ELECTROLYTEFM: UNIFYING ELECTROLYTE PROPERTY PREDICTION THROUGH CROSS-PROPERTY KNOWLEDGE LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Electrolyte formulation design requires balancing multiple physicochemical properties, yet existing models often focus on a limited subset. Learning each property in isolation can overlook transferable chemical information, whereas indiscriminate sharing can introduce cross-property interference. Our directed transfer analysis shows that jointly learning two property prediction tasks can improve or degrade prediction relative to separate training, with asymmetric transfer effects between the tasks. We propose ElectrolyteFM, a unified multi-property prediction model which can more accurately predict multiple properties of each electrolyte by effectively identifying and utilizing property-specific features and knowledge shared across properties. More specifically, ElectrolyteFM learns property-specific representations independently and captures cross-property knowledge through a separately trained expert pool. A router selects relevant shared information for each formulation and target property, and property-specific residual adapters convert this information into corrections to the corresponding representation for prediction. Experiments on Electrolyte12 show that ElectrolyteFM reduces normalized mean absolute error averaged across 12 electrolyte properties by 14.8% relative to the strongest electrolyte-specific baseline. On an independent sodium-electrolyte dataset unseen during training, it reduces conductivity mean absolute error by 6.7% relative to the best-performing baseline.

1 INTRODUCTION

Electrolytes govern ion transport, interfacial stability, and other coupled processes that determine the performance, lifetime, and safety of rechargeable batteries. Designing high-performing electrolyte formulations for practical battery applications is a multi-objective optimization problem over a combinatorial space of salts, solvents, compositions, and operating conditions, requiring a balance among multiple performance objectives. Deep-learning-based property prediction offers an efficient way to evaluate candidate formulations and guide the search toward those that meet application-specific performance requirements (Zhang et al., 2024; Li et al., 2025; Yang et al., 2026; Wang & You, 2026).

Existing electrolyte predictors often focus on a limited set of properties, such as conductivity, providing incomplete information for formulation selection (Li et al., 2025; Wang & You, 2026). Electrolyte formulations are characterized not by a single property, but by a family of observables, including transport, solvation, thermodynamic, and electrochemical properties, which jointly determine their practical behavior. These properties arise from the same underlying formulation, molecular interactions, composition, and operating conditions, suggesting that they should not be modeled as isolated prediction problems. A unified predictive model that learns across heterogeneous electrolyte properties could reuse information across datasets, improve data efficiency, and provide a more comprehensive representation of electrolyte behavior.

However, a common physical origin does not imply uniform transferability across properties. Different observables reflect distinct physical processes and therefore may benefit differently from cross-property supervision. To characterize this structure, we systematically measure directed transfer among 12 electrolyte properties by evaluating how supervision from each source property affects the prediction of every target property. As shown in Fig. 1, 61 of the 132 directed transfer scores

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

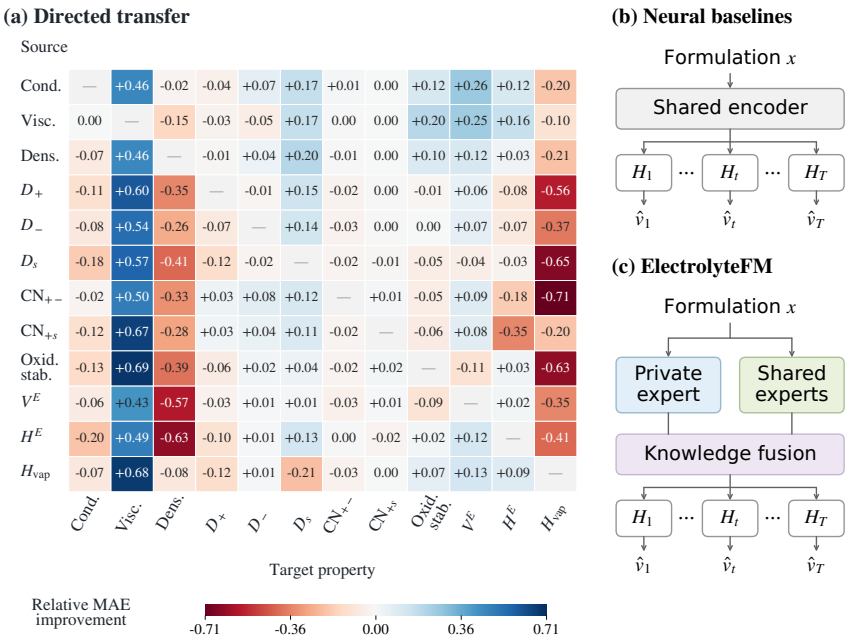


Figure 1: Cross-property transfer and sharing strategies. **(a)** Directed transfer scores on the test set used in Table 1 (rows: sources; columns: targets). Off-diagonal entries show relative target-MAE reductions; positive/negative values indicate beneficial transfer/interference. The score is defined in Section 3.1. **(b)** Shared-representation prediction with property-specific heads. **(c)** ElectrolyteFM fuses private and shared representations before prediction through property-specific heads. Abbreviations: conductivity (Cond.), viscosity (Visc.), density (Dens.), oxidation stability (Oxid. stab.), and coordination number (CN).

are positive and 71 are negative. Moreover, these effects are strongly asymmetric: knowledge that benefits one property may provide little benefit, or even cause interference, in the reverse direction.

These observations motivate a model that preserves property-specific knowledge while selectively incorporating shared information. To this end, we propose **ElectrolyteFM**, which learns private and shared representations separately and then adapts shared knowledge to complement each property's private representation, enabling accurate prediction across a broad range of electrolyte properties.

Specifically, ElectrolyteFM adopts a **Shared–Specific Knowledge Learning** framework comprising three components. Specific Knowledge Learning (SpKL) captures property-specific predictive patterns with an independent private expert for each property. Shared Knowledge Learning (ShKL) learns reusable cross-property knowledge through a shared expert pool, with formulation–property conditioned routing selecting relevant shared information. Shared-to-Specific Knowledge Fusion (S2SKF) transforms the routed shared information into complementary corrections to the private representations through property-specific residual adapters.

Our contributions are threefold:

- **Directed Cross-Property Transfer Analysis.** We systematically characterize directed transfer among 12 electrolyte properties, revealing beneficial, harmful, and asymmetric interactions that provide an empirical basis for selective cross-property knowledge sharing.
- **Shared–Specific Knowledge Learning.** We propose ElectrolyteFM, which combines independently learned property-specific representations with shared representations from a separately trained expert pool through formulation–property conditioned routing and residual fusion.
- **Multi-property evaluation and external generalization.** We curate Electrolyte12 and evaluate multi-property prediction, learning strategies, and knowledge-branch contributions, together with zero-shot generalization and few-shot adaptation on an independent sodium-electrolyte dataset.

Across 12 electrolyte properties, ElectrolyteFM reduces macro normalized mean absolute error (NMAE) by 14.8% relative to the strongest electrolyte-specific baseline. On an independent sodium-electrolyte dataset, it reduces conductivity MAE by 6.7% relative to the best-performing baseline evaluated on that dataset, with further gains from few-shot adaptation.

2 Related Work

2.1 Learning for Electrolyte Formulations

Learning-based electrolyte modeling builds on molecular pretraining and formulation-level representations. Molecular pretraining methods, including MolT5 and Uni-Mol, learn transferable representations from molecular strings, graphs, language, and three-dimensional structures (Edwards et al., 2022; Xia et al., 2023; Zhou et al., 2023; Ji et al., 2024). At the formulation level, MolSets, GeoMix, and SCAN explicitly model mixture composition and component interactions for electrolyte property prediction (Zhang et al., 2024; Li et al., 2025; Wang & You, 2026). DiffMix and related mixture models incorporate thermodynamic structure into property prediction (Zhu et al., 2024; Specht et al., 2024), while BAMBOO connects microscopic interactions with macroscopic electrolyte properties through learned interatomic potentials and molecular dynamics (Gong et al., 2025). Uni-ELF and Bamboo-Mixer further explore formulation-level pretraining and unified predictive–generative modeling, respectively, to support property prediction and formulation design (Zeng et al., 2024; Yang et al., 2026).

These studies improve molecular and formulation representations, but such advances alone do not resolve how heterogeneous property supervision should be shared. Joint learning can still benefit some targets while interfering with others. ElectrolyteFM combines property-specific learning with selective sharing to support unified multi-property prediction.

