example, image or text datasets can be increased by rotating images or randomly swapping words in a sentence, respectively. This concept has also been transferred

* Corresponding author.
E-mail address: jan.rittig@rwth-aachen.de (J.G. Rittig).

https://doi.org/10.1016/j.compchemeng.2025.109253
Received 13 February 2025; Received in revised form 14 June 2025; Accepted 18 June 2025
Available online 2 July 2025

to the molecular context, e.g., by masking atoms, bonds, and substructures (Magar et al., 2022), adding noise to molecular descriptors and properties (Cortes-Ciriano and Bender, 2015), and using different SMILES permutations (Bjerrum, 2017; Schwaller et al., 2020; Jiang et al., 2023) and tautomers (Ulrich et al., 2021), hence increasing the size of the dataset for training. Similarly, self-supervised learning applies alternation or modifications to unlabeled molecular data, such as atom or bond masking, so that a ML model can be trained to reconstruct the missing information, thereby learning meaningful molecular representations, cf. works by Zhang et al. (2021), Zang et al. (2023), Wang et al. (2024), Gao et al. (2024) and Zhou et al. (2025). Data augmentation through alternating data, particularly, self-supervised learning, has already been applied to train large molecular ML models, e.g., by Rong et al. (2020), Chithrananda et al. (2020), Méndez-Lucio et al. (2024), Li and Fourches (2020) and Li et al. (2021), We argue that large molecular ML models should additionally use readily available data for a wide spectrum of molecular properties. Thus, we are interested in investigating the use of related data.

Several data augmentation approaches that utilize related data have been applied to molecules. These include multi-fidelity, pre-training/transfer, and multi-task learning (MTL) that increase training data sets with simulated or task-related molecular data, cf. overviews by Vermeire and Green (2021), Nevolianis et al. (2024), Alhamoud et al. (2024) and Qian et al. (2025). Since we envision large molecular ML models to utilize large collections of available molecular property data, thereby enabling to predict a variety of molecular properties, similar to recent works by Beaini et al. (2023) and Klaser et al. (2024), we focus on MTL with task-related data.

MTL (Caruana, 1997; Ruder, 2017; Zhang and Yang, 2021) for molecules poses the idea of training a single model simultaneously on multiple related molecular property prediction tasks. This forces the ML model to learn a shared molecular representation that exploits relations between different properties and can thereby enhance prediction accuracies for the individual tasks. Numerous studies have utilized MTL for predicting molecular properties, frequently reporting accuracy improvements, e.g., Schweidtmann et al. (2020), Beaini et al. (2023), Dahl et al. (2014), Ramsundar et al. (2015), Li et al. (2022), Allenspach et al. (2024), Dey and Ning (2024), Brozos et al. (2024), Yang et al. (2024), Zubatyuk et al. (2019) and Liu et al. (2021). However, MTL in molecular applications comes with several challenges, such as (unknown) property relationships, incomplete and imbalanced datasets, and balancing properties losses during training.

The performance of MTL generally strongly depends on the task relationships. In fact, MTL can lead to worse prediction accuracy than single-task learning (STL), i.e., training individual models for each of the properties, due to a phenomenon called *negative transfer* (Standley et al., 2020). This occurs when the inclusion of auxiliary tasks in an MTL framework inadvertently impairs the performance of the model on a primary task of interest, compared to training a model solely for that primary task. This can happen if tasks are too dissimilar, if the model capacity is insufficient, or if the optimization process leads to shared representations that are counterproductive for certain tasks. Even if the average task performance improves, individual tasks might suffer from negative transfer (Liu et al., 2019). Therefore, many prior works have explored methods to optimize MTL according to the task relations, usually learning the task relations implicitly during training (Liu et al., 2019; Yu et al., 2020). If domain-specific knowledge about task relations is available, they can also be explicitly considered in training. Liu et al. (2022), for example, used an protein-protein interaction graph in the domain of drug discovery to advance MTL.

In addition, MTL requires the combination of multiple, potentially interdependent properties into a single training objective (Sosnin et al., 2019). This gives rise to a significant optimization challenge: ensuring each task contributes effectively to training. A foundational step is to standardize the target properties, which aligns disparate physical scales and prevents tasks with large raw values from initially dominating the loss. However, this static pre-processing does not account for the learning dynamics that emerge during training, where tasks often learn at different rates or generate gradients of mismatched magnitudes. To address this problem, task balancing strategies are employed. These methods, such as normalizing task-specific gradients (Chen et al., 2018) or adaptively weighting properties within the loss function (Biswas et al., 2023), algorithmically manage the influence of each task to promote more stable and equitable learning. Beyond task balancing, broader investigations in MTL have also explored data scale and sparsity, including the impact of adding more data or more tasks (Ramsundar et al., 2015) and missing values (de la Vega de León et al., 2018).

A particular challenge for MTL in the molecular domain is the aforementioned limited data availability. That is, dataset sizes differ between the properties and can be small in the order of tens to hundreds of data points. This leads to incomplete and imbalanced datasets of molecules with multiple properties. While such datasets have been considered in previous works, e.g., in Schweidtmann et al. (2020), Brozos et al. (2024) and Biswas et al. (2023), analyzing MTL for different molecular data availabilities is currently limited.

We explore MTL of molecular properties by considering different data availability scenarios of practical relevance. For this, we first use the QM9 dataset (Ruddigkeit et al., 2012; Ramakrishnan et al., 2014), which contains 12 properties for over 130,000 molecules with up to 9 heavy atoms as calculated by quantum mechanics. Modifying this dataset allows us to evaluate the effect of data augmentation in scenarios of:

- varying molecular dataset sizes, from a few hundreds to hundred thousand of property values;
- incomplete molecular datasets, i.e., containing missing data values for some properties of some molecules;
- varying degrees of correlation between molecular properties;
- availability of related property data for different molecules, i.e., the molecules used for training, the target molecules for which predictions are desired, and molecules beyond.

Secondly, we separately consider a real-world dataset of experimentally obtained autoignition indicators for oxygenated hydrocarbons from our previous work (Schweidtmann et al., 2020). This dataset includes three ignition properties of interest for 505 molecules, hence represents the task of property prediction in low data regimes. As the underlying ML model, we use graph neural networks (GNNs), which have been shown as particular promising for molecular property prediction (Reiser et al., 2022; Heid et al., 2023).

Our contribution is to systematically investigate the effect of data augmentation by MTL for established ML models in property prediction, depending on the availability of molecular data. We find that MTL is often helpful in scenarios of predicting properties for molecules for which data on other related properties is available, i.e., hence for data/property completion tasks. Yet, we do not observe a proportionally larger performance gain from MTL for small datasets in comparison to larger ones. Our results also reveal that MTL can lead to decreasing prediction accuracy, even in some cases for highly correlated properties, highlighting the difficulty of incorporating different prediction tasks into the loss function and finding corresponding optima during ML model training compared to STL.

2. Methods

We compare different related data augmentation strategies of molecular data to train an ML model for predicting a molecular property p_1, as illustrated in Fig. 1. The accuracy of the model is evaluated on the basis of a test set that is separated from the available data and includes molecules and corresponding data labels for the property of interest, i.e., for p_1.

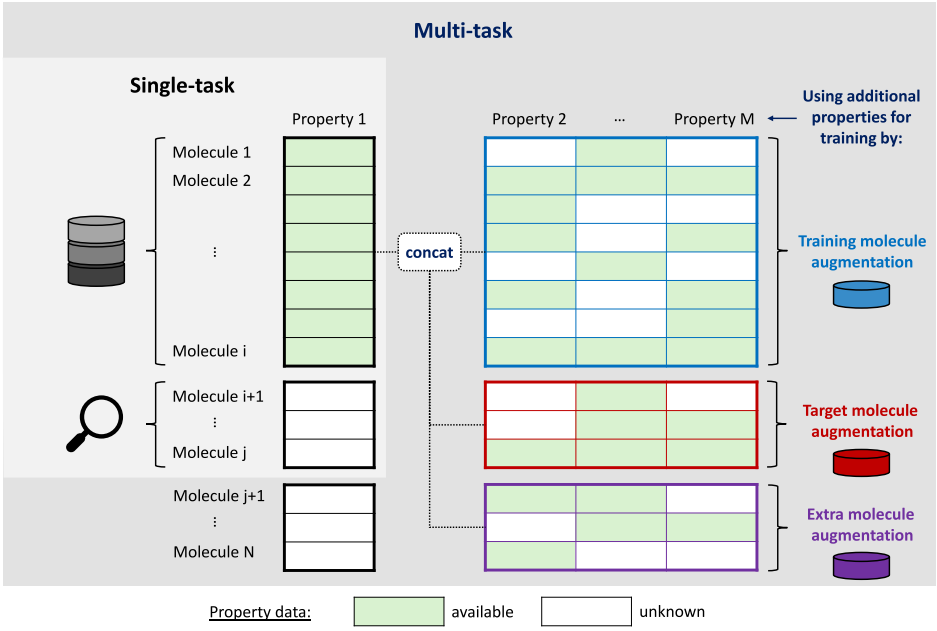

Fig. 1. Schematic illustration of learning setups for property 1: (i) single-task learning and (ii) multi-task learning with different data augmentation options.

