Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Chemia: An LLM-Powered Automated Machine Learning Framework for Chemistry and Materials Discovery

Ben Gao
Shanghai Artificial Intelligence Laboratory
Penghui Li
Shanghai Artificial Intelligence Laboratory
Jiaqing Xie
Shanghai Artificial Intelligence Laboratory
Haiyuan Wan
Shanghai Artificial Intelligence Laboratory
Dongzhan Zhou
Shanghai Artificial Intelligence Laboratory
Guoyin Yin
The Institute for Advanced Studies, Wuhan University
Yuqiang Li
liyuqiang@pjlab.org.cn

Shanghai Artificial Intelligence Laboratory

Research Article

Keywords: Automated machine learning, Cheminformatics, Molecular representation, Reaction yield prediction, Out-of-distribution generalization, Large language model agents, Declarative configuration, Reproducibility

Posted Date: November 26th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11345178/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Abstract

Machine learning (ML) is increasingly used in chemistry and materials science, but wider adoption is limited by programming effort, model and hyperparameter selection difficulty, and limited reproducibility. We present Chemia, an open-source, configuration-driven automated ML framework in which a complete modeling workflow—from data loading, splitting and featurization to model competition, Bayesian hyperparameter optimization and result export—is declared in a single YAML file. Chemia combines physicochemical descriptors, topological fingerprints and pretrained chemical foundation model embeddings; accepts single molecules, multicomponent reactions and user-supplied precomputed descriptors; and compares gradient-boosted trees, kernel methods and neural networks under a common protocol. We evaluated Chemia on molecular property regression and classification, reaction yield and selectivity prediction under random and out-of-distribution splits, and electrochemical property prediction for metal–organic framework supercapacitors from externally computed descriptors. In most tasks Chemia reached test performance similar to original studies or benchmark literature without task-specific code, but was less accurate for volumetric capacitance and extrapolation to unseen ligands. We also describe Chemia-Agent, in which an LLM agent operates Chemia only by editing YAML configurations; in a recorded case study, the agent repaired configuration errors and aligned its evaluation protocol with the original study. Chemia is freely available under the MIT license at https://github.com/flyben97/Chemia.

Scientific contribution

Chemia brings single-molecule property prediction, multicomponent reaction modeling and precomputed-descriptor materials modeling into one declarative, schema-validated YAML workflow that combines descriptor, fingerprint and pretrained-embedding representations with automated model competition, whereas existing chemistry AutoML tools such as Uni-QSAR, QSARtuna, ChemTorch and ProQSAR each cover only part of this scope. Because every modeling decision is an explicit configuration field, Chemia workflows are reproducible and auditable, and the same YAML layer gives LLM agents a constrained action space in which they can configure, diagnose and repair modeling workflows without generating Python code.

Introduction

Over the past decade, data-driven artificial intelligence has fundamentally transformed the paradigm for exploring chemical space. From high-throughput virtual screening and the discovery of lead compounds for drug development to the design of next-generation energy materials, machine learning (ML) has become an indispensable engine for accelerating scientific discovery. In recent years, with the rise of chemistry-oriented pretrained foundation models such as ChemBERTa,[1, 2] MolT5,[3] and Uni-Mol,[4–6] as well as graph neural networks (GNNs), AI's ability to extract complex molecular representations and learn structure-activity relationships has grown rapidly.[7–9]

However, this surge in computational capability has not been equally “democratized” among the broader community of experimental chemists. State-of-the-art ML workflows typically require a strong foundation in Python programming, proficiency with tensor computation frameworks such as PyTorch, and substantial experience in feature engineering and hyperparameter tuning. Although AutoML tools such as AutoGluon[10] perform well on generic tabular data, they do not natively handle chemistry-specific data structures, such as SMILES string representations and multicomponent reaction systems. As a result, experimental scientists with deep chemical intuition and domain expertise often cannot apply state-of-the-art ML tools directly.

Recent advances in LLM agents offer an opportunity to narrow this gap. Here we introduce Chemia, a standardized, configuration-driven AutoML framework for chemical and materials modeling. Several automated modeling tools for chemistry already exist and are compared with Chemia below (Table 1). Chemia differs from them in handling single-molecule, multicomponent-reaction and precomputed-descriptor tasks within one declarative workflow: machine-readable YAML configuration files specify the entire workflow, from multimodal representation construction and fusion to automated model search. Chemia is also distributed with an agent-skill description file (SKILL.md) so that LLM agents can discover and operate it.