2.2 Cross-Task Transfer Learning

Cross-task transfer learning seeks to reuse knowledge across prediction tasks, but sharing can also introduce negative transfer (Liu et al., 2019). Taskonomy characterizes directed transfer between visual tasks, while subsequent multi-task studies examine cooperation and competition under joint learning (Zamir et al., 2018; Standley et al., 2020). Task affinities and predicted task-combination gains further guide task grouping and selective group updates (Fifty et al., 2021; Song et al., 2022; Jeong & Yoon, 2025). At the optimization level, PCGrad and CAGrad address conflicting task gradients, while Nash-MTL and FAMO balance task contributions through bargaining-based aggregation and adaptive loss weighting, respectively (Yu et al., 2020; Liu et al., 2021; Navon et al., 2022; Liu et al., 2023). In scientific prediction, cross-property transfer and source-selection strategies exploit data-rich properties and estimated task similarities to support data-scarce targets (Gupta et al., 2021; Li et al., 2022; Yao et al., 2024). Adaptive checkpointing and specialization further address negative transfer in molecular property learning (Eraqi et al., 2025).

Task selection and grouping determine which tasks share supervision, while gradient coordination regulates joint optimization. These mechanisms do not directly specify how shared knowledge contributes to an individual formulation–property prediction. Motivated by directed transfer analysis, ElectrolyteFM conditions expert allocation on both the formulation and target property to select relevant shared information.

2.3 Mixture-of-Experts for Knowledge Sharing

Mixture-of-experts architectures support selective knowledge sharing through learned combinations of expert representations. MMoE uses task-specific gates over shared experts, PLE explicitly separates shared and task-specific experts, and TaskExpert dynamically assembles task-specific features from multiple expert representations (Ma et al., 2018; Tang et al., 2020; Ye & Xu, 2023). Related architectures regulate sharing through selective layer specialization, orthogonal expert representations, or modality-specific model spaces (Shi et al., 2023; Hendawy et al., 2024; Peng et al., 2025). Another line of work reuses pretrained task modules: PEMT and MeteoRA selectively combine task-specific adapters in language models (Lin et al., 2024; Xu et al., 2025), while materials expert-composition methods and MoMa combine source-property experts or specialized modules for downstream prediction (Chang et al., 2022; Wang et al., 2026).

MMoE and PLE control expert allocation through routing, but task-specific representations remain coupled to joint optimization. Source-expert composition instead adapts separately pretrained modules to downstream targets. ElectrolyteFM learns property-specific private representations and a shared expert pool separately, then freezes both branches for fusion. Property-specific residual adapters convert routed shared information into corrections to the private representations.

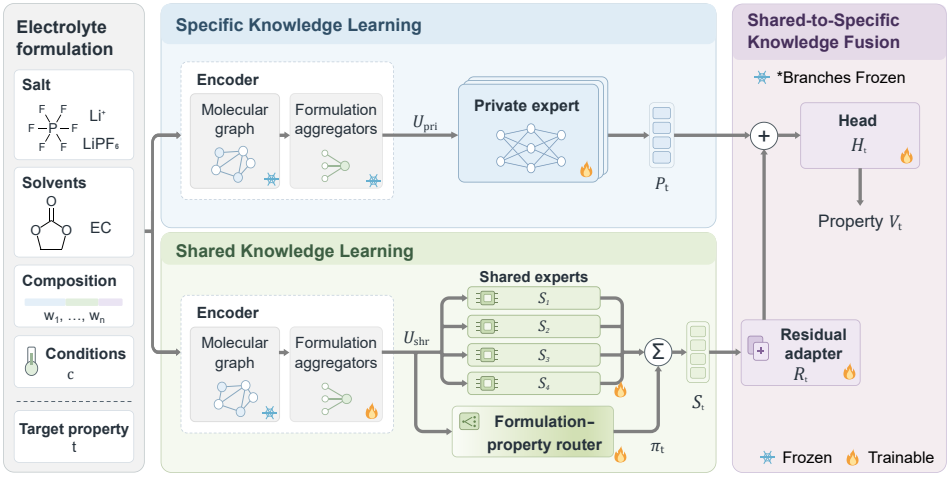


Figure 2: Overview of ElectrolyteFM. In Specific Knowledge Learning (SpKL), private experts learn property-specific knowledge. In Shared Knowledge Learning (ShKL), a shared expert pool learns reusable cross-property knowledge, and formulation–property conditioned routing combines shared expert outputs according to the input formulation and target property. Shared-to-Specific Knowledge Fusion uses property-specific residual adapters to incorporate the routed shared information into private representations, with both learned branches frozen. Snowflakes and flames indicate frozen and trainable modules, respectively.

3 Method

3.1 Directed Cross-Property Transfer Analysis

We first characterize how supervision from one electrolyte property affects the prediction of another. For each ordered source–target pair (a, t), we compare a target-only model trained on property t with a pairwise model jointly trained on properties a and t. Both models use the same architecture and initialization, consisting of a shared backbone and property-specific prediction heads, as illustrated in Figure 1(b).

To quantify the directional effect of source property a on target property t, we define the directed transfer score as the relative reduction in target mean absolute error (MAE):

$$\Delta_{a\to t} = \frac{\mathrm{MAE}_t^{\mathrm{single}} - \mathrm{MAE}_t^{\mathrm{pair}(a,t)}}{\mathrm{MAE}_t^{\mathrm{single}}}. \tag{1}$$

Both models are evaluated on the same test samples of the target property, with MAE computed in physical units. A positive $\Delta_{a\to t}$ indicates that supervision from property a improves prediction of property t, whereas a negative value indicates cross-property interference.

As shown in Figure 1(a), cross-property transfer is highly heterogeneous and directional: the same source property can benefit some targets while interfering with others, and the transfer effect between two properties is generally asymmetric. For example, adding conductivity supervision reduces solvent diffusivity MAE by 17.2%, whereas adding solvent diffusivity supervision increases conductivity MAE by 17.9%. These observations motivate ElectrolyteFM to preserve independent property-specific representations while selectively incorporating shared knowledge according to the target property and input formulation.

3.2 Problem Formulation and Overview

Let $\mathcal{T} = \{1, \ldots, T\}$ denote the set of electrolyte properties, with $T = 12$ in this work. Each formulation x_i is defined by its molecular components, composition, and experimental conditions. Since property labels are partially observed, the supervision for property t is

$$\mathcal{D}_t = \{(x_i, y_{it}) \mid m_{it} = 1\}, \tag{2}$$

where m_{it} indicates label availability. Our goal is to learn a unified predictor

$$\widehat{y}_{it} = f(x_i, t) \tag{3}$$

that captures both property-specific and cross-property knowledge.

Figure 2 illustrates ElectrolyteFM. Separate copies of the molecular and formulation encoders produce private and shared representations $\mathbf{u}_i^{\mathrm{pri}}$ and $\mathbf{u}_i^{\mathrm{shr}}$. Specific Knowledge Learning (SpKL) uses a property-specific expert P_t to obtain $\mathbf{p}_{it}$, while Shared Knowledge Learning (ShKL) combines shared expert outputs using formulation–property conditioned routing weights $\boldsymbol{\pi}_{it}$ to obtain $\mathbf{s}_{it}$. Shared-to-Specific Knowledge Fusion (S2SKF) then uses a property-specific residual adapter R_t to convert $\mathbf{s}_{it}$ into a correction to $\mathbf{p}_{it}$, followed by the prediction head H_t.

All objectives are defined on transformed and standardized targets v_{it}, using mean squared error in the standardized space (*z-MSE*).

3.3 Specific Knowledge Learning

Specific Knowledge Learning captures property-specific predictive patterns through independent experts. For a query (x_i, t), the corresponding private expert P_t transforms $\mathbf{u}_i^{\mathrm{pri}}$ into a private state $\mathbf{p}_{it}$, and its head H_t^{pri} produces the standardized prediction:

$$\mathbf{p}_{it} = P_t(\mathbf{u}_i^{\mathrm{pri}}), \qquad \widehat{v}_{it}^{\mathrm{pri}} = H_t^{\mathrm{pri}}(\mathbf{p}_{it}). \tag{4}$$

Each P_t is a multilayer perceptron (MLP) producing a private state $\mathbf{p}_{it} \in \mathbb{R}^{d_h}$, where d_h is the hidden dimension. Each property has its own expert and scalar prediction head.

We train each expert and its head only on $\mathcal{D}_t$, minimizing $\mathcal{L}_t^{\mathrm{pri}} = \mathbb{E}_{\mathcal{D}_t}[\ell(\widehat{v}_{it}^{\mathrm{pri}}, v_{it})]$, with the complete encoder frozen.