Specifically, we consider STL and MTL. In **STL**, a single model is trained on data only for the target property p_1. The concept of **MTL** is to train a single model predicting multiple properties simultaneously, allowing the use of correlations between the target property p_1 and other properties, i.e., $p_2, \ldots, p_M$, to improve prediction accuracy. In MTL, the training data set used in STL is thus augmented with additional readily available property data.

We distinguish three types of additional property data that can be available for MTL, cf. Fig. 1, specifically data on:

- Training molecules: Additional data of related properties for the training molecules, i.e., the molecules, for which data on the target property p_1 is readily available, is utilized in training. This allows information from other, potentially related properties to be used for learning. For example, when training a model to predict the enthalpy of formation of a molecule, information about electronic stability, as indicated by the HOMO-LUMO gap, could be used.
- Target molecules: Data for other properties of the target molecules, i.e., the molecules to which a practitioner applies the model to obtain predictions, is included in the training set, whereas the target property labels are indeed unknown. Thus, contextual information about the target molecular structures is provided during training, which can be used for learning, analogously to the additional data of related properties for the training molecules.
- Extra molecules: Additional molecules for which data on the target property are lacking but data on other properties are readily available are added to the training set, thereby increasing the diversity of molecular structures, i.e., the coverage of the chemical space, used in training.

It should be noted that – in particular for experimental data – the augmented property labels are typically incomplete, that is, not all properties are readily available for all added molecules.

To investigate the effect of data augmentation with respect to the different data availability possibilities, we evaluate the following MTL forms:

- **MTL-Train:** MTL-Train refers to data augmentation for the training molecules only, i.e., property data for the target molecules is not known and extra molecules are also not utilized in model training. We can thereby study the effect of data augmentation when predictions are required for novel molecules, e.g., molecules proposed in computational molecular design, whose properties still need to be determined, i.e., by simulations, experiments or predictive modeling.
- **MTL-Complete:** Often data on other (related) properties is readily available for the target molecules, i.e., the molecules for which predictions of a particular property are required. This data can then be used in model training. MTL-Complete thus refers to using available additional property data for both the training and the target molecules. This allows us to investigate the scenario of *data completion*, predicting properties for molecules for which other property data is available.
- **MTL-Train/Complete + Extra:** Furthermore, there are many molecular databases that may contain other property data for molecules that are not in the training set and not explicitly targeted, i.e., extra molecules. MTL-Train/Complete + Extra thus refers to using properties of extra molecules in addition to the training and target molecules. Thereby, we analyze the effect of an increased molecular diversity with potentially related property data that is used in model training.

The data augmentation scenarios and MTL forms are applicable to any predictive molecular property prediction model, including descriptor-based approaches, also referred to as quantitative structure–property relationships (QSPRs), and deep end-to-end ML approaches such as GNNs and transformers. Here, we focus on GNNs, which we briefly describe next (Section 2.1). We then outline the computational experiments for comparing the different learning strategies (Section 2.2) including a description of the used data sets (Section 2.3), hyperparameter settings (Section 2.4), and evaluation metrics (Section 2.5).

2.1. Property prediction model

For developing molecular property prediction models, we use the Chemprop library (Heid et al., 2023), a GNN framework based on PyTorch (Ansel et al., 2024) for molecular applications. Chemprop was chosen for its various applications to molecular property prediction

tasks, ease of use, prior usage molecular benchmarks (e.g., QM9, see Section 2.3), and its built-in support for hyperparameter optimization and single- and multi-task learning setups. Chemprop utilizes directed message-passing neural networks (D-MPNNs), a GNN architecture designed for predicting molecular properties, where molecules are represented as graphs with atoms serving as vertices and bonds as edges; each bond is treated as two directed edges to capture directionality.

The Chemprop models learn a direct mapping from molecular graphs to the properties of interest. First, the SMILES representation of a molecule is converted into a molecular graph using RDKit, which generates initial feature vectors for both atoms (e.g., atomic number, formal charge) and bonds (e.g., bond type, ring membership). These features include information about stereochemistry, such as chirality. However, they do not capture the full spatial information possible with geometric GNN architectures, which require calculating atom coordinates and entail higher computational costs (Satorras et al., 2021; Batzner et al., 2022; Joshi et al., 2023; Adams et al., 2021; Duval et al., 2023). The atom and bond features are used to construct directed edge features by concatenating the initial atom features with the initial (undirected) bond features, providing the input for the message-passing phase. Secondly, the D-MPNN propagates information along the directed edges, iteratively updating their hidden representations. These updated edge representations capture local structural information in the molecular graph. After message passing, the hidden representations of the incoming edges to each atom are aggregated with that atom's initial feature vector using a learnable weight matrix, resulting in learned atomic embeddings. Finally, these atomic embeddings are combined (e.g., via summation or averaging, also cf. Schweidtmann et al. (2023)) to form a single molecular embedding, which is processed by a feed-forward neural network to predict molecule-level target properties. For single-task models, the neural network has one output, whereas for multi-task models, the number of outputs corresponds to the number of properties that are targeted.

2.2. Single- and multi-task training

Before training the GNN models, the property data are standardized to zero mean and unit standard deviation for each target independently by Chemprop. For single-task training, we then use a standard mean-squared error (MSE) loss function based on the deviation of predictions $\hat{p}$ and target property values p across all molecules used in training $m \in M_{\text{train}}$:

$$\text{STL-LOSS} = \frac{1}{|M_{\text{train}}|} \sum_{m \in M_{\text{train}}} (p_m - \hat{p}_m)^2$$

The loss function for the multi-task training is also based on the MSE and sums up the deviations for all target properties $p \in P_{target}$ of the training molecules. If the value for a property of a molecule is missing in the dataset used for training D_{train}, the individual loss term is neglected, i.e.:

$$\text{MTL-LOSS} = \frac{1}{|M_{\text{train}}|} \sum_{m \in M_{\text{train}}} \sum_{p \in P_{\text{target}}} (p_m - \hat{p}_m)^2 \cdot \mathbb{1}_{p_m \in D_{\text{train}}}$$

This approach of ignoring contributions from missing labels is standard in MTL for incomplete datasets. Although it prevents penalizing the model for unknown ground truth, a high degree of missing values for a particular task will naturally reduce its influence on the shared learned representation. The impact of such data sparsity is examined in our "Incomplete data" experiments (see Sections 3.2.2 and 3.2.3).

2.3. Datasets

We conducted computational experiments on two datasets in separate case studies: first, we consider the QM9 dataset (Ruddigkeit et al., 2012; Ramakrishnan et al., 2014) and then the fuel ignition dataset (Schweidtmann et al., 2020).

2.3.1. QM9 dataset

QM9 (Ruddigkeit et al., 2012; Ramakrishnan et al., 2014) is a comprehensive collection of approximately 134,000 small organic molecules, each containing up to nine heavy atoms (C, N, O, and F). We choose a large dataset to be able to construct different scenarios of data availability, e.g. by considering only small subsets. The dataset includes a wide range of properties, such as energetic and thermodynamic properties at constant pressure and temperature, derived from quantum chemical calculations based on density functional theory (DFT), making it a standard benchmark for molecular property prediction. As an initial exploratory step, we examined the inter-relationships between the 12 QM9 targets by calculating Pearson correlation coefficients (Fig. 2). To verify the robustness of these relationships, we also computed Spearman's and Kendall's rank correlations, which are non-parametric methods that account for non-linear monotonic relationships without assuming the data are normally distributed. Since these methods confirmed the same general trends, we present only the Pearson correlation matrix for clarity.

We selected 8 target properties from the QM9 dataset for our experiments, excluding highly correlated or derivative properties to ensure meaningful learning and comparisons. Specifically, we excluded $\Delta\epsilon$ (HOMO-LUMO gap) due to its direct derivation from ϵ_{HOMO} and ϵ_{LUMO}, which could lead to trivial predictions in the MTL-Complete case. Furthermore, U_{298}^{atom}, H_{298}^{atom}, and G_{298}^{atom} were excluded due to their extremely high correlations with each other and with U_0^{atom}. Only U_0^{atom} was retained to increase diversity in the selected targets. A summary of the eight selected target properties is presented in Table 1.

2.3.2. Fuel ignition dataset

This smaller dataset contains 505 molecules with three targets related to fuel ignition properties (Schweidtmann et al., 2020): Derived Cetane Number (DCN), Research Octane Number (RON), and Motor Octane Number (MON). They provide information about the ignition/knocking behavior, which are highly relevant to assess the suitability of molecules as potential fuel components in spark-ignition (RON/MON) and compression-ignition engines (Schweidtmann et al., 2020). Predicting DCNs, RONs, and MONs is thus desired to develop more sustainable fuel candidates. As such, the DCN is negatively correlated with RON and MON, while RON and MON are positively correlated with each other (see Fig. 3). Unlike the highly correlated and excluded properties in the QM9 dataset, DCN, RON, and MON cannot be derived directly from each other using mechanistic equations. Since obtaining each of the ignition indicators requires costly engine experiments, the considered fuel ignition dataset is inherently incomplete, i.e., only a subset of the three target properties is readily available for most molecules. Table 2 shows the number of molecules in the dataset that have each of the respective target properties available, as well as the mean and standard deviation of each property. We note that the data does not contain pressure- and temperature dependencies, as the properties are determined under standardized experimental engine test conditions, cf. Schweidtmann et al. (2020).