More importantly, as generative AI evolves toward autonomous agents,[9, 11] asking large language models (LLMs) to write complex ML scripts directly is prone to hallucinated APIs and runtime failures. In Chemia, the YAML file acts as a constrained interface between humans and LLMs: it can be generated and parsed by LLMs, checked against the configuration schema, and then executed deterministically by the underlying computational engine. In this work, we benchmark Chemia on molecular property, reaction and materials modeling tasks and compare the results with values reported in the original studies. We also present a case study in which Chemia-Agent configures, executes, diagnoses and revises a materials modeling workflow through YAML edits, and we discuss the limitations of this evidence.

Results and discussion

Chemia Architecture and Multimodal Molecular Representation Engine

Chemia’s overall architecture balances high automation with the integration of domain knowledge in chemistry (Fig. 1). Users only need to provide a concise dataset and a YAML configuration file, after which the system automatically drives a standardized machine learning workflow encompassing data parsing, molecular representation, feature fusion, model optimization, and result export. Its central component is a unified multimodal molecular representation engine that can integrate molecular information from different levels in parallel: (i) physicochemical descriptors (1D). More than 200 molecular descriptors are automatically computed based on RDKit, covering macroscopic and topological properties such as molecular weight, LogP, TPSA, and the numbers of hydrogen-bond donors

and acceptors, thereby characterizing the overall physicochemical features of molecules. (ii) Topological molecular fingerprints (2D). Chemia supports mainstream topological fingerprints such as Morgan, MACCS, and AtomPair to capture local pharmacophores, bond connectivity, and substructure environments, thereby enhancing the model's ability to perceive structural similarity. (iii) Pretrained foundation-model embeddings (sequence-based and 3D pretrained embeddings). Chemia supports Transformer-based chemical foundation models ranging from 84M to 1.1B parameters. Through automatic conformer generation, graph-structure encoding, and attention-pooling mechanisms, the system can extract high-dimensional representations from models based on three-dimensional spatial-structure learning, such as Uni-Mol V2, as well as semantic embedding vectors from sequence masked-modeling models such as MolT5 and ChemBERTa.

In addition, given that many specialized modeling tasks in chemistry and materials science heavily rely on electronic-structure descriptors obtained from density functional theory (DFT) calculations, Chemia further provides a standardized external-feature interface that allows users to import externally computed descriptor matrices. To maintain consistency with the molecular representations automatically generated by the system, external descriptors are likewise declared and loaded through YAML configuration files, thereby ensuring consistency and reproducibility across the entire workflow. Moreover, because Chemia provides standardized interfaces, it can be combined with other workflows that output structured tabular features, linking feature computation with model training.

During the feature-fusion stage, the system automatically records the concatenation order of different feature types, normalization parameters, and corresponding preprocessors, ensuring that trained models can be stably reproduced and deployed. The fused feature matrix is then passed to an adaptive algorithm library, which includes gradient-boosted tree models such as XGBoost, LightGBM, and CatBoost, as well as deep tabular learning architectures such as TabNet. Hyperparameter search is automatically performed by a Bayesian optimizer based on the tree-structured Parzen estimator (TPE), enabling efficient and robust model optimization. Finally, the best trained model, hyperparameter configuration, dataset split results, and all preprocessing states are fully saved to the output directory to support subsequent reproduction experiments and inference deployment. Both multimodal molecular representation and multi-model comparison can be implemented through efficient and structured YAML files. Example YAML files are provided in the code repository, and a descriptor-only example is given in Supplementary Information (Section 1).

To clarify the position of Chemia within the rapidly developing ecosystem of automated chemical modeling tools, we compared representative frameworks according to workflow-level capabilities rather than reported predictive scores (Table 1). Existing tools automate complementary parts of this space: Uni-QSAR[12] focuses on automated molecular property prediction with pretrained 1D, 2D, and 3D representations; QSARtuna[13] automates the comparison of molecular representations and learning algorithms while incorporating uncertainty and explainability; ChemTorch[14] provides standardized, configuration-driven pipelines for reaction-property modeling and in-/out-of-distribution benchmarking; and ProQSAR[15] emphasizes leakage-aware small-data QSAR modeling with conformal uncertainty and

applicability-domain diagnostics. Chemia is designed to target the intersection of these directions by combining single-molecule prediction, multicomponent reaction modeling, externally supplied descriptors, multimodal feature construction, model competition, and a schema-validated YAML control layer that can be generated, modified, and repaired by an LLM agent.