3.4 Shared Knowledge Learning

Shared Knowledge Learning captures transferable patterns across properties using a pool of K shared experts $\{S_k\}_{k=1}^{K}$. To account for the heterogeneous relevance of shared knowledge, we introduce a formulation–property conditioned router:

$$\boldsymbol{\pi}_{it} = \mathrm{softmax}\big(W_r \mathbf{u}_i^{\mathrm{shr}} + \mathbf{b}_t\big), \tag{5}$$

where $W_r \in \mathbb{R}^{K \times d_u}$ is a bias-free linear projection. The learned vector $\mathbf{b}_t \in \mathbb{R}^K$ expresses the property's preference over experts, while $W_r \mathbf{u}_i^{\mathrm{shr}}$ adjusts that preference for the input formulation. Each expert receives the same shared input, and its contribution is weighted by the corresponding router output:

$$\mathbf{s}_{it} = \sum_{k=1}^{K} \pi_{itk} S_k(\mathbf{u}_i^{\mathrm{shr}}), \qquad \widehat{v}_{it}^{\mathrm{shr}} = H_t^{\mathrm{shr}}(\mathbf{s}_{it}). \tag{6}$$

A property-specific ShKL prediction head H_t^{shr} maps the routed shared representation $\mathbf{s}_{it} \in \mathbb{R}^{d_h}$ to the standardized prediction and provides supervision to the shared experts and router.

ShKL draws a property with probability $q(t) \propto |\mathcal{D}_t|^{\alpha}$, with $0 < \alpha < 1$ to moderate the influence of data-rich properties, and then samples a batch from its labeled set, optimizing

$$\mathcal{L}^{\mathrm{shr}} = \mathbb{E}_{t \sim q} \mathbb{E}_{(x_i, y_{it}) \sim \mathcal{D}_t} \left[\ell(\widehat{v}_{it}^{\mathrm{shr}}, v_{it}) \right]. \tag{7}$$

The neutral and ionic formulation aggregators, shared experts, router, and ShKL prediction heads are trained jointly, with the molecular graph module fixed. We retain the learned shared encoder, experts, and router to produce $\mathbf{s}_{it}$ for S2SKF. These prediction heads are used only during ShKL.

3.5 SHARED-TO-SPECIFIC KNOWLEDGE FUSION

Shared-to-Specific Knowledge Fusion (S2SKF) integrates transferable shared knowledge into the independently learned specific representation. For each property t, a property-specific residual adapter R_t transforms the routed shared representation $\mathbf{s}_{it}$ into a complementary residual:

$$\mathbf{z}_{it} = \mathbf{p}_{it} + R_t(\mathbf{s}_{it}), \qquad \widehat{v}_{it} = H_t(\mathbf{z}_{it}), \tag{8}$$

where $\mathbf{p}_{it}$ denotes the specific representation learned for property t, and $\mathbf{z}_{it}$ is the fused representation. The adapter adopts a bottleneck architecture:

$$R_t(\mathbf{s}_{it}) = W_t^{\mathrm{up}} \,\mathrm{SiLU}\big(W_t^{\mathrm{down}} \,\mathrm{LN}(\mathbf{s}_{it}) + \mathbf{a}_t^{\mathrm{down}}\big) + \mathbf{a}_t^{\mathrm{up}}, \tag{9}$$

where d_h is the representation dimension and $d_b < d_h$ is the bottleneck dimension.

We zero-initialize the up-projection W_t^{up} and $\mathbf{a}_t^{\mathrm{up}}$, and initialize H_t from the corresponding specific prediction head. The fused model therefore starts from the learned specific predictor and gradually incorporates useful shared information. During fusion, the specific and shared branches are frozen, and only R_t and H_t are optimized:

$$\mathcal{L}_t^{\mathrm{fus}} = \mathbb{E}_{(x_i, y_{it}) \sim \mathcal{D}_t} \left[\ell(\widehat{v}_{it}, v_{it}) \right]. \tag{10}$$

This formulation preserves property-specific predictive knowledge while allowing transferable cross-property information to selectively complement it. Architecture and optimization details are provided in Appendix B.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Datasets and settings. We evaluate all methods on **Electrolyte12**, a large-scale electrolyte dataset curated in this work, covering 12 electrolyte properties. The training and validation sets contain both experimental and computed labels, comprising 366,825 and 44,811 records, respectively. The test set contains 43,709 records, including experimental measurements for conductivity, viscosity, excess molar enthalpy, and excess molar volume, and computed labels for the remaining eight properties. All methods use the same training, validation, and test splits.

Baselines. We compare ElectrolyteFM with RDKit-XGB (Landrum et al., 2024; Chen & Guestrin, 2016), MolT5-XGB (Edwards et al., 2022; Chen & Guestrin, 2016), Bamboo-Mixer (Yang et al., 2026), GeoMix (Li et al., 2025), MolSets (Zhang et al., 2024), and SCAN (Wang & You, 2026). All methods follow the same data and target preprocessing, including target transformations and missing-label handling. Dataset and implementation details are in Appendices A and B, respectively.

Evaluation metrics. We use mean absolute error (MAE) as the primary metric for evaluating individual electrolyte properties, with errors reported in their original physical units. To summarize performance across properties with different numerical scales, we additionally report macro normalized MAE (NMAE), defined as $\frac{1}{|G|}\sum_{t\in G} \mathrm{MAE}_t/\sigma_t^{\mathrm{train}}$, where G denotes the set of evaluated properties and $\sigma_t^{\mathrm{train}}$ is the standard deviation of property t in the training set. We further report Pearson correlation coefficient (PCC) and coefficient of determination (R^2).

4.2 MULTI-PROPERTY PREDICTION

Table 1 compares the methods on Electrolyte12 in terms of per-property MAE, macro NMAE across all twelve properties, and the number of properties on which each method achieves the best MAE. Results are grouped according to whether the test labels are obtained experimentally or computationally.

ElectrolyteFM achieves the best MAE on 10 of the 12 properties, including all four properties with experimental labels and six of the eight properties with computed labels. It also obtains the lowest macro NMAE of 0.1571, compared with 0.1845 for the strongest baseline. The improvements are particularly pronounced on the experimentally measured properties: relative to the strongest baseline for each property, ElectrolyteFM reduces MAE by 50.1% for conductivity, 27.2% for viscosity, 67.8% for excess molar enthalpy, and 26.7% for excess molar volume. Among the computed properties, it further reduces MAE by 31.3% for density and 30.3% for heat of vaporization.

Table 1: Per-property MAE on Electrolyte12 (lower is better). Blocks denote experimental and computed test labels. Bold and underline indicate the best and second-best unrounded values. The macro and best-count rows summarize all twelve properties.

Property	RDKit-XGB (Landrum et al. 2024)	MolT5-XGB (Edwards et al. 2022)	Bamboo-Mixer (Yang et al. 2026)	GeoMix (Li et al. 2025)	MolSets (Zhang et al. 2024)	SCAN (Wang and You 2026)	**ElectrolyteFM (Ours)**
Experimental labels							
Cond. (mS/cm)	4.444	4.211	<u>1.766</u>	3.211	1.980	2.335	**0.881**
Visc. (mPa·s)	156.457	156.654	<u>125.313</u>	147.831	139.246	148.301	**91.187**
H^E (J/mol)	311.071	299.599	<u>167.964</u>	374.825	191.577	248.297	**54.092**
V^E (cm^3/mol)	0.140	0.136	<u>0.068</u>	0.181	0.111	0.142	**0.050**
Computed labels							
Dens. (g/cm^3)	0.0566	0.0506	<u>0.0079</u>	0.0296	0.0175	0.0321	**0.0054**
D_+ (10^{-11} m^2/s)	8.911	8.204	4.295	4.659	**4.102**	<u>4.114</u>	4.144
D_- (10^{-11} m^2/s)	8.785	7.806	3.897	4.095	3.846	<u>3.769</u>	**3.769**
D_s (10^{-11} m^2/s)	22.227	19.611	10.745	9.968	<u>9.130</u>	9.949	**8.843**
CN_{+-}	1.127	1.079	<u>0.948</u>	0.949	0.962	0.966	**0.946**
CN_{+s}	2.084	2.066	1.851	<u>1.799</u>	1.885	1.912	**1.788**
Oxid. stab. (eV)	0.517	0.555	**0.210**	0.638	0.389	0.518	<u>0.217</u>
H_{vap} (J/mol)	4242.9	4790.6	<u>684.9</u>	2735.9	2569.0	2409.8	**477.2**
Macro NMAE	0.3643	0.3536	<u>0.1845</u>	0.2936	0.2279	0.2592	**0.1571**
Best count	0/12	0/12	<u>1/12</u>	0/12	<u>1/12</u>	0/12	**10/12**

4.3 Generalization to Unseen Formulations

We further evaluate generalization on an independent experimental dataset of 106 multicomponent sodium-electrolyte formulations, including mixtures with up to 14 components (Muñoz-Perales et al., 2026).