2.3.3. Dataset splits

For the QM9 dataset, we used both the full dataset split provided by the Chemprop benchmark (Heid et al., 2023) and three additional subsets: Small (1000 training molecules), Medium (10,000 training molecules), and Large (50,000 training molecules). In each case, the size of test and validation sets were each 10% of the training set size used. For each size category (apart from the full dataset case), we performed 10 independent runs with different random seeds to generate the train, validation, and test splits.

For the fuel ignition dataset, we used 10-fold cross-validation, ensuring that every molecule was used in both training/validation and testing across the folds. From each training fold, we left out what amounted to 10% of the full dataset for validation (for early stopping). The use of k-fold cross-validation is particularly advantageous for small datasets, as it maximizes the use of available data by allowing every data point to be used during both training and testing. This approach reduces the variance associated with limited sample sizes.

Table 1
The eight selected QM9 target properties, their descriptions, units, and summary statistics across the dataset (total size: 133,885).

Symbol	Description	Unit	Mean	Std Dev
μ	Dipole moment	Debye	2.71	1.53
α	Isotropic polarizability	$Bohr^3$	75.2	8.19
ϵ_{HOMO}	Highest occupied molecular orbital energy	eV	−0.240	0.0221
ϵ_{LUMO}	Lowest unoccupied molecular orbital energy	eV	0.0111	0.0469
$\langle R^2 \rangle$	Electronic spatial extent	$Bohr^2$	1190	280
ZPVE	Zero point vibrational energy	eV	0.149	0.0333
c_v	Molar heat capacity at 298.15 K	$cal\,mol^{-1}K^{-1}$	31.6	4.06
U_0^{atom}	Atomization energy at 0 K	eV	−1750	239

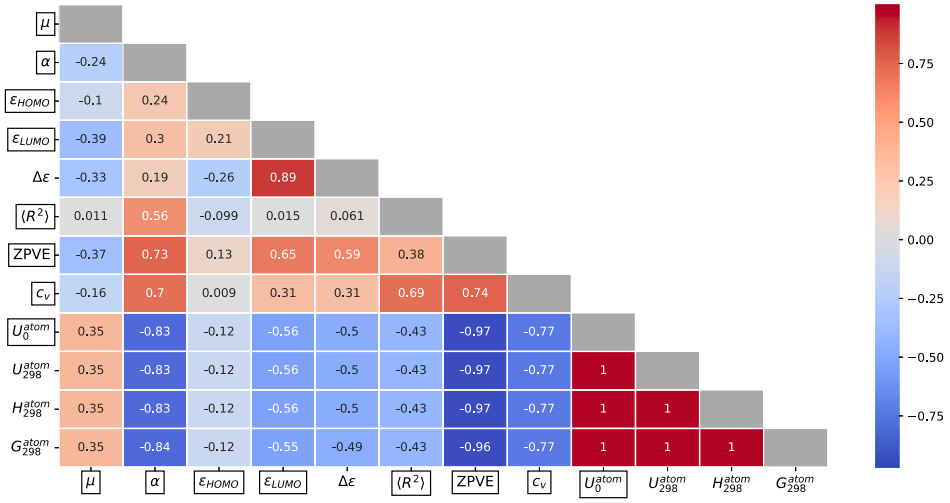

Fig. 2. Pearson correlation coefficients between all pairs of 12 QM9 target properties, with the 8 chosen for our experiments outlined.

Table 2
Number of molecules with available target property values, along with the summary statistics of each property, in the fuel ignition dataset (total size: 505).

Target property	Number of molecules	Mean	Std Dev
DCN	236	33.2	23.6
RON	335	86.0	25.2
MON	318	78.6	21.5

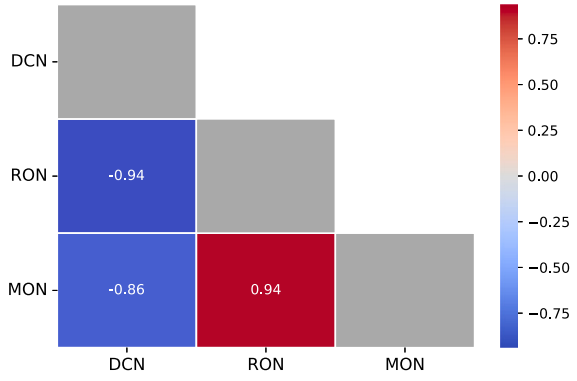

Fig. 3. Pearson correlation coefficients between DCN, RON, and MON.

2.4. Hyperparameter settings

For the QM9 experiments, we followed the same scheme outlined in the Chemprop benchmark. We used the hyperparameter optimization module in Chemprop which utilizes a Tree-structured Parzen Estimator (TPE). We optimized over 30 iterations with 50 epochs each for key hyperparameters: number of message-passing steps, hidden size, feed-forward network layers, feed-forward hidden size, and dropout ratio. Hyperparameter optimization was done separately for each size category, and furthermore carried out separately for each target task in the STL case. The optimized parameters identified for each size category in MTL-Train were reused for the corresponding size category in MTL-Complete.

For the fuel ignition experiments, we used a more extensive optimization scheme due to the small dataset size. Specifically, we performed 100 iterations with 200 epochs each, for the same hyperparameters as for QM9, with the addition of learning rate, batch size, and warm-up period. Each single-task case underwent independent optimization, while the parameters for MTL were reused for MTL-Complete. To ensure no data leakage across folds, hyperparameter optimization was carried out separately for each fold, using the train and validation data.

Models were trained for 50 epochs for QM9 experiments and 200 epochs for fuel ignition experiments. For each case, scaled sums were used to aggregate atomic features into molecular feature vectors. Test results were reported as the mean and standard deviation across splits or folds.

2.5. Evaluation metrics

We evaluated model performance using Root Mean Square Error (RMSE) for the QM9 experiments and Mean Absolute Error (MAE) for the DCN-RON-MON fuel ignition dataset. This distinction was made to best reflect the objectives for each task. For the large, computational QM9 dataset, we selected RMSE as the primary evaluation metric. Since our models were trained to minimize Mean Squared Error (MSE), RMSE provides the most direct assessment of performance against the training objective. Furthermore, its inherent sensitivity to large deviations is a key diagnostic feature for evaluation: a low RMSE score signifies that a model is consistently reliable and avoids significant, physically unrealistic predictions across QM9's chemical space. This metric is also reported in the Chemprop benchmark on QM9, facilitating comparison with that work (Heid et al., 2023).

Conversely, for the small, experimental fuel ignition dataset, we selected MAE primarily for its robustness to the outliers that are common in experimental measurements and for its direct interpretation of the average error magnitude. This choice is also consistent with the precedent set by prior work which utilizes this dataset (Schweidtmann et al., 2020; Neumann et al., 2024).

Note that, for completeness, we report both error metrics, RMSE and MAE, for all computational experiments in the Supporting Information.

In the QM9 experiments, we compared results across STL and variations of MTL-Train, MTL-Complete, and MTL-Complete + Extra, for all 8 target properties. Similarly, for the fuel ignition dataset, we conducted comparisons for its 3 target properties.

3. Computational experiments and results

3.1. Comparison to Chemprop-QM9 benchmark

To provide a fair and direct comparison with existing work, we first reproduced the experiments from the Chemprop-QM9 benchmark by Heid et al. (2023) on the QM9 dataset, which provided results for STL models (for 2 properties only) and an MTL-Train model trained on all 12 QM9 properties, as by our definition in Section 2. We additionally provide results for the 10 remaining STL and the MTL-Complete scenarios using the same train–test–validation split (with a 80:10:10 ratio) as provided in the benchmark. To prevent MTL-Complete simply benefiting from an increased amount of training data, we deleted the same number of original training molecules as the target molecules that were added to the training set. To maintain the same conditions as in the original benchmark, during final training, each model was trained on a single data split with an ensemble size of 5, while no ensembling was applied during hyperparameter optimization. The "Extra" scenarios are not considered here because the full dataset is already used for MTL-Train and MTL-Complete, so we do not have additional related data of the 12 properties available for training.

The results for the Chemprop-QM9 are presented in Table 3. The results indicate that, for nearly all target properties (with the exception of ZPVE), the MTL-Complete approach achieves the lowest error, often by a considerable margin. In contrast, MTL generally underperforms compared to STL, suggesting that MTL-Complete is using correlations among the various properties provided during training to improve predictions of the missing target property.