Table 1
Capability matrix of representative AutoML frameworks for predictive chemical modeling

Framework	Mol.	Rxn.	Ext. feat.	Repr. breadth	Auto search	Config	LLM control/repair
Uni-QSAR	✓	—	—	✓	✓	—	—
QSARtuna	✓	△	△	△	✓	✓	—
ChemTorch	—	✓	—	✓	✓	✓	—
ProQSAR	✓	—	—	△	✓	✓	—
Chemia	✓	✓	✓	✓	✓	✓	✓*

Notes: a, ✓: native and core support; △: partial, limited, or application-level support; —: not reported as a core capability in the corresponding publication or documentation. b, Mol.: single-molecule supervised prediction; Rxn.: reaction or multicomponent-system prediction; Ext. feat.: direct ingestion of user-supplied precomputed descriptor matrices; Repr. breadth: support for multiple representation families, including descriptors, fingerprints, sequence/graph representations, and 3D or pretrained embeddings; Auto search: automated model and/or hyperparameter search; Config: structured configuration-driven execution; *LLM-based configuration repair is demonstrated in Chemia through a recorded autonomous trajectory.

Multimodal Molecular Representation and Molecular Property Prediction

To quantitatively evaluate the practical value of this multimodal representation engine, we used FreeSolv, a hydration free energy dataset,[16] and IA,[17] a chiral chromatographic retention-time dataset, as examples to systematically compare the predictive performance of different molecular representations when used individually and in combination (Table 2). This comparison examines whether combining representations improves on single representations. The results reveal two patterns that are instructive for modeling practice.

FreeSolv contains 642 molecular records, with $\exp t$ as the prediction target; IA contains 1,849 chromatographic records, with RT as the prediction target. The project's data preprocessing pipeline includes SMILES missing-value checking, RDKit-based canonical SMILES normalization, and the generation of molecular fingerprints or pretrained embeddings, with different representations combined through feature concatenation. Regression targets retain their original numerical scale; feature standardization is an optional step, and when enabled, the StandardScaler is fitted only on the training set, with the same transformation applied to the validation and test sets. The splits described here were reconstructed post hoc to match the archived prediction files (Supplementary Information, Section 3).

FreeSolv was first grouped by Bemis–Murcko scaffold and randomly split at an 80:20 scaffold-group ratio, yielding 449 training candidate records and 193 test records; 15% of the training candidates were then randomly selected as an internal validation set, ultimately resulting in 381/68/193 training, validation, and test records. Therefore, 80:20 refers to the scaffold-group ratio, not the molecule-count ratio. IA was randomly split by record at approximately 90:5:5, yielding 1,663/93/93 training, validation, and test records. All the above reconstructions used random seed 42. The IA split was not grouped by molecular identity, so the same molecule may appear in different sets; its evaluation therefore reflects predictive capability under random record hold-out conditions.

First, there is no universally optimal single representation. On FreeSolv, the sequence-based pretrained embedding (MolT5) performed the worst when used alone (R^2 = 0.21), and the three-dimensional structural representation Uni-Mol also showed unsatisfactory performance (R^2 = 0.33). In contrast, the classical MACCS topological fingerprint achieved the best performance (R^2 = 0.74). On the IA dataset for chiral chromatographic column separation, however, the situation was almost reversed: the Uni-Mol three-dimensional representation (R^2 = 0.74) substantially outperformed MACCS (R^2 = 0.65), while the Morgan fingerprint (R^2 = 0.76) and the semantic embedding MolT5 (R^2 = 0.75) also performed strongly. These results indicate that, at least for these two datasets, the best-performing representation is task dependent, so a single fixed representation is unlikely to be optimal across tasks.