We evaluate conductivity prediction under two settings: zero-shot generalization, where the trained model is directly applied to the unseen sodium formulations, and few-shot adaptation, where a small number of target-domain labels are provided for adaptation.

4.3.1 Zero-shot Generalization

Table 2 compares conductivity prediction on the experimental in-distribution (ID) subset and the independent out-of-distribution (OOD) dataset.

Table 2: Conductivity prediction on experimental ID and independent OOD data. MAE is in mS/cm.

	Experimental ID $N = 7{,}153$			OOD $N = 106$		
Method	MAE ↓	PCC ↑	R^2 ↑	MAE ↓	PCC ↑	R^2 ↑
SCAN	2.335	0.856	0.723	<u>2.778</u>	<u>0.775</u>	<u>0.599</u>
MolSets	1.980	0.899	0.795	7.369	0.201	-1.839
GeoMix	3.211	0.771	0.574	3.321	0.685	0.208
Bamboo-Mixer	<u>1.766</u>	<u>0.909</u>	<u>0.822</u>	3.131	0.582	0.295
RDKit-XGB	4.444	0.729	0.274	3.624	0.349	-0.228
MolT5-XGB	4.211	0.748	0.303	3.317	0.733	0.041
ElectrolyteFM (Ours)	**0.881**	**0.942**	**0.887**	**2.592**	**0.791**	**0.605**

The baseline ranking changes under distribution shift: Bamboo-Mixer outperforms SCAN on the ID test set but falls behind it on the OOD test set. Higher ID accuracy therefore does not necessarily translate into better prediction on unseen formulations. ElectrolyteFM ranks first across all three metrics in both settings, reducing OOD MAE by 6.7% relative to SCAN, the strongest OOD baseline. Its predictive advantage thus extends to the independent sodium-electrolyte dataset without target-domain adaptation.

4.3.2 Few-shot Adaptation

We evaluate affine calibration and final-layer adaptation over 20 fixed support/query episodes with nested support sizes $K \in \{0, 1, 2, 4, 8, 16\}$. Figure 3 compares six methods from Table 1 under affine calibration and four neural methods under final-layer adaptation using query z-MSE. Each episode uses 74 query formulations, held fixed across support sizes; Appendix E.2 provides the protocol.

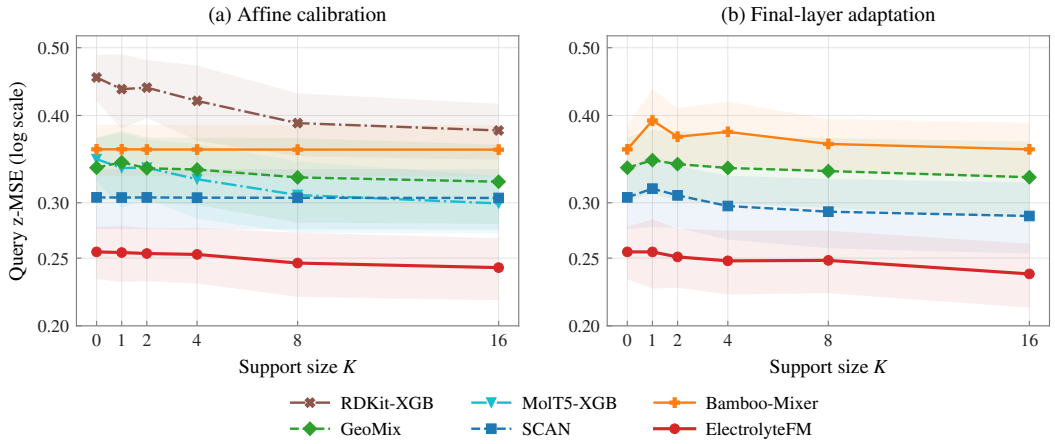


Figure 3: Conductivity OOD few-shot adaptation for six methods from Table 1. Curves show mean query z-MSE on logarithmic y-axes; shaded regions show the sample standard deviation over 20 fixed episodes. XGBoost models are evaluated under affine calibration only. MolSets is omitted because some values exceed the plotted y-axis range.

ElectrolyteFM achieves the lowest mean query z-MSE at every evaluated support size under both protocols. With 16 labeled formulations and the learned representations fixed, affine calibration and final-layer adaptation reduce its mean query z-MSE from 0.2552 to 0.2423 and 0.2373, respectively.

4.4 Ablation Studies

4.4.1 Comparison of Learning Strategies

To assess how different learning strategies affect multi-property prediction, we compare ElectrolyteFM with single-task learning (STL), multi-gate mixture-of-experts (MMoE) (Ma et al., 2018), and progressive layered extraction (PLE) (Tang et al., 2020). Table 3 reports the macro metrics across all twelve properties, with equal weight assigned to each property. All methods start from the same pretrained molecular encoders. STL learns each property independently, whereas MMoE and PLE learn jointly across properties; ElectrolyteFM separates representation learning from subsequent target-specific fusion.

Both multi-task baselines outperform STL on all three macro metrics, demonstrating the benefit of cross-property learning in this comparison. However, the label-source breakdown in Table 11 shows that MMoE and PLE improve all three macro metrics over STL for the experimental-label group, but worsen them for the computed-label group. ElectrolyteFM improves all three metrics over STL in both groups. ElectrolyteFM further reduces macro NMAE by 2.5% relative to PLE while achieving the highest macro PCC and R^2. ElectrolyteFM combines decoupled knowledge learning with shared-to-specific fusion, using shared information to complement independently learned property-specific representations. This combination achieves better aggregate predictive performance than either independent learning or the two joint-learning alternatives.

Table 3: Learning strategies: macro metrics over all 12 properties.

Method	NMAE ↓	PCC ↑	R^2 ↑
STL	0.1653	0.8931	0.8071
MMoE	0.1641	0.9058	0.8220
PLE	<u>0.1611</u>	<u>0.9228</u>	<u>0.8228</u>
ElectrolyteFM	**0.1571**	**0.9268**	**0.8357**

Table 4: Knowledge learning and fusion ablation: macro NMAE over all 12 properties.

SpKL	ShKL	S2SKF	NMAE ↓
✓	–	–	0.1841
–	✓	–	<u>0.1674</u>
✓	✓	✓	**0.1571**

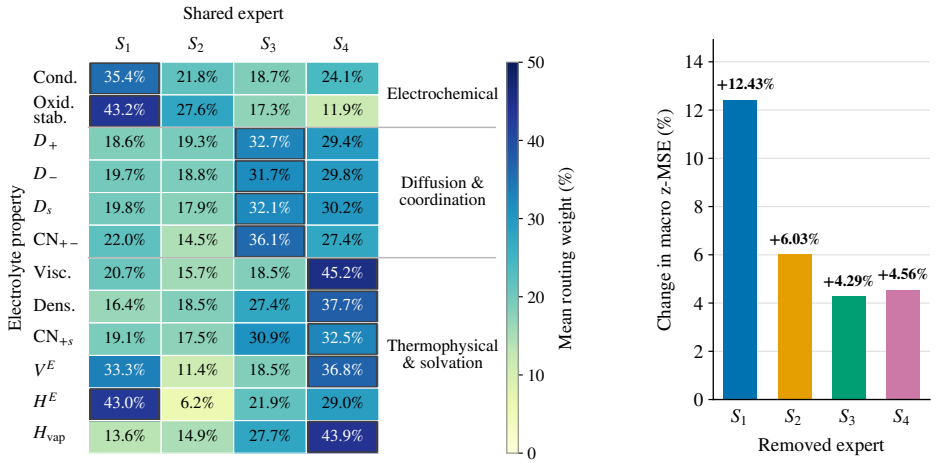
4.4.2 Contribution of Knowledge Learning and Fusion

We compare the private branch learned through SpKL, the shared branch learned through ShKL, and their combination through S2SKF. Table 4 reports macro NMAE over all twelve properties. Unlike

STL, SpKL keeps the formulation aggregators fixed. The shared branch alone outperforms the private branch, while the complete model reduces macro NMAE by 14.7% and 6.1% relative to the private and shared branches, respectively.

The improvement over the shared branch alone indicates that the independently learned private representation provides complementary predictive information. S2SKF uses this private representation as a foundation and learns a property-specific residual correction from the routed shared representation. The lower macro NMAE relative to either branch alone supports combining property-specific and shared knowledge through shared-to-specific fusion.

4.4.3 Expert Allocation and Predictive Contributions



(a) (b)

Figure 4: Expert allocation and contribution in ElectrolyteFM. (a) Mean routing weights by property; boxes mark each row maximum. (b) Relative increase in macro z-MSE after removing each expert at inference and renormalizing the remaining weights. All twelve properties receive equal weight.