For certain properties with well-established correlations, such as the HOMO-LUMO energies and the gap $\Delta\epsilon$ – derived by subtracting ϵ_{LUMO} from ϵ_{HOMO} – the performance improvement from MTL-Complete is particularly pronounced. Notably, while STL still outperforms or is on par with the simple MTL approach in predicting these three properties, the MTL-Complete model achieves the highest accuracy on them by a dramatic margin. This suggests that having access to related properties for the specific target molecules is a critical factor in achieving enhanced predictive performance. Importantly, even for targets where the connections between them are not trivial (i.e., μ, α, $\langle R^2 \rangle$, c_v), the MTL-Complete method provides an advantage, indicating that it is capturing subtle inter-property relationships. As previously noted in Section 2.3.1, to ensure meaningful learning and a situation where MTL-Complete does not have an "unfair advantage" due to the presence of derivative or very similar auxiliary targets, in the subsequent experiments we exclude from usage the HOMO-LUMO gap $\Delta\epsilon$ and all except one (U_0^{atom}) of the highly inter-correlated atomization energies.

3.2. QM9 experiments

Following the Chemprop comparison, we conducted a broader set of experiments on the QM9 dataset to systematically evaluate the performance of single-task and multi-task approaches under varying conditions.

Table 3
Comparison to the Chemprop benchmark: test RMSEs for 12 QM9 targets across STL, MTL-Train, and MTL-Complete configurations.

Target	STL	MTL-Train	MTL-Complete
μ	0.577	0.586	**0.539**
α	0.542	0.52	**0.382**
ϵ_{HOMO}	0.00417	0.00421	**0.00213**
ϵ_{LUMO}	0.00384	0.0041	**0.00132**
$\Delta\epsilon$	0.00596	0.00587	**0.00149**
$\langle R^2 \rangle$	31.6	33.1	**28.8**
ZPVE	**0.000239**	0.000375	0.000361
c_v	0.211	0.224	**0.174**
U_0^{atom}	2.57	3.32	**1.65**
U_{298}^{atom}	2.61	3.33	**1.66**
H_{298}^{atom}	2.57	3.33	**1.67**
G_{298}^{atom}	2.52	3.3	**1.78**

3.2.1. Dataset size variations

We used three progressively larger subsets of the QM9 dataset (Small, Medium, Large) and a Full split:

- **Small:** 1000 training molecules, 100 target molecules, and 100 validation molecules.
- **Medium:** 10,000 training molecules, 1000 target molecules, and 1000 validation molecules.
- **Large:** 50,000 training molecules, 5000 target molecules, and 5000 validation molecules.
- **Full:** Full QM9 dataset with the Chemprop benchmark split.

3.2.2. Experimental cases

We are first interested in the general comparison of STL and MTL using different data augmentation schemes. Then, in two ablation studies, we respectively investigate the effects of the molecular diversity and the completeness of the data set on the prediction accuracy.

STL & MTL comparison

For each split and across all targets, we trained single-task models (STL) and multi-task models (i.e., MTL-Train and MTL-Complete). For STL, training was conducted separately for each target, with an independent model trained for each property. MTL-Train and MTL-Complete, one model is trained on all 8 targets simultaneously. MTL-Train only considers the molecules and corresponding property labels of the training set, whereas MTL-Complete also includes the molecules of the test set but only the property labels for 7 of the properties, excluding the target property and thereby avoiding data leakage (cf. Section 2).

Ablation study: Molecular diversity

To investigate whether potential performance changes by MTL-Complete in comparison to MTL-Train originate from an increased molecular diversity (or data set size), we also performed MTL-Complete with a reduced training set size, to which we refer to as MTL-Complete + Deletion. That is, we maintained the same number of molecules used for training as in the MTL-Train case. This was achieved by removing a corresponding number of molecules from the original training set upon adding target molecules. This setup thus isolates the effect of observing other target properties of target molecules during training without increasing the overall data volume.

To also investigate whether adding further molecules beyond the train and target ones has an additional effect on the performance, we here also consider MTL-Complete + Extra (cf. Section 2) as described in the following. Molecules from the full QM9 dataset's training set (from the original Chemprop benchmark) that were not part of the current training, validation, or test splits are added to the training set. The number of added molecules matches the size of the training set for each category: 1000 for Small, 10,000 for Medium, and 50,000 for Large splits. Notably, this case is not conducted for the Full split

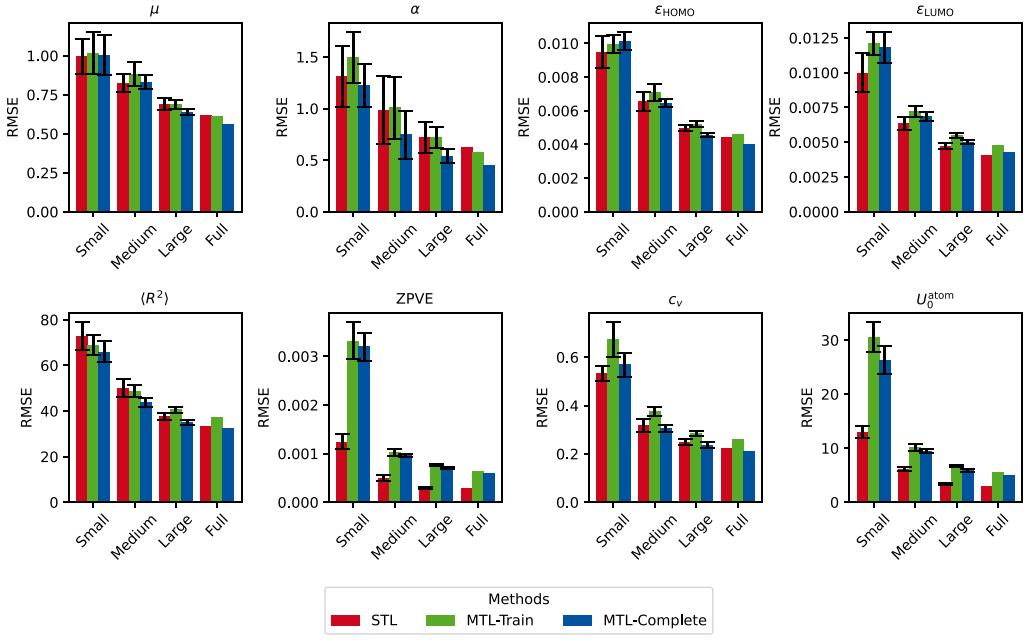

Fig. 4. Test performance for QM9 experiments across varying dataset sizes with STL, MTL, and MTL-Complete.

due to the lack of additional molecules. Analogously to MTL-Complete, the target property under evaluation was removed from the extra molecules avoiding data leakage. This setup thus isolates the effect of increasing molecular diversity by using unseen molecules with other targets available but not the specific target of interest.

Ablation study: Incomplete data

As molecular property data sets, particularly experimental ones, typically contain missing property labels, we also conducted training runs with artificially generated incomplete QM9 data subsets. Specifically, for MTL-Train and MTL-Complete, we randomly removed 25%/50% of the data from each of the 7 property columns, that is, all properties except for the target property, to ensure comparability with STL. The indices of the rows from which the values are removed are independently random and not necessarily the same for all these columns. This ablation study thus simulates the situation where we have a dataset with a single target label (i.e., one of the 8 targets) that is augmented with other datasets containing other target properties with some overlapping molecules. This experiment was only carried out on the full QM9 dataset.

3.2.3. Discussion of results

The results indicate several overarching trends across the selected targets in the QM9 dataset subsets. Here, we discuss the key patterns, noting exceptions, and their possible explanations.

Effect of multi-task learning

As observed in Fig. 4, for many targets, STL consistently outperforms MTL-Train, particularly at smaller dataset sizes. This effect is most pronounced for ϵ_{HOMO}, ϵ_{LUMO}, c_v, and U_0^{atom}, where MTL-Train struggles regardless of the dataset size. A likely explanation is that the additional targets used in MTL-Train training provide little relevant information for these specific properties. In some cases, they may even introduce conflicting signals that hinder learning, since the loss function involves multiple properties which can make optimization more difficult. For $\langle R^2 \rangle$, we see the opposite trend: MTL-Train shows better performance than STL for smaller subsets but begins to lag as the dataset size increases. This suggests that MTL-Train may benefit more in cases where learnable relations between targets are stronger and the available data is limited, but the advantage diminishes when larger datasets allow STL to dominate.

Also observable in Fig. 4 is that MTL-Complete generally outperforms both STL and MTL-Train as the dataset size increases. Notably, MTL-Complete outperforms both STL and MTL-Train across all dataset sizes for the targets α and $\langle R^2 \rangle$. For the targets μ, ϵ_{HOMO}, and c_v, with increasing dataset size, MTL-Complete consistently narrows and reverses the advantage STL holds at smaller scales. This highlights MTL-Complete's ability to leverage inter-target relationships improving when sufficient data is available to learn contextual information about the target molecules during training. More generally, having any form of information about the target molecules can provide the model with a valuable starting point for predictions, allowing it to potentially exploit relevant features and correlations.

Notably, for ZPVE and U_0^{atom}, STL greatly outperforms *both* MTL-Train and MTL-Complete across all dataset sizes. Similar behavior is observed for ϵ_{LUMO}, although with a much narrower performance gap between STL and MTL-Train/Complete. However, MTL-Complete still outperforms MTL-Train for these properties.