Table 2
Molecular Representation Ablation: Single Representations and Multimodal Combinations

Molecular Representation	**Modality Level**	**FreeSolv Test R^2**	**IA Test R^2**
MACCS	2D Topological Fingerprint	0.744	0.651
Morgan	2D Topological Fingerprint	0.595	0.757
MolT5	Semantic Embedding	0.206	0.751
Uni-Mol (v1)	3D Structure	0.33	0.743
MACCS + Morgan	2D + 2D	**0.756**	0.672
MolT5 + MACCS	Semantic + 2D	0.637	0.744
Uni-Mol + MACCS	3D + 2D	0.717	**0.797**

Note: Values in the table are R^2 scores on the fixed test set after Bayesian optimization of XGBoost using each representation. Differences between the best combination and the best single representation are not statistically resolved (95% bootstrap intervals include zero; Supplementary Information, Table S2).

Second, combining complementary representations gave the highest point estimates in both tasks, but the gains were modest. On FreeSolv, the MACCS and Morgan combination increased R^2 from 0.744 (MACCS alone) to 0.756; on IA, combining Uni-Mol and MACCS (three-dimensional structure and topology) increased R^2 from the best single-representation value of 0.757 (Morgan) to 0.797. Paired

bootstrap analysis of the test predictions gave 95% intervals for these differences of − 0.039 to 0.068 and − 0.099 to 0.188, respectively (Supplementary Information, Table S2), so neither improvement is statistically resolved on test sets of this size (193 and 93 records). The practical conclusion is therefore not that fusion is always better, but that the most suitable representation is task dependent and worth exploring; in Chemia, this requires changing only a few lines of YAML, and the combination can be selected on validation data.

The previous section demonstrated Chemia's ability to handle regression tasks using its multimodal representation engine. To examine whether Chemia also provides predictive capability in classification scenarios, we evaluated it on two representative molecular property classification tasks from the MoleculeNet benchmark: BACE, β-secretase 1 inhibitory activity with 1,513 molecules, and BBBP, blood–brain barrier permeability with 2,050 records. For all tasks, Chemia used its built-in multimodal representations, including MolT5 semantic embeddings, Morgan fingerprints with radius 2 and 2,048 bits, MACCS topological fingerprints, and RDKit physicochemical descriptors, which were automatically concatenated into a feature vector of approximately 3,456 dimensions. XGBoost was then automatically optimized through tree-structured Parzen estimator Bayesian search with 50 trials. The evaluation metric was AUC ROC, the standard metric for classification tasks.

We strictly adopted the deterministic Bemis Murcko scaffold splitting protocol widely used by the MoleculeNet[18] benchmark community. Under an 80/10/10 ratio, scaffold groups were deterministically sorted by the number of molecules they contained, with chirality preserved during scaffold computation. The largest scaffolds were assigned preferentially to the training set, the next largest to the validation set, and the remaining smallest and rare scaffolds, including many single molecule scaffolds, to the test set. The 10% validation set was used only for Bayesian hyperparameter search and model selection,[19] while the test set was used strictly for a single final evaluation, so that no test-set information was used for hyperparameter search or model selection. We verified offline that, applied to the same input molecules, our implementation reproduces a reference implementation of this protocol molecule by molecule (BACE: 1,210/151/152; BBBP, 2,039 RDKit-parseable molecules: 1,640/205/194), and that the scaffolds in the three sets were mutually disjoint (Supplementary Information, Section 3). For BACE, the 1,513 molecules were split into 1,210 training, 151 validation, and 152 test molecules. For BBBP, 2,021 of the 2,050 records were retained after removing SMILES that could not be parsed or converted by all representation models, and the retained molecules were split into 1,625 training, 203 validation, and 193 test molecules. Because 18 more molecules were removed than in the commonly used set of 2,039 RDKit-parseable molecules, the BBBP split is close to, but not identical with, the reference split of that set, and literature values for BBBP are therefore reference ranges rather than strictly matched baselines.