Figure 4(a) shows mean routing weights by property. S_1 receives the largest weight for conductivity, H^E, and oxidation stability; S_3 for D_+, D_-, D_s, and CN_{+-}; and S_4 for viscosity, V^E, density, CN_{+s}, and H_{vap}. These preferences suggest property-dependent use of the shared representations. Multiple experts receive substantial weight for each property, allowing their representations to contribute jointly to prediction.

We assess each expert's predictive contribution by removing it without retraining and renormalizing the remaining routing weights. Figure 4(b) shows the resulting changes in macro z-MSE. Across the twelve equally weighted properties, removing S_1–S_4 changes macro z-MSE by +12.43%, +6.03%, +4.29%, and +4.56%, respectively. Removing any expert increases the error, with the largest effect for S_1. S_2 also contributes despite receiving no property's largest mean routing weight, illustrating why allocation weights alone do not measure predictive importance.

5 Conclusion

Motivated by heterogeneous cross-property transfer, ElectrolyteFM learns private and shared representations separately and integrates them through property-specific residual adapters, enabling accurate prediction across a broad range of electrolyte properties. ElectrolyteFM achieves the lowest macro NMAE and the lowest MAE on 10 of the 12 properties. Conductivity OOD and few-shot results further support its use for prediction and adaptation under formulation shift.

Limitations. The present study focuses on liquid electrolyte formulations. Extending the framework to solid-state electrolytes would require structure- and microstructure-aware representations. Connecting the predicted formulation properties to electrode-dependent interfacial evolution and long-term cell performance further requires interfacial characterization and cell-level validation.

Artificial intelligence (AI) use statement

Generative AI tools were used to assist with language editing, figure preparation, LaTeX formatting, and consistency checks in the manuscript. The authors are responsible for the scientific content and final version of the paper.

Reproducibility statement

Section 3 specifies the model and knowledge-learning procedure. Section 4.1 describes the evaluation protocol. Appendices A and B provide data curation, target transforms, architecture and optimization settings, and model selection procedures needed to reproduce the experiments.

References

Rees Chang, Yu-Xiong Wang, and Elif Ertekin. Towards overcoming data scarcity in materials science: Unifying models and datasets with a mixture of experts framework. *npj Computational Materials*, 8:242, 2022.

Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794. ACM, 2016.

Paolo de Blasio, Jonas Elsborg, Tejs Vegge, Eibar Flores, and Arghya Bhowmik. CALiSol-23: Experimental electrolyte conductivity data for various Li-salts and solvent combinations. *Scientific Data*, 11:750, 2024.

Qian Dong, Chris D. Muzny, Andrei Kazakov, Vladimir Diky, Joseph W. Magee, Jason A. Widegren, Robert D. Chirico, Kenneth N. Marsh, and Michael Frenkel. ILThermo: A free-access web database for thermodynamic properties of ionic liquids. *Journal of Chemical & Engineering Data*, 52(4):1151–1159, 2007.

Carl Edwards, Tuan Lai, Kevin Ros, Garrett Honke, Kyunghyun Cho, and Heng Ji. Translation between molecules and natural language. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 375–413. Association for Computational Linguistics, 2022.

Basem A. Eraqi, Dmitrii Khizbullin, Shashank S. Nagaraja, and S. Mani Sarathy. Molecular property prediction in the ultra-low data regime. *Communications Chemistry*, 8:201, 2025.

Chris Fifty, Ehsan Amid, Zhe Zhao, Tianhe Yu, Rohan Anil, and Chelsea Finn. Efficiently identifying task groupings for multi-task learning. In *Advances in Neural Information Processing Systems*, volume 34, pp. 27503–27516. Curran Associates, Inc., 2021.

Sheng Gong, Yumin Zhang, Zhenliang Mu, Zhichen Pu, Hongyi Wang, Xu Han, Zhiao Yu, Mengyi Chen, Tianze Zheng, Zhi Wang, Lifei Chen, Zhenze Yang, Xiaojie Wu, Shaochen Shi, Weihao Gao, Wen Yan, and Liang Xiang. A predictive machine learning force-field framework for liquid electrolyte development. *Nature Machine Intelligence*, 7(4):543–552, 2025.

Vishu Gupta, Kamal Choudhary, Francesca Tavazza, Carelyn E. Campbell, Wei-Keng Liao, Alok Choudhary, and Ankit Agrawal. Cross-property deep transfer learning framework for enhanced predictive analytics on small materials data. *Nature Communications*, 12:6595, 2021.

Ahmed Hendawy, Jan Peters, and Carlo D'Eramo. Multi-task reinforcement learning with mixture of orthogonal experts. In *International Conference on Learning Representations*, 2024.

Wooseong Jeong and Kuk-Jin Yoon. Selective task group updates for multi-task optimization. In *International Conference on Learning Representations*, 2025.

Xiaohong Ji, Zhen Wang, Zhifeng Gao, Hang Zheng, Linfeng Zhang, Guolin Ke, and Weinan E. Exploring molecular pretraining model at scale. In *Advances in Neural Information Processing Systems*, volume 37, pp. 46956–46978, 2024.

Ritesh Kumar, Minh Canh Vu, Peiyuan Ma, and Chibueze V. Amanchukwu. Electrolytomics: A unified big data approach for electrolyte design and discovery. *Chemistry of Materials*, 37(8): 2720–2734, 2025. doi: 10.1021/acs.chemmater.4c03196.

Greg Landrum et al. RDKit: Open-source cheminformatics. Zenodo, 2024. URL https://doi.org/10.5281/zenodo.10793672. Version 2023.09.6.

Anyi Li, Jiacheng Cen, Songyou Li, Mingze Li, Yang Yu, and Wenbing Huang. Geometric mixture models for electrolyte conductivity prediction. In *Advances in Neural Information Processing Systems*, volume 38, pp. 13236–13264, 2025.

Han Li, Xinyi Zhao, Shuya Li, Fangping Wan, Dan Zhao, and Jianyang Zeng. Improving molecular property prediction through a task similarity enhanced transfer learning strategy. *iScience*, 25(10): 105231, 2022.

Zhisheng Lin, Han Fu, Chenghao Liu, Zhuo Li, and Jianling Sun. PEMT: Multi-task correlation guided mixture-of-experts enables parameter-efficient transfer learning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 6869–6883. Association for Computational Linguistics, 2024.

Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. In *Advances in Neural Information Processing Systems*, volume 34, pp. 18878–18890, 2021.

Bo Liu, Yihao Feng, Peter Stone, and Qiang Liu. FAMO: Fast adaptive multitask optimization. In *Advances in Neural Information Processing Systems*, volume 36, pp. 57226–57243. Curran Associates, Inc., 2023.

Shengchao Liu, Yingyu Liang, and Anthony Gitter. Loss-balanced task weighting to reduce negative transfer in multi-task learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 9977–9978, 2019.

Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H. Chi. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1930–1939. ACM, 2018.

Vanesa Muñoz-Perales, Jason K. Phong, Sokseiha Muy, Ankita Morankar, Joseph R. Geniesse, Nianhan Tian, Sawyer D. Cawthern, Fuminori Mizuno, Brian D. Storey, Jeremiah A. Johnson, and Yang Shao-Horn. Data-driven insights into ionic conductivity in high-dimensional sodium battery electrolytes. *ACS Energy Letters*, 11(8):5384–5393, 2026.

Kou Nakamura, Norio Takenaka, Masatoshi Hanai, Yuna Oikawa, Ryo Tamura, Koji Tsuda, Masanobu Nakayama, Junichiro Shiomi, and Atsuo Yamada. Open electrolyte database generated via an automated molecular dynamics simulation framework. *npj Computational Materials*, 12:215, 2026.

Aviv Navon, Aviv Shamsian, Idan Achituve, Haggai Maron, Kenji Kawaguchi, Gal Chechik, and Ethan Fetaya. Multi-task learning as a bargaining game. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 16428–16446. PMLR, 2022.

Jie Peng, Sukwon Yun, Kaixiong Zhou, Ruida Zhou, Thomas Hartvigsen, Yanyong Zhang, Zhangyang Wang, and Tianlong Chen. Sparse MoE as a new treatment: Addressing forgetting, fitting, learning issues in multi-modal multi-task learning. In *Conference on Parsimony and Learning*, volume 280 of *Proceedings of Machine Learning Research*, pp. 1112–1145. PMLR, 2025.

Ella Miray Rajaonson, Mahyar Rajabi Kochi, Luis Martin Mejia Mendoza, Mohamad Moosavi, and Benjamin Sanchez-Lengeling. CheMixHub: Datasets and benchmarks for chemical mixture property prediction. In *Advances in Neural Information Processing Systems*, volume 38. Curran Associates, Inc., 2025. doi: 10.52202/085713-1107.