Effect of molecular diversity

We also analyze the effects of decreasing and increasing the volume of training data available to MTL-Complete via the MTL-Complete + Deletion and MTL-Complete + Extra cases. These are visualized in Fig. 5. The MTL-Complete + Deletion experiments test the effect of keeping the overall training data volume the same as in regular STL/MTL while incorporating related property data for the target molecules. Meanwhile, the "extra molecules" experiments are designed to increase the volume of training data beyond that of MTL-Complete and assess whether the benefits of MTL-Complete stem specifically from including information on the target molecules during training. These experiments involve adding data that is not part of the train–test–validation splits from the full QM9 training set. For small and medium dataset sizes, including extra molecules generally improves MTL-Complete performance. This indicates that the added data provides useful additional context when the available training set is limited. However, for the large dataset size, adding extra molecules no longer confers a clear advantage and occasionally leads to slightly worse performance. This suggests that as the dataset size grows, the inclusion of unrelated data

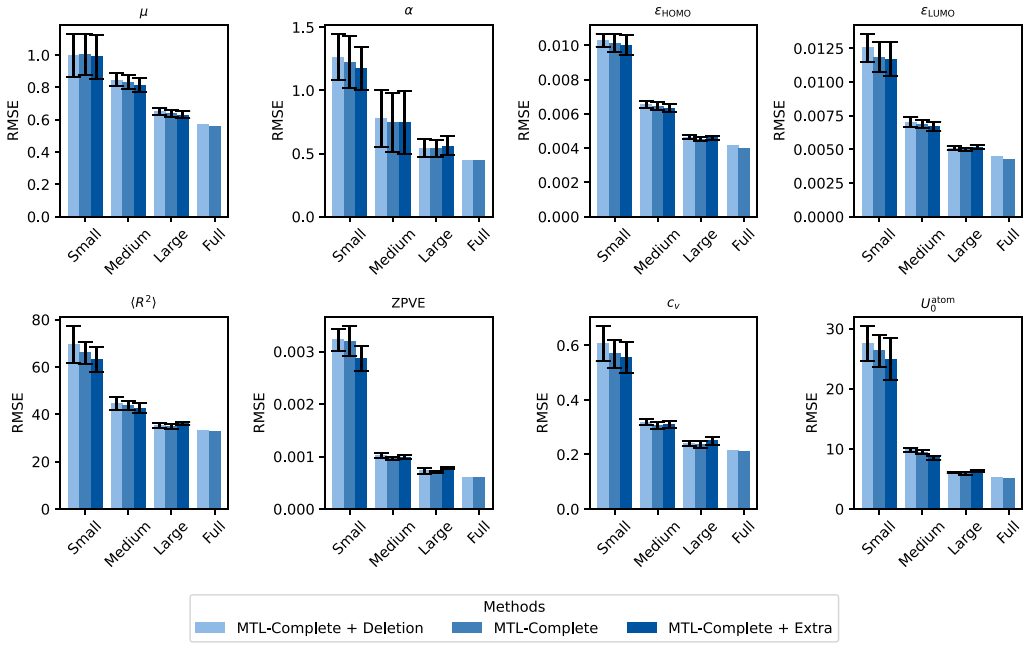

Fig. 5. Test results for QM9 experiments across varying dataset sizes with varying amounts of training data.

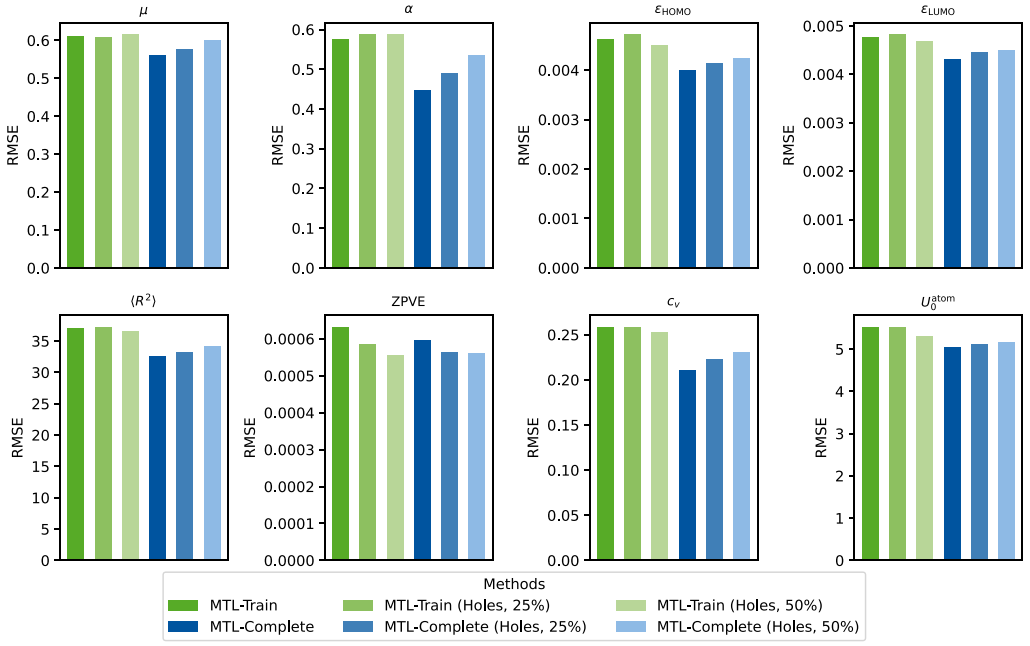

Fig. 6. Test results for QM9 experiments (with the full dataset) with varying proportions of missing data; i.e., "holes" in the data.

may increase noise or saturate the ability of the model to extract meaningful relationships.

The performance of MTL-Complete + Deletion is comparable to that of MTL-Complete in most cases, see Fig. 5. Although there is a somewhat consistent decrease in performance for MTL-Complete + Deletion compared to MTL-Complete on the small and, to a lesser extent, medium-sized dataset, there is no overall appreciable performance gap for large and full dataset sizes. This indicates that losing a similar proportion of training data is more detrimental when working with a smaller dataset as a whole. Nevertheless, the gap between MTL-Complete and MTL-Complete + Deletion is usually much smaller than the gap between MTL-Complete and MTL-Train. This strongly suggests that the primary reason MTL-Complete outperformed MTL-Train is its ability to leverage information about other properties of the target molecules during training, rather than it merely benefiting from a larger overall volume of training data.

Effect of incomplete data

We also examine the effect of varying amounts of missing data ("holes") in the full dataset case for MTL-Train and MTL-Complete (see Fig. 6). As expected, performance generally declines for increasing number of missing entries. Interestingly, for targets like ZPVE and U_0^{atom}, where STL already significantly outperforms the MTL approaches, this trend is partially reversed. This suggests that removing conflicting or redundant auxiliary targets – by way of missing entries – can, in some cases, improve learning for specific targets where MTL fails to provide a clear benefit.

Overall, for the QM9 training runs, MTL-Complete demonstrates significant improvements for many targets, as it can exploit inter-target correlations more effectively and provides some contextual information on the target molecules during training that the model seems to learn. However, its performance remains target-dependent, with STL consistently performing better for certain properties, i.e., ZPVE, U_0^{atom}, and ϵ_{LUMO}. The inclusion of extra data helps primarily for smaller datasets, while its utility diminishes as the dataset size increases. Moreover, the behavior of MTL-Train and MTL-Complete under missing data underscores the importance of carefully selecting auxiliary targets to avoid introducing signals that may hinder learning.

3.3. Fuel ignition dataset experiments

Similarly to the experiments conducted with QM9, we performed single-task and different multi-task training runs on the fuel ignition dataset which is based on experimental data and is by nature incomplete and of small size (cf. Section 2.3). We consider STL, i.e., training a separate model for each target (DCN, RON, and MON), and MTL, where one model is trained on all targets simultaneously for the training molecules (MTL-Train) or for both the training and target molecules, excluding the target property for the latter (MTL-Complete). Since we found in the QM9 experiments that augmenting extra molecules can lead to increased accuracy, especially for small data set sizes, and since the fuel ignition data set is incomplete, i.e., not all property values are available for all molecules, we here directly include the "extra molecules" (cf. Section 2) for the two MTL approaches. For example, if only RON and MON values are available for a specific molecule, this data is also included in the MTL training runs targeting DCN. We note that we also found the use of the full fuel ignition data set for MTL to be helpful in our previous study of DCN, RON and MON prediction (Schweidtmann et al., 2020).

In addition to extra molecules with available DCN, RON, and MON data, we are also interested in whether additional data augmentation further increases predictive accuracy. We consider two additional data augmentation strategies:

Extra QM9 targets: We identify molecules in the fuel ignition dataset that also appear in the QM9 dataset. We augment the fuel ignition dataset for these shared molecules by adding their corresponding 8 QM9 properties as additional targets. The objective is to assess whether including additional target properties during training could help the model predict DCN, RON, and MON by potentially leveraging correlations between the original targets and the augmented QM9 properties through MTL.