Table 3
Scaffold Extrapolation Performance of Chemia on MoleculeNet Molecular Property Classification Tasks

Dataset	Chemia Test AUC	MoleculeNet Baseline	SOTA Pretrained Model
BACE	0.811	~ 0.78–0.81	~ 0.85
BBBP	0.716	~ 0.69–0.73	~ 0.72

Chemia achieved a test AUC of 0.811 on BACE and 0.716 on BBBP (Table 3). The BACE result, 0.811, lies at the upper end of the classical baseline range, approximately 0.78 to 0.81, but remains below the approximately 0.85 reported for the best pretrained models. The BBBP result, 0.716, is on par with the classical baselines reported in the original benchmark, approximately 0.69 to 0.73, as well as recent state of the art graph models based on large scale molecular pretraining, such as GROVER[20] and Uni-Mol, which achieve approximately 0.72 on BBBP. This performance was obtained with Chemia's standard representations (MolT5 embeddings, topological fingerprints and physicochemical descriptors) and gradient-boosted trees under a single YAML configuration, without task-specific pretraining, manual feature engineering or modeling code. BBBP is challenging under scaffold splitting because the test scaffolds are structurally distant from the training set. Because each result comes from a single split and a single seed, and scaffold-split AUC values can vary by several points between splits and seeds, we interpret these results as showing that a code-free Chemia configuration reaches the level of established baselines, not that it outperforms specialized models.

It is also worth noting that, under the stringent scaffold extrapolation setting, deep tabular models such as ANN and TabNet performed substantially worse than gradient boosted trees. In preliminary comparisons using the same representations on an earlier scaffold split (random assignment of scaffold groups, 80/20), XGBoost achieved the highest test AUC on both datasets, whereas ANN and TabNet reached only approximately 0.46 to 0.63, with the ANN close to random performance on BBBP; the deep models were therefore not rerun on the final split. This is consistent with a common observation in the field: for high dimensional sparse features and out of distribution samples under scaffold extrapolation, tree based ensemble models are often more robust than deep neural networks. XGBoost is therefore reported as the primary model for this task.

Molecular Representation of Multicomponent Systems: Reaction Yield Prediction

Beyond property prediction with single compounds as inputs, Chemia also supports molecular representation for multicomponent systems. We demonstrate this capability through organic reaction modeling. Chemia's hierarchical multi-SMILES parsing pipeline independently extracts molecular representations for each reaction component, using Morgan fingerprints, MACCS keys, RDKit physicochemical descriptors, or deep representations such as MolT5, ChemBERTa, and Uni-Mol. These component-level representations are concatenated in the user-declared order to form a unified reaction-

state vector, which is then passed to the adaptive algorithm library for full model competition and Bayesian hyperparameter optimization.

To systematically evaluate the scope of Chemia in multicomponent reaction modeling, we benchmarked it on four representative reaction data sets: (i) BHC,[21] a Buchwald-Hartwig C-N cross-coupling data set involving four components, namely ligand, additive, base, and aryl halide, with reaction yield as the target and an approximately 7:3 random train/test split; (ii) CPAC,[22] an asymmetric chiral phosphoric acid catalysis data set involving three components, namely catalyst, imine, and thiol, with the activation free-energy difference $\Delta\Delta G^{\ddagger}$ as the target. CPAC was evaluated under three extrapolative splitting protocols, substrate extrapolation, catalyst extrapolation, and joint substrate-catalyst extrapolation, to probe generalization to unseen chemical space; (iii) SHC-oAK,[23] a sulfonium salt and ortho-acetyl ketone cross-coupling data set involving three components, substrate S1, aryltrifluoroborate S2, and product, with reaction yield as the target and the predefined train/test split from the original study; and (iv) SMC,[24] a Suzuki-Miyaura cross-coupling data set involving six components, electrophile SM1, boronic ester or boronate SM2, product, ligand, base, and solvent, with reaction yield as the target. SMC was evaluated under three protocols: random split, borane extrapolation, and ligand extrapolation.

For preprocessing, component SMILES were canonicalized using RDKit, and Morgan fingerprints (radius 2, 2,048 bits), MACCS fingerprints (167 bits), and RDKit molecular descriptors were calculated separately for each component. These features were concatenated in a fixed component order to construct a reaction-level representation. Entirely missing descriptor columns were removed before filtering records with remaining non-finite feature values. The predefined test partitions were retained, while internal validation subsets drawn from the training pools were used for hyperparameter optimization. Predictive performance was evaluated using R^2, root mean squared error, and mean absolute error, with targets retained on their original numerical scales; consequently, absolute error values should be interpreted within each task rather than compared directly across datasets.