Guangyuan Shi, Qimai Li, Wenlong Zhang, Jiaxin Chen, and Xiao-Ming Wu. Recon: Reducing conflicting gradients from the root for multi-task learning. In *International Conference on Learning Representations*, 2023.

Xiaozhuang Song, Shun Zheng, Wei Cao, James Yu, and Jiang Bian. Efficient and effective multi-task grouping via meta learning on task combinations. In *Advances in Neural Information Processing Systems*, volume 35, pp. 37647–37659. Curran Associates, Inc., 2022.

Thomas Specht, Mayank Nagda, Sophie Fellenz, Stephan Mandt, Hans Hasse, and Fabian Jirasek. HANNA: Hard-constraint neural network for consistent activity coefficient prediction. *Chemical Science*, 15(47):19777–19786, 2024.

Trevor Standley, Amir Zamir, Dawn Chen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. Which tasks should be learned together in multi-task learning? In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9120–9132. PMLR, 2020.

Hongyan Tang, Junning Liu, Ming Zhao, and Xudong Gong. Progressive layered extraction (PLE): A novel multi-task learning (MTL) model for personalized recommendations. In *Proceedings of the 14th ACM Conference on Recommender Systems*, pp. 269–278. ACM, 2020.

Botian Wang, Yawen Ouyang, Yaohui Li, Mianzhi Pan, Yuanhang Tang, Yiqun Wang, Haorui Cui, Jianbing Zhang, Xiaonan Wang, Wei-Ying Ma, and Hao Zhou. MoMa: A simple modular learning framework for material property prediction. In *International Conference on Learning Representations*, 2026.

Zhilong Wang and Fengqi You. A dynamic routing-guided interpretable framework for salt–solvent chemistry. *Nature Computational Science*, 6:271–284, 2026.

Jun Xia, Chengshuai Zhao, Bozhen Hu, Zhangyang Gao, Cheng Tan, Yue Liu, Siyuan Li, and Stan Z. Li. Mole-BERT: Rethinking pre-training graph neural networks for molecules. In *International Conference on Learning Representations*, 2023.

Jingwei Xu, Junyu Lai, and Yunpeng Huang. MeteoRA: Multiple-tasks embedded LoRA for large language models. In *International Conference on Learning Representations*, 2025.

Zhenze Yang, Yifan Wu, Xu Han, Ziqing Zhang, Haoen Lai, Zhenliang Mu, Tianze Zheng, Siyuan Liu, Zhichen Pu, Zhi Wang, Zhiao Yu, Sheng Gong, and Wen Yan. A unified predictive and generative solution for liquid electrolyte formulation. *Nature Machine Intelligence*, 8:186–196, 2026.

Shaolun Yao, Jie Song, Lingxiang Jia, Lechao Cheng, Zipeng Zhong, Mingli Song, and Zunlei Feng. Fast and effective molecular property prediction with transferability map. *Communications Chemistry*, 7:85, 2024.

Hanrong Ye and Dan Xu. TaskExpert: Dynamically assembling multi-task representations with memorial mixture-of-experts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 21828–21837, 2023.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 5824–5836, 2020.

Amir R. Zamir, Alexander Sax, William Shen, Leonidas J. Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3712–3722, 2018.

Boshen Zeng, Sian Chen, Xinxin Liu, Changhong Chen, Bin Deng, Xiaoxu Wang, Zhifeng Gao, Yuzhi Zhang, Weinan E, and Linfeng Zhang. Uni-ELF: A multi-level representation learning framework for electrolyte formulation design. *arXiv preprint arXiv:2407.06152*, 2024.

Hengrui Zhang, Tianxing Lai, Jie Chen, Arumugam Manthiram, James M. Rondinelli, and Wei Chen. Learning molecular mixture property using chemistry-aware graph neural network. *PRX Energy*, 3 (2):023006, 2024.

Gengmo Zhou, Zhifeng Gao, Qiankun Ding, Hang Zheng, Hongteng Xu, Zhewei Wei, Linfeng Zhang, and Guolin Ke. Uni-Mol: A universal 3D molecular representation learning framework. In *International Conference on Learning Representations*, 2023.

Shang Zhu, Bharath Ramsundar, Emil Annevelink, Hongyi Lin, Adarsh Dave, Pin-Wen Guan, Kevin Gering, and Venkatasubramanian Viswanathan. Differentiable modeling and optimization of non-aqueous Li-based battery electrolyte solutions using geometric deep learning. *Nature Communications*, 15:8649, 2024.

Murtaza Zohair, Vidushi Sharma, Eduardo A. Soares, Khanh Nguyen, Maxwell Giammona, Linda Sundberg, Andy Tek, Emilio Vital Brazil, and Young-Hye La. Chemical foundation model-guided design of high ionic conductivity electrolyte formulations. *npj Computational Materials*, 11:283, 2025. doi: 10.1038/s41524-025-01774-4.

A Dataset Construction and Evaluation Protocol

Data provenance and evaluation are specified below. Architecture and optimization are detailed in Appendix B; transfer diagnostics and external evaluation are described in Appendices C and E.

A.1 Data Sources and Label Provenance

Electrolyte12 combines experimental measurements and computed labels for 12 electrolyte properties from the 13 sources in Table 5. CheMixHub supplies ILThermo ionic-liquid records, published and NIST-derived viscosity measurements, and computed miscible-solvent properties (Dong et al., 2007; Rajaonson et al., 2025). Other sources include CALiSol-23 (de Blasio et al., 2024), Bamboo-Mixer (Yang et al., 2026), DiffMix and its Clio measurements (Zhu et al., 2024), Electrolytomics (Kumar et al., 2025), IBM SMI-TED-IC (Zohair et al., 2025), and the molecular-dynamics-based OEDB (Nakamura et al., 2026). DiffMix AEM provides model-generated conductivity labels, whereas Clio provides experimental measurements. Each record has a source identifier and label-level provenance, allowing evaluation to distinguish measured properties from computational targets.

Table 5: Sources contributing to Electrolyte12. Counts represent the number of training records retained after molecular compatibility checks; a single record may contain multiple property labels.

Data Source	Label Type	Training Records
Bamboo-Mixer	Experimental conductivity	8,320
CALiSol	Experimental conductivity	9,760
CheMixHub (ILThermo)	Experimental transport	85,379
CheMixHub (Viscosity subset)	Experimental viscosity	26,313
CheMixHub (Miscible solvents)	Computed thermophysical	24,008
CheMixHub (NIST archive)	Experimental viscosity	166,718
Clio	Experimental conductivity	97
DiffMix (AEM subset)	Model-generated conductivity	8,818
DiffMix (Thermodynamics)	Experimental excess properties	1,312
Electrolytomics (Experimental)	Experimental conductivity	7,330
Electrolytomics (Computed)	Computed oxidation stability	14,198
IBM SMI-TED-IC	Experimental conductivity	10,082
OEDB	Computed transport/structure	4,490
Total		366,825

A.2 Curation and Data Partitions

Records are retained when molecular species can be resolved, their elements are supported by the encoder, and at least one of the 12 targets has a finite label. ILThermo observations are reconciled with their source measurements and units. The fixed training, validation, and test partitions are disjoint in record identity and model-input grouping. Conductivity records from the encoder's experimental fine-tuning source are excluded from validation and testing. Additional cross-source test overlaps are removed by matching normalized salt–solvent composition and temperature.

The training and validation sets contain 366,825 and 44,811 records, respectively, with both experimental and computed supervision. The ID benchmark contains 43,709 records: 38,358 with experimental measurements and 5,351 with computed labels. Its 49,520 observed labels comprise experimental conductivity, viscosity, excess molar enthalpy, and excess molar volume, together with computed labels for the remaining eight properties. Table 6 gives the property-wise counts. A record can contain several labels, and missing targets are excluded from both the training loss and evaluation for that property.

A.3 Property Definitions and Target Transformations

For each property t in Table 7, a fixed transform g_t is standardized using the mean μ_t and population standard deviation σ_t of its transformed training labels:

$$v_{it} = \frac{g_t(y_{it}) - \mu_t}{\sigma_t}, \qquad \widehat{y}_{it} = g_t^{-1}(\sigma_t \widehat{v}_{it} + \mu_t). \tag{11}$$

Table 6: Observed property labels in Electrolyte12. Training and validation include experimental and computed labels; ID test blocks identify the label origin used for each property.