Extra QM9 molecules: We augment molecules from the QM9 dataset that are not available in the fuel ignition dataset. Specifically, we randomly add as many extra QM9 molecules as there are molecules in the training set of the fuel ignition data set. Note that these extra molecules only have the 8 QM9 target labels available and none of DCN, RON, or MON. Furthermore, we only use those QM9 molecules as extra molecules with no atom types other than those present in the fuel ignition data set, i.e., C, H, and O. Here, we aim to further increase the molecular diversity available for MTL.

Discussion of results

As observed in Fig. 7, for DCN, STL outperforms all variants of MTL-Train, suggesting that the inclusion of additional targets (RON and MON) introduces limited or even conflicting information that does not aid in predicting DCN. This negative effect of MTL-Train is unexpected given the high correlation between DCN and RON, cf. Section 2.3. In our previous works (Schweidtmann et al., 2020; Neumann et al., 2024), we found that MTL based on another GNN architecture outperformed STL for DCN prediction; notably, we used another performance evaluation strategy with a fixed external test set instead of cross validation, so the results are not directly comparable and are also sensitive to outliers given the small dataset size. MTL-Complete, however, shows a clear improvement over both STL and MTL-Train, indicating that leveraging related properties of the target molecules helps capture useful inter-property relationships. Augmenting the data with extra QM9 properties, on the other hand, does not confer any benefit for learning DCN for MTL-Train, and worsens performance when using MTL-Complete. Using extra molecules worsens the result significantly for both MTL-Train and MTL-Complete. This suggests that the unrelated additional properties introduce noise rather than helpful information for this particular property, which already suffers from limited data availability.

For RON and MON, the patterns are notably different. MTL-Train consistently outperforms STL, demonstrating that these two properties benefit from shared learning when trained together. Augmenting with extra QM9 properties further improves performance, or at the very least does not degrade it, indicating that the additional molecular properties provide relevant auxiliary information that helps the model. However, adding extra QM9 molecules, significantly hinders performance. This suggests that while RON and MON benefit from extra target properties

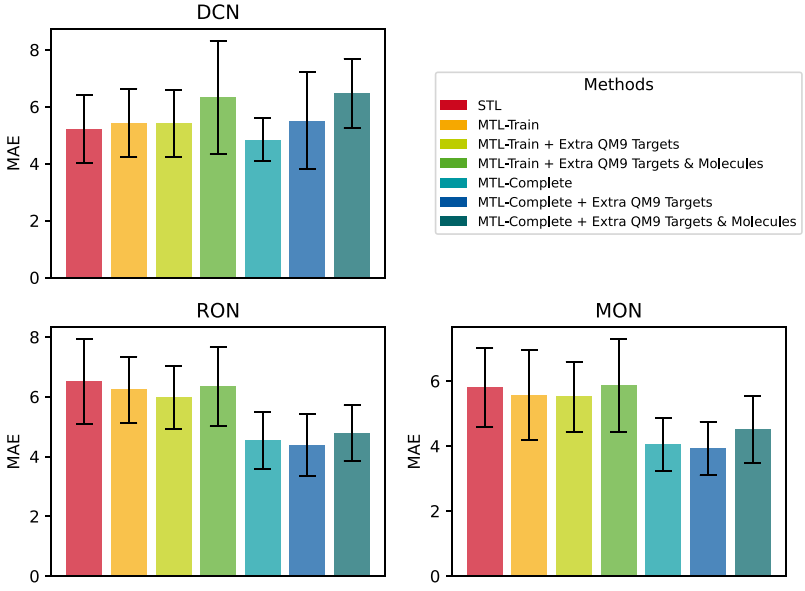

Fig. 7. Test performance for fuel ignition experiments across varying training configurations.

available for the same molecules, the inclusion of extra molecules from the QM9 dataset, which only have QM9 properties available, introduces noise that disrupts learning of RON and MON. This is a case of negative transfer when augmenting with dissimilar data. These trends hold for MTL-Complete as well, with the overall performance being consistently better than STL and MTL-Train across all variations.

Overall, MTL-Complete provides clear advantages for all three targets when compared to STL and MTL-Train. However, the extent of the benefit is target-dependent. DCN remains the most challenging property to learn (likely due to having the least data available), with significant performance degradation from extra QM9 molecules. In contrast, RON and MON demonstrate the advantages of MTL setups, with MTL-Train outperforming STL and with MTL-Complete with augmented targets achieving the best results overall for these two targets. Adding unrelated extra molecules with other targets tends to hinder performance across the board, indicating the importance of carefully selecting relevant auxiliary data for small and incomplete datasets.

4. Conclusion

We systematically explore the utility of MTL for molecular property prediction depending on molecular data availabilities. Specifically, we consider practically-motivated scenarios, e.g., small and incomplete datasets, and the availability of related property data for training molecules (MTL-Train) plus target molecules (MTL-Complete) as well as extra molecules. We train GNNs on different datasets to empirically analyze the effects of these MTL forms on the prediction accuracy in comparison to STL.

Studying the QM9 dataset with multiple, differently correlated properties shows that MTL-Train often leads to decreased accuracy compared to STL, indicating that property relationships of the training molecules can hardly be utilized and are diminished by increased difficulty of model training due to multiple loss terms. In contrast, MTL-Complete results in significant accuracy increases for many targets, so contextual information on the target molecules in addition to the training molecules often facilitates the models in exploiting additional property information. MTL is thus particularly helpful in completing property data, i.e., predicting properties of molecules for which other property data is readily available. We further find augmenting extra molecules with related property data, thereby increasing the molecular diversity available in training, to be typically helpful for smaller datasets. As expected, missing property data for molecules, i.e., sparse/incomplete datasets, decrease the accuracy gains by MTL.

Our findings are also transferable to a real-world, experimental dataset of fuel ignition data, representing property prediction in low data regimes. Here, augmenting property data on the training and target molecules with MTL-Complete also leads to the most prominent accuracy improvements; notably, the improvements vary for different properties. In contrast, augmenting molecular data with unrelated properties or with extra molecules from a dissimilar chemical domain can decrease the accuracy, highlighting the critical importance of relevance in data augmentation strategies. Overall, if related property data for target molecules is readily available, MTL-Complete should thus be utilized for improving predictive accuracy in real-world chemical applications.

While we have focused on different data availability scenarios using a GNN model with a equal contribution of the properties to the loss function, future work could analyze our scenarios with more sophisticated loss formulations of MTL, e.g., Liu et al. (2022) and Chen et al. (2018), and other ML approaches. Moreover, to overcome the negative transfer observed when augmenting with dissimilar data, future work could explore more advanced strategies. One promising avenue is to employ a pre-train and fine-tune approach, leveraging large molecular models trained with self-supervision (Rong et al., 2020; Chithrananda et al., 2020; Méndez-Lucio et al., 2024; Li and Fourches, 2020; Li et al., 2021) to create robust initial representations. Alternatively, for MTL scenarios that still incorporate such disparate data, methods to mitigate negative transfer during training are crucial. These could range from simple similarity-based filtering of the additional data to employing the task-balancing strategies we previously mentioned, such as normalizing task-specific gradients (Chen et al., 2018) or adaptively weighting property losses (Biswas et al., 2023), in order to actively down-weight the influence of less relevant tasks.

Further, it would be highly interesting to address the question of why models employing MTL are not able to learn that the properties are not related. In fact, if a model is large enough, it could also learn with MTL independent mappings from the molecular structure to the properties, i.e., the ones learned with STL. This could be further investigated by considering the model size as well as the loss function and its optimization in training.

CRediT authorship contribution statement

Muhammad bin Javaid: Writing – original draft, Visualization, Software, Methodology, Formal analysis, Data curation, Conceptualization. **Timo Gervens:** Writing – original draft, Methodology, Formal analysis, Conceptualization. **Alexander Mitsos:** Writing – review & editing, Supervision, Funding acquisition. **Martin Grohe:** Writing – review & editing, Supervision, Funding acquisition. **Jan G. Rittig:** Writing – original draft, Visualization, Methodology, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This project was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation), Germany – 466417970 – within the Priority Programme “SPP 2331: Machine Learning in Chemical Engineering”.

This work was also performed as part of the Helmholtz School for Data Science in Life, Earth and Energy (HDS-LEE).

The project was also funded by the European Union (ERC, SymSim, 101054974). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

Further Funding by the Werner Siemens Foundation within the WSS project of the century “catalaix” is acknowledged.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.compchemeng.2025.109253.

Data availability

The code and data used in this study are available at our GitLab repository at https://git.rwth-aachen.de/muhammad.javaid/exploring-data-augmentation.

References

Adams, K., Pattanaik, L., Coley, C.W., 2021. Learning 3d representations of molecular chirality with invariance to bond rotations. arXiv preprint arXiv:2110.04383.

Alhamoud, K., Ghunaim, Y., Alshehri, A.S., Li, G., Ghanem, B., You, F., 2024. Leveraging 2D molecular graph pretraining for improved 3D conformer generation with graph neural networks. Comput. Chem. Eng. 183, 108622.