Across all reaction data sets, we adopted a unified component-wise representation strategy based on Morgan fingerprints with radius 2 and 2,048 bits, MACCS keys with 167 bits, and RDKit physicochemical descriptors. The final feature dimensionality increased with the number of reaction components, reaching approximately 7,000 dimensions for CPAC and SHC-oAK, approximately 9,700 dimensions for BHC, and approximately 14,000 dimensions for the six-component SMC system. All models were automatically optimized with 50 to 100 trials of Bayesian search based on the tree-structured Parzen estimator. A fixed random seed was used, either 42 or multiple seeds for robustness checks. The internal validation set was used for hyperparameter optimization, and each tuned candidate model was evaluated once on the test set. Unless stated otherwise, the best model reported for each benchmark is the candidate with the highest test score in Chemia's model-comparison table.

We first evaluated three reaction data sets under random or predefined splits. On BHC (Fig. 2a), Chemia achieved an R^2 of 0.95 using XGBoost, comparable to results obtained by pretrained models such as Yield-BERT and RXNGraphormer. LightGBM (R^2 = 0.928) and CatBoost (R^2 = 0.940) also delivered

similarly strong performance. In contrast, the ANN failed to learn a useful model in this high-dimensional, moderate-sample-size setting (R^2 = − 1.07), and TabNet was also less accurate (R^2 = 0.896). This observation is consistent with the trend observed in molecular property prediction, where tree-based models are often favored for high-dimensional sparse features.

In the SMC random split (Fig. 2b), XGBoost achieved R^2 = 0.837 using RDKit fingerprints and descriptors on an 800-reaction subset (561 training and 239 test reactions). When the representation was switched to MolT5 semantic embeddings, with 1,024 dimensions per component and 6,144 dimensions in total for the six-component system, and the model was trained on the full split of 4,031 training and 1,729 test reactions, ANN achieved R^2 = 0.876, with RMSE = 10.10 and MAE = 6.66. This was slightly higher than XGBoost with the same representation and split (R^2 = 0.862); in contrast to the fingerprint-based BHC results, the neural network outperformed the gradient-boosted trees in this setting, although the margin is small and comes from a single split. Because both the training-set size and the representation differ from those of the 800-reaction experiment, this comparison cannot separate the effect of data size from that of the representation, but it is consistent with the general expectation that neural networks benefit from larger training sets and from continuous embeddings such as MolT5.

On the SHC-oAK data set, using the predefined split of 860 training and 172 test reactions (Fig. 2c), CatBoost achieved a test R^2 = 0.670, RMSE = 0.104, and MAE = 0.083 after 100 trials of Bayesian optimization, with a training R^2 = 0.968. To exclude the possibility that this result was driven by a particular random seed, we repeated the complete training workflow under three independent seeds, 42, 123, and 456. All three runs gave a test R^2 of approximately 0.67, indicating that the result is insensitive to the random seed for this fixed split. Together with the large gap between training and test R^2, this suggests that performance on this dataset is limited by the information captured by the present features rather than by hyperparameter optimization, although testing this hypothesis would require alternative representations. It is also worth noting that the potassium aryltrifluoroborate type SMILES of the S2 component, such as '*F[B-](F)(F)c1ccccc1.[K+]*', triggered column-wise NaN values in charge-sensitive RDKit 2D descriptors such as BCUT2D. These columns were automatically removed by the framework, as described in the preprocessing paragraph above. Such cases are not uncommon in multicomponent reaction systems, where different reaction components impose distinct robustness requirements on molecular featurizers. Chemia's built-in abnormal-value detection and column-level cleaning mechanisms therefore provide necessary fault tolerance.

We next examined the more challenging problem of out-of-distribution (OOD) extrapolation in reaction prediction. The three extrapolative splits of the CPAC data set form a natural difficulty gradient and directly test the ability of the model to generalize to chemical space unseen during training (Fig. 2d). Substrate extrapolation, with 381 training and 219 test reactions, where the imine and thiol combinations in the test set were absent from training, was the least difficult among the three settings. The best model, CatBoost, achieved a test R^2 = 0.908, RMSE = 0.209, and MAE = 0.154. The gap between the

training R^2 of 0.959 and the test R^2 was only approximately 0.05, indicating good transfer to unseen