Property	Training	Validation	ID test
Experimental test labels			
Cond.	78,818	8,831	7,153
Visc.	252,960	31,605	31,041
H^E	24,515	3,063	62
V^E	805	102	102
Computed test labels			
Dens.	28,498	3,565	3,565
D_+	4,484	563	561
D_-	4,484	563	561
D_s	4,490	564	562
CN_{+-}	4,490	564	562
CN_{+s}	4,490	564	562
Oxid. stab.	14,198	1,773	1,786
H_{vap}	24,008	3,001	3,003
Total	446,240	54,758	49,520

The standard deviation is bounded below by 10^{-6}. All learning objectives minimize $\ell(\widehat{v}, v) = (\widehat{v}-v)^2$ in this space; its mean is denoted z-MSE. Table 7 lists the units and transforms. The transforms accommodate heterogeneous target scales while the inverse mappings return predictions in physical units.

Table 7: Property units and transforms, with $y_+ = \max(y, 0)$ and $y_\epsilon = \max(y, 10^{-12})$. Subscripts $+$, $-$, and s denote cations, anions, and solvents, respectively.

Property	Unit	$g_t(y)$
Conductivity	mS/cm	$\ln(1 + y_+)$
Viscosity	mPa·s	$\log_{10}(y_\epsilon)$
Excess molar enthalpy (H^E)	J/mol	$\mathrm{asinh}(y/1000)$
Excess molar volume (V^E)	cm^3/mol	$\mathrm{asinh}(y)$
Density	$\mathrm{g/cm}^3$	y
Diffusivity (D_+, D_-, D_s)	m^2/s	$\log_{10}(y_\epsilon)$
Coordination (CN_{+-}, CN_{+s})	Dimensionless	$\ln(1 + y_+)$
Oxidation stability	eV	y
Heat of vaporization (H_{vap})	J/mol	$\ln(1 + y_+/1000)$

A.4 Evaluation Metrics

MAE, PCC, and R^2 are computed from predictions and targets in their original physical units. For a set of properties G, macro NMAE is

$$\mathrm{NMAE}_G = \frac{1}{|G|} \sum_{t \in G} \frac{\mathrm{MAE}_t}{\sigma_t^{\mathrm{train,raw}}},$$

where $\sigma_t^{\mathrm{train,raw}}$ is the population standard deviation of the mixed-training labels in physical units. Macro PCC and R^2 also give equal weight to each property. Main prediction tables summarize all 12 targets; grouped ablations report separate four- and eight-property means. Ranks, best counts, and relative improvements use unrounded values. All methods use identical property-wise test samples.

B Implementation Details

B.1 Encoder and Input Representation

We adopt the molecular and formulation encoders of Bamboo-Mixer (Yang et al., 2026), denoting the encoder by E_{BM} with parameters Θ. For formulation x_i, $\mathcal{G}_i^{\mathrm{neu}}$ and $\mathcal{G}_i^{\mathrm{ion}}$ are its neutral and ionic

molecular graphs, $\mathbf{w}_i^{\mathrm{neu}}$ and $\mathbf{w}_i^{\mathrm{ion}}$ specify their compositions, and $\mathbf{c}_i$ contains condition features. A molecular graph module B and neutral/ionic formulation aggregators produce

$$\begin{aligned} \mathbf{h}_i^{\mathrm{neu}} &= A_{\mathrm{neu}}(\{B(G) : G \in \mathcal{G}_i^{\mathrm{neu}}\}, \mathbf{w}_i^{\mathrm{neu}}), \\ \mathbf{h}_i^{\mathrm{ion}} &= A_{\mathrm{ion}}\big(\{B(G) : G \in \mathcal{G}_i^{\mathrm{ion}}\}, \mathbf{w}_i^{\mathrm{ion}}\big), \\ E_{\mathrm{BM}}(x_i;\Theta) &= [\mathbf{h}_i^{\mathrm{neu}}; \mathbf{h}_i^{\mathrm{ion}}; \mathbf{c}_i]. \end{aligned} \tag{12}$$

The private and shared branches use separate encoder copies, yielding $\mathbf{u}_i^{\mathrm{pri}}$ and $\mathbf{u}_i^{\mathrm{shr}}$. The 384-dimensional formulation representation is concatenated with 27 condition features to give $d_u = 411$. Bamboo-Mixer and ElectrolyteFM use the same encoder initializations. The private encoder is fixed, whereas ShKL updates the formulation aggregators with the molecular graph module frozen.

B.2 Prediction Modules and Initialization

The hidden and bottleneck dimensions are $d_h = 256$ and $d_b = 64$, and the shared pool contains $K = 4$ experts. Each private or shared expert consists of a $411 \to 256$ linear layer, SiLU, LayerNorm, dropout with probability 0.1, a $256 \to 256$ linear layer, and SiLU. Each prediction head is a $256 \to 64 \to 1$ MLP with a SiLU activation between the two linear layers.

The property-conditioned router adds a bias-free $411 \to 4$ linear projection to a learned four-dimensional property preference before softmax. Both terms are initialized to zero, so the router initially assigns uniform expert weights. Each residual adapter applies LayerNorm without affine parameters, a $256 \to 64$ down-projection, SiLU, and a $64 \to 256$ up-projection. The up-projection weights and bias are initialized to zero, and the fusion head is initialized from its private head. S2SKF therefore starts from the learned private predictor while retaining separately learned private and shared representations.

B.3 Optimization and Model Selection

SpKL trains each expert and head using only labels for its target property. ShKL samples properties with probability proportional to $N_t^{0.7}$, where N_t is the number of observed training labels, and samples a batch from the selected property. This reduces the dominance of data-rich targets. S2SKF trains each property's adapter and final head with both learned branches and the router frozen. Table 8 summarizes trainable modules and learning rates.

Table 8: Optimization settings for knowledge learning and fusion.

Component	Trainable modules	Initial LR
SpKL	Private expert and property head	5×10^{-4}
ShKL	Neutral and ionic formulation aggregators	10^{-4}
ShKL	Shared experts, router, and prediction heads	5×10^{-4}
S2SKF	Residual adapter and final head	5×10^{-4}

B.4 Baseline Implementations

RDKit-XGB, MolT5-XGB, Bamboo-Mixer, GeoMix, MolSets, and SCAN use the same partitions, target transforms, and missing-label protocol. Joint multi-property baselines select the epoch or boosting round minimizing 12-property macro validation NMAE. Hyperparameters and checkpoints are selected using validation data.

For the learning-strategy comparison, we train 12 independent STL models, one MMoE model, and one two-layer PLE model. Their molecular graph encoders are initialized identically from pretrained weights and remain frozen; formulation aggregators and downstream predictors are trainable. Each STL model has its own aggregators, expert, and property head. MMoE uses four shared experts with a separate input-dependent gate and head per property. Each PLE layer contains four shared experts and one private expert per property, together with property-specific gates.

These models use the optimizer settings, learning rates, z-MSE objective, and $N_t^{0.7}$ multi-task sampling schedule described above.

C Directed Cross-Property Transfer Analysis

C.1 Diagnostic Protocol

The 12 single-property and 66 two-property models share a Bamboo-Mixer backbone, including molecular and formulation encoders and a shared MLP, followed by property-specific heads. Each run uses the union of training records with labels for its active properties; other labels are masked out. Within a batch, MSE is first averaged over the available labels of each active property and then equally over the properties present in that batch.

All runs use AdamW, batch size 128, at most 20 epochs, early-stopping patience 5, and weight decay 10^{-5}. Learning rates are 10^{-4} for the backbone and 5×10^{-4} for the shared MLP and heads. Checkpoints minimize macro validation z-MSE over active properties.

D Additional Multi-Property Prediction Results

Tables 9 and 10 complement the MAE results in Table 1 with PCC and R^2 from the same test predictions. Property order, label-source groups, and twelve-property summaries follow Table 1; sample counts are listed in Table 6.

ElectrolyteFM ranks first on nine properties for each metric: all four experimental-label targets and five computed-label targets. Its macro PCC and R^2 are 0.9268 and 0.8357, respectively.

Table 9: Per-property PCC on Electrolyte12 (higher is better). Blocks denote experimental and computed test labels. Bold and underline indicate the best and second-best unrounded values. The macro and best-count rows summarize all twelve properties.