Allenspach, S., Hiss, J.A., Schneider, G., 2024. Neural multi-task learning in drug design. Nat. Mach. Intell. 6 (2), 124–137.

Alshehri, A.S., Gani, R., You, F., 2020. Deep learning and knowledge-based methods for computer-aided molecular design—toward a unified approach: State-of-the-art and future directions. Comput. Chem. Eng. 141, 107005.

Alshehri, A.S., You, F., 2022. Deep learning to catalyze inverse molecular design. Chem. Eng. J. 444, 136669.

Ansel, J., Yang, E., He, H., Gimelshein, N., Jain, A., Voznesensky, M., Bao, B., Bell, P., Berard, D., Burovski, E., Chauhan, G., Chourdia, A., Constable, W., Desmaison, A., DeVito, Z., Ellison, E., Feng, W., Gong, J., Gschwind, M., Hirsh, B., Huang, S., Kalambarkar, K., Kirsch, L., Lazos, M., Lezcano, M., Liang, Y., Liang, J., Lu, Y., Luk, C., Maher, B., Pan, Y., Puhrsch, C., Reso, M., Saroufim, M., Siraichi, M.Y., Suk, H., Suo, M., Tillet, P., Wang, E., Wang, X., Wen, W., Zhang, S., Zhao, X., Zhou, K., Zou, R., Mathews, A., Chanan, G., Wu, P., Chintala, S., 2024. Pytorch 2: Faster machine learning through dynamic python bytecode transformation and graph compilation. In: Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems. ASPLOS, ACM, pp. 929–947.

Atz, K., Grisoni, F., Schneider, G., 2021. Geometric deep learning on molecular representations. Nat. Mach. Intell. 3 (12), 1023–1032.

Batzner, S., Musaelian, A., Sun, L., Geiger, M., Mailoa, J.P., Kornbluth, M., Molinari, N., Smidt, T.E., Kozinsky, B., 2022. E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. Nat. Commun. 13 (1), 2453.

Beaini, D., Huang, S., Cunha, J.A., Li, Z., Moisescu-Pareja, G., Dymov, O., Maddrell-Mander, S., McLean, C., Wenkel, F., Müller, L., Mohamud, J.H., Parviz, A., Craig, M., Koziarski, M., Lu, J., Zhu, Z., Gabellini, C., Klaser, K., Dean, J., Wognum, C., Sypetkowski, M., Rabusseau, G., Rabbany, R., Tang, J., Morris, C., Koutis, I., Ravanelli, M., Wolf, G., Tossou, P., Mary, H., Bois, T., Fitzgibbon, A., ażej Banaszewski, B., Martin, C., Masters, D., 2023. Towards foundational models for molecular learning on large-scale multi-task datasets. arXiv preprint arXiv:2310.04292.

Biswas, S., Chung, Y., Ramirez, J., Wu, H., Green, W.H., 2023. Predicting critical properties and acentric factors of fluids using multitask machine learning. J. Chem. Inf. Model. 63 (15), 4574–4588.

Bjerrum, E.J., 2017. SMILES enumeration as data augmentation for neural network modeling of molecules. arXiv preprint arXiv:1703.07076.

Brozos, C., Rittig, J.G., Bhattacharya, S., Akanny, E., Kohlmann, C., Mitsos, A., 2024. Graph neural networks for surfactant multi-property prediction. Colloids Surf. A: Physicochem. Eng. Asp. 694, 134133.

Caruana, R., 1997. Multitask learning. Mach. Learn. 28 (1), 41–75.

Chaparro, G., Müller, E.A., 2023. Development of thermodynamically consistent machine-learning equations of state: Application to the Mie fluid. J. Chem. Phys. 158 (18), 184505.

Chaparro, G., Müller, E.A., 2024. Development of a Helmholtz free energy equation of state for fluid and solid phases via artificial neural networks. Commun. Phys. 7 (1), 406.

Chen, Z., Badrinarayanan, V., Lee, C.-Y., Rabinovich, A., 2018. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In: International Conference on Machine Learning. PMLR, pp. 794–803.

Chithrananda, S., Grand, G., Ramsundar, B., 2020. ChemBERTa: large-scale self-supervised pretraining for molecular property prediction. arXiv preprint arXiv:2010.09885.

Cortes-Ciriano, I., Bender, A., 2015. Improved chemical structure–activity modeling through data augmentation. J. Chem. Inf. Model. 55 (12), 2682–2692.

Dahl, G.E., Jaitly, N., Salakhutdinov, R., 2014. Multi-task neural networks for QSAR predictions. arXiv preprint arXiv:1406.1231.

Dey, V., Ning, X., 2024. Enhancing molecular property prediction with auxiliary learning and task-specific adaptation. J. Cheminformatics 16 (1), 85.

Duval, A., Mathis, S.V., Joshi, C.K., Schmidt, V., Miret, S., Malliaros, F.D., Cohen, T., Liò, P., Bengio, Y., Bronstein, M., 2023. A hitchhiker's guide to geometric gnns for 3d atomic systems. arXiv preprint arXiv:2312.07511.

Felton, K.C., Raßpe-Lange, L., Rittig, J.G., Leonhard, K., Mitsos, A., Meyer-Kirschner, J., Knösche, C., Lapkin, A.A., 2024. ML-SAFT: a machine learning framework for PCP-SAFT parameter prediction. Chem. Eng. J. 492, 151999.

Gao, Q., Dukker, T., Schweidtmann, A.M., Weber, J.M., 2024. Self-supervised graph neural networks for polymer property prediction. Mol. Syst. Des. Eng. 9 (11), 1130–1143.

Gertig, C., Leonhard, K., Bardow, A., 2020. Computer-aided molecular and processes design based on quantum chemistry: current status and future prospects. Curr. Opin. Chem. Eng. 27, 89–97.

Heid, E., Greenman, K.P., Chung, Y., Li, S.-C., Graff, D.E., Vermeire, F.H., Wu, H., Green, W.H., McGill, C.J., 2023. Chemprop: a machine learning package for chemical property prediction. J. Chem. Inf. Model. 64 (1), 9–17.

Jiang, J., Zhang, R., Yuan, Y., Li, T., Li, G., Zhao, Z., Yu, Z., 2023. NoiseMol: A noise-robusted data augmentation via perturbing noise for molecular property prediction. J. Mol. Graph. Model. 121, 108454.

Jirasek, F., Hasse, H., 2023. Combining machine learning with physical knowledge in thermodynamic modeling of fluid mixtures. Annu. Rev. Chem. Biomol. Eng. 14, 31–51.

Joshi, C.K., Bodnar, C., Mathis, S.V., Cohen, T., Lio, P., 2023. On the expressive power of geometric graph neural networks. In: International Conference on Machine Learning. PMLR, pp. 15330–15355.

Karniadakis, G.E., Kevrekidis, I.G., Lu, L., Perdikaris, P., Wang, S., Yang, L., 2021. Physics-informed machine learning. Nat. Rev. Phys. 3 (6), 422–440.

Klaser, K., Banaszewski, B., Maddrell-Mander, S., McLean, C., Müller, L., Parviz, A., Huang, S., Fitzgibbon, A.W., 2024. Minimol: A parameter-efficient foundation model for molecular learning. In: ICML Workshop on Efficient and Accessible Foundation Models for Biological Discovery.

Korolev, V., Mitrofanov, A., Korotcov, A., Tkachenko, V., 2019. Graph convolutional neural networks as "general-purpose" property predictors: the universality and limits of applicability. J. Chem. Inf. Model. 60 (1), 22–28.

Koscher, B.A., Canty, R.B., McDonald, M.A., Greenman, K.P., McGill, C.J., Bilodeau, C.L., Jin, W., Wu, H., Vermeire, F.H., Jin, B., Hart, T., Kulesza, T., Li, S.-C., Jaakkola, T., Barzilay, R., Gómez-Bombarelli, R., Green, W.H., Jensen, K.F., 2023. Autonomous, multiproperty-driven molecular discovery: From predictions to measurements and back. Science 382 (6677), eadi1407.

Li, X., Fourches, D., 2020. Inductive transfer learning for molecular activity prediction: Next-Gen QSAR models with MolPMoFiT. J. Cheminformatics 12 (1), 27.

Li, Z., Jiang, M., Wang, S., Zhang, S., 2022. Deep learning methods for molecular representation and property prediction. Drug Discov. Today 27 (12), 103373.

Li, P., Wang, J., Qiao, Y., Chen, H., Yu, Y., Yao, X., Gao, P., Xie, G., Song, S., 2021. An effective self-supervised framework for learning expressive molecular global representations to drug discovery. Brief. Bioinform. 22 (6), bbab109.

Li, S.-C., Wu, H., Menon, A., Spiekermann, K.A., Li, Y.-P., Green, W.H., 2024. When do quantum mechanical descriptors help graph neural networks to predict chemical properties? J. Am. Chem. Soc. 146 (33), 23103–23120.

Liu, S., Liang, Y., Gitter, A., 2019. Loss-balanced task weighting to reduce negative transfer in multi-task learning. In: Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 33, pp. 9977–9978.