Property	RDKit-XGB (Landrum et al. 2024)	MolT5-XGB (Edwards et al. 2022)	Bamboo-Mixer (Yang et al. 2026)	GeoMix (Li et al. 2025)	MolSets (Zhang et al. 2024)	SCAN (Wang and You 2026)	**ElectrolyteFM (Ours)**
Experimental labels							
Cond.	0.729	0.748	<u>0.909</u>	0.771	0.899	0.856	**0.942**
Visc.	0.113	0.241	0.823	0.759	<u>0.875</u>	0.258	**0.978**
H^E	0.419	0.528	<u>0.854</u>	0.510	0.821	0.596	**0.983**
V^E	0.927	0.924	<u>0.955</u>	0.854	0.919	0.885	**0.959**
Computed labels							
Dens.	0.920	0.938	<u>0.997</u>	0.978	0.986	0.949	**0.999**
D_+	0.909	0.904	0.954	0.958	<u>0.959</u>	**0.960**	0.958
D_-	0.891	0.893	0.957	0.956	0.962	**0.964**	<u>0.963</u>
D_s	0.889	0.890	0.966	0.956	<u>0.966</u>	0.956	**0.967**
CN_{+-}	0.675	0.677	0.707	<u>0.713</u>	0.696	0.699	**0.714**
CN_{+s}	0.664	0.653	0.678	<u>0.689</u>	0.659	0.668	**0.708**
Oxid. stab.	0.838	0.805	**0.958**	0.833	0.883	0.805	<u>0.953</u>
H_{vap}	0.929	0.896	<u>0.995</u>	0.974	0.938	0.954	**0.997**
Macro PCC	0.7418	0.7582	<u>0.8960</u>	0.8294	0.8802	0.7958	**0.9268**
Best count	0/12	0/12	1/12	0/12	0/12	<u>2/12</u>	**9/12**

Table 10: Per-property R^2 on Electrolyte12 (higher is better). Blocks denote experimental and computed test labels. Bold and underline indicate the best and second-best unrounded values. The macro and best-count rows summarize all twelve properties.

Property	RDKit-XGB (Landrum et al. 2024)	MolT5-XGB (Edwards et al. 2022)	Bamboo-Mixer (Yang et al. 2026)	GeoMix (Li et al. 2025)	MolSets (Zhang et al. 2024)	SCAN (Wang and You 2026)	**ElectrolyteFM (Ours)**
Experimental labels							
Cond.	0.274	0.303	<u>0.822</u>	0.574	0.795	0.723	**0.887**
Visc.	0.000	0.000	<u>0.259</u>	0.025	0.164	0.005	**0.608**
H^E	-0.129	0.006	<u>0.679</u>	-0.206	0.662	0.310	**0.967**
V^E	0.777	0.806	<u>0.910</u>	0.709	0.824	0.780	**0.915**
Computed labels							
Dens.	0.747	0.808	<u>0.995</u>	0.939	0.972	0.901	**0.998**
D_+	0.401	0.490	0.909	0.877	**0.917**	<u>0.913</u>	0.904
D_-	0.386	0.512	0.915	0.905	<u>0.922</u>	**0.927**	0.920
D_s	0.487	0.587	0.906	0.910	<u>0.929</u>	0.909	**0.934**
CN_{+-}	0.326	0.351	0.485	<u>0.493</u>	0.467	0.479	**0.501**
CN_{+s}	0.282	0.293	0.441	<u>0.460</u>	0.424	0.424	**0.492**
Oxid. stab.	0.646	0.597	**0.913**	0.469	0.776	0.647	<u>0.908</u>
H_{vap}	0.681	0.621	<u>0.990</u>	0.905	0.876	0.910	**0.995**
Macro R^2	0.4064	0.4478	<u>0.7687</u>	0.5882	0.7273	0.6607	**0.8357**
Best count	0/12	0/12	<u>1/12</u>	0/12	<u>1/12</u>	<u>1/12</u>	**9/12**

E Out-of-Distribution Generalization and Few-Shot Adaptation

E.1 External Conductivity Dataset

The external dataset is drawn from the supplementary composition and measurement data of Muñoz-Perales et al. (2026). It contains 106 sodium-electrolyte formulations spanning five salts and 15 solvents, with 2–14 components per formulation, salt concentrations of 0.05–3.95 mol/L, and a recorded temperature of 298 K. Conductivity is measured in mS/cm.

The external dataset is reserved for zero-shot evaluation and few-shot adaptation. Table 2 reports zero-shot results on all 106 formulations.

E.2 Support–Query Protocol and Adaptation Methods

We construct 20 episodes, stratified by total component count into groups of 2–6, 7–9, and 10–14. Each episode contains a 32-formulation support pool and 74 query formulations. Nested support sets contain the first K formulations in a fixed random ordering of the pool, with $K \in \{0, 1, 2, 4, 8, 16\}$ for both adaptation protocols. The query set remains fixed across support sizes, and all methods use identical samples.

Affine calibration fits $\widehat{v}_{\mathrm{cal}} = a\widehat{v} + b$ using ridge regularization centered at $(a, b) = (1, 0)$. Final-layer adaptation freezes the representation and the first part of the prediction head, then fits the final $64 \to 1$ layer using ridge regularization centered at its pre-adaptation weights. Regularization strength is selected separately for each method, protocol, and support size using ten ID-validation pseudo-episodes. Query labels are used only for evaluation.

Figure 3 displays affine calibration for RDKit-XGB, MolT5-XGB, Bamboo-Mixer, GeoMix, SCAN, and ElectrolyteFM, and final-layer adaptation for the four neural methods.

F Ablation Studies and Expert Analysis

F.1 Comparison of Learning Strategies

Table 11 provides the label-source breakdown of the twelve-property results in Table 3. STL, MMoE, PLE, and ElectrolyteFM are compared using the training and selection procedures in Appendix B. ElectrolyteFM has the lowest macro NMAE and highest macro PCC and R^2 in each group.

Table 11: Complete learning-strategy comparison. Metrics are equal-weight means over the four experimental-label or eight computed-label properties. Bold and underline indicate the best and second-best values within each column.

	Experimental labels (4)			Computed labels (8)		
Method	NMAE ↓	PCC ↑	R^2 ↑	NMAE ↓	PCC ↑	R^2 ↑
STL	0.0865	0.8705	0.7664	<u>0.2047</u>	<u>0.9044</u>	<u>0.8274</u>
MMoE	0.0728	0.9120	0.8214	0.2098	0.9027	0.8223
PLE	<u>0.0647</u>	<u>0.9647</u>	<u>0.8352</u>	0.2093	0.9019	0.8166
ElectrolyteFM (Ours)	**0.0640**	**0.9655**	**0.8442**	**0.2036**	**0.9075**	**0.8314**

F.2 Contributions of Knowledge Learning and Fusion

Table 12 reports twelve-property macro NMAE, PCC, and R^2 for the configurations in Table 4. SpKL and ShKL are evaluated through their own prediction heads; S2SKF combines both learned representations. The check marks indicate included components.

Table 12: Complete knowledge learning and fusion ablation across all twelve properties. Check marks indicate included components; metrics give equal weight to each property. Bold and underline indicate the best and second-best unrounded values within each column.

SpKL	ShKL	S2SKF	NMAE ↓	PCC ↑	R^2 ↑
✓	–	–	0.1841	0.9171	0.8275
–	✓	–	<u>0.1674</u>	<u>0.9203</u>	<u>0.8318</u>
✓	✓	✓	**0.1571**	**0.9268**	**0.8357**

Across all twelve properties, S2SKF reduces macro NMAE by 14.7% and 6.1% relative to SpKL and ShKL, respectively. It also achieves the highest macro PCC and R^2, improving aggregate performance across all three metrics.

F.3 Expert Allocation and Removal

The routing weights in Figure 4(a) describe which experts each property uses on average. To assess their predictive contributions, we remove each shared expert in turn without retraining. For expert k, its routing weight is set to zero and the remaining weights are renormalized. The resulting shared state is

$$\mathbf{s}_{it}^{(-k)} = \sum_{j \neq k} \frac{\pi_{itj}}{1 - \pi_{itk}} S_j(\mathbf{u}_i^{\mathrm{shr}}).$$

The intervention is evaluated through the same frozen adapter and head, using the training target transforms and standardization statistics.

Table 13: Effect of shared-expert removal on macro z-MSE (lower is better). Each column averages equally over the indicated properties; parentheses give the relative change from the complete model.

Intervention	Experimental (4)	Computed (8)	All (12)
Complete model	0.0472	0.1668	0.1269
Remove S_1	0.0790 (+67.4%)	0.1745 (+4.7%)	0.1427 (+12.4%)
Remove S_2	0.0559 (+18.4%)	0.1739 (+4.3%)	0.1346 (+6.0%)
Remove S_3	0.0534 (+13.1%)	0.1718 (+3.0%)	0.1323 (+4.3%)
Remove S_4	0.0583 (+23.4%)	0.1699 (+1.9%)	0.1327 (+4.6%)

Removing any expert increases twelve-property macro z-MSE, with the largest increase for S_1 (+12.43%). The effect of removing S_1 is much larger for the experimental-label group (+67.35%) than for the computed-label group (+4.65%). Thus average routing preference and the cost of removal provide complementary views of expert use, and contributions vary across the evaluated properties.