Liu, Z., Lin, L., Jia, Q., Cheng, Z., Jiang, Y., Guo, Y., Ma, J., 2021. Transferable multilevel attention neural network for accurate prediction of quantum chemistry properties via multitask learning. J. Chem. Inf. Model. 61 (3), 1066–1082.

Liu, S., Qu, M., Zhang, Z., Cai, H., Tang, J., 2022. Structured multi-task learning for molecular property prediction. In: Camps-Valls, G., Ruiz, F.J.R., Valera, I. (Eds.), Proceedings of the 25th International Conference on Artificial Intelligence and Statistics. In: Proceedings of Machine Learning Research, vol. 151, PMLR, pp. 8906–8920.

Magar, R., Wang, Y., Lorsung, C., Liang, C., Ramasubramanian, H., Li, P., Farimani, A.B., 2022. AugLiChem: data augmentation library of chemical structures for machine learning. Mach. Learn.: Sci. Technol. 3 (4), 045015.

Masi, F., Stefanou, I., Vannucci, P., Maffi-Berthier, V., 2021. Thermodynamics-based artificial neural networks for constitutive modeling. J. Mech. Phys. Solids 147, 104277.

Méndez-Lucio, O., Nicolaou, C.A., Earnshaw, B., 2024. MolE: A foundation model for molecular graphs using disentangled attention. Nat. Commun. 15 (1), 9431.

Neumann, M., Rittig, J.G., Letaief, A.B., Honecker, C., Ackermann, P., Mitsos, A., Dahmen, M., Pischinger, S., 2024. Fuel ignition delay maps for molecularly controlled combustion. Energy Fuels 38 (14), 13264–13277.

Nevolianis, T., Rittig, J.G., Mitsos, A., Leonhard, K., 2024. Multi-fidelity graph neural networks for predicting toluene/water partition coefficients. ChemRxiv preprint 10.26434/chemrxiv-2024-3t818.

Pappu, A., Paige, B., 2020. Making graph neural networks worth it for low-data molecular machine learning. arXiv preprint arXiv:2011.12203.

Qian, E., Kang, D., Sella, V., Chaudhuri, A., 2025. Multifidelity linear regression for scientific machine learning from scarce data. Found. Data Sci. 7 (1), 271–297.

Ramakrishnan, R., Dral, P.O., Rupp, M., von Lilienfeld, O.A., 2014. Quantum chemistry structures and properties of 134 kilo molecules. Sci. Data 1 (1), 140022.

Ramsundar, B., Kearnes, S., Riley, P., Webster, D., Konerding, D., Pande, V., 2015. Massively multitask networks for drug discovery. arXiv preprint arXiv:1502.02072.

Reiser, P., Neubert, M., Eberhard, A., Torresi, L., Zhou, C., Shao, C., Metni, H., van Hoesel, C., Schopmans, H., Sommer, T., Friederich, P., 2022. Graph neural networks for materials science and chemistry. Commun. Mater. 3 (1), 93.

Rittig, J.G., Felton, K.C., Lapkin, A.A., Mitsos, A., 2023. Gibbs-Duhem-Informed neural networks for binary activity coefficient prediction. Digit. Discov. 2, 1752–1767.

Rittig, J.G., Mitsos, A., 2024. Thermodynamics-consistent graph neural networks. Chem. Sci. 15, 18504–18512.

Rong, Y., Bian, Y., Xu, T., Xie, W., Wei, Y., Huang, W., Huang, J., 2020. Self-supervised graph transformer on large-scale molecular data. In: Advances in Neural Information Processing Systems. Vol. 33, Curran Associates, Inc., pp. 12559–12571.

Rosenberger, D., Barros, K., Germann, T.C., Lubbers, N., 2022. Machine learning of consistent thermodynamic models using automatic differentiation. Phys. Rev. E 105 (4), 045301.

Ruddigkeit, L., van Deursen, R., Blum, L.C., Reymond, J.-L., 2012. Enumeration of 166 billion organic small molecules in the chemical universe database GDB-17. J. Chem. Inf. Model. 52 (11), 2864–2875.

Ruder, S., 2017. An overview of multi-task learning in deep neural networks. arXiv preprint arXiv:1706.05098.

Satorras, V.G., Hoogeboom, E., Welling, M., 2021. E (n) equivariant graph neural networks. In: International Conference on Machine Learning. PMLR, pp. 9323–9332.

Schwaller, P., Vaucher, A.C., Laino, T., Reymond, J.-L., 2020. Data augmentation strategies to improve reaction yield predictions and estimate uncertainty.
Schweidtmann, A.M., Rittig, J.G., König, A., Grohe, M., Mitsos, A., Dahmen, M., 2020. Graph neural networks for prediction of fuel ignition quality. Energy Fuels 34 (9), 11395–11407.
Schweidtmann, A.M., Rittig, J.G., Weber, J.M., Grohe, M., Dahmen, M., Leonhard, K., Mitsos, A., 2023. Physical pooling functions in graph neural networks for molecular property prediction. Comput. Chem. Eng. 172, 108202.
Shorten, C., Khoshgoftaar, T.M., 2019. A survey on image data augmentation for deep learning. J. Big Data 6 (1), 1–48.
Sosnin, S., Vashurina, M., Withnall, M., Karpov, P., Fedorov, M., Tetko, I.V., 2019. A survey of multi-task learning methods in chemoinformatics. Mol. Inform. 38 (4), 1800108.
Specht, T., Nagda, M., Fellenz, S., Mandt, S., Hasse, H., Jirasek, F., 2024. HANNA: Hard-constraint neural network for consistent activity coefficient prediction. arXiv preprint arXiv:2407.18011.
Standley, T., Zamir, A., Chen, D., Guibas, L., Malik, J., Savarese, S., 2020. Which tasks should be learned together in multi-task learning? In: International Conference on Machine Learning. PMLR, pp. 9120–9132.
Stokes, J.M., Yang, K., Swanson, K., Jin, W., Cubillos-Ruiz, A., Donghia, N.M., MacNair, C.R., French, S., Carfrae, L.A., Bloom-Ackermann, Z., Tran, V.M., Chiappino-Pepe, A., Badran, A.H., Andrews, I.W., Chory, E.J., Church, G., Brown, E.D., Jaakkola, T., Barzilay, R., Collins, J.J., 2020. A deep learning approach to antibiotic discovery. Cell 180 (4), 688–702.
Ulrich, N., Goss, K.-U., Ebert, A., 2021. Exploring the octanol–water partition coefficient dataset using deep learning techniques and data augmentation. Commun. Chem. 4 (1), 90.
de la Vega de León, A., Chen, B., Gillet, V.J., 2018. Effect of missing data on multitask prediction methods. J. Cheminformatics 10, 1–12.
Vermeire, F.H., Green, W.H., 2021. Transfer learning for solvation free energies: From quantum chemistry to experiments. Chem. Eng. J. 418, 129307.
Wang, H., Kaddour, J., Liu, S., Tang, J., Lasenby, J., Liu, Q., 2024. Evaluating self-supervised learning for molecular graph embeddings. Adv. Neural Inf. Process. Syst. 36.
Wei, J., Zou, K., 2019. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. arXiv preprint arXiv:1901.11196.
Winter, B., Rehner, P., Esper, T., Schilling, J., Bardow, A., 2023a. Understanding the language of molecules: Predicting pure component parameters for the PC-SAFT equation of state from SMILES. arXiv preprint arXiv:2309.12404.
Winter, B., Winter, C., Esper, T., Schilling, J., Bardow, A., 2023b. SPT-NRTL: A physics-guided machine learning model to predict thermodynamically consistent activity coefficients. Fluid Phase Equilib. 568, 113731.
Yang, X., Duan, Y., Cheng, Z., Li, K., Liu, Y., Zeng, X., Cao, D., 2024. MPCD: A multitask graph transformer for molecular property prediction by integrating common and domain knowledge. J. Med. Chem. 67 (23), 21303–21316.
Yu, T., Kumar, S., Gupta, A., Levine, S., Hausman, K., Finn, C., 2020. Gradient surgery for multi-task learning. Adv. Neural Inf. Process. Syst. 33, 5824–5836.
Zang, X., Zhao, X., Tang, B., 2023. Hierarchical molecular graph self-supervised learning for property prediction. Commun. Chem. 6 (1), 34.
Zhang, Z., Liu, Q., Wang, H., Lu, C., Lee, C.-K., 2021. Motif-based graph self-supervised learning for molecular property prediction. In: Advances in Neural Information Processing Systems. Vol. 34, Curran Associates, Inc., pp. 15870–15882.
Zhang, Y., Yang, Q., 2021. A survey on multi-task learning. IEEE Trans. Knowl. Data Eng. 34 (12), 5586–5609.
Zhou, J., Yang, Y., Mroz, A.M., Jelfs, K.E., 2025. Polycl: contrastive learning for polymer representation learning via explicit and implicit augmentations. Digit. Discov..
Zubatyuk, R., Smith, J.S., Leszczynski, J., Isayev, O., 2019. Accurate and transferable multitask prediction of chemical properties with an atoms-in-molecules neural network. Sci. Adv. 5 (8), eaav6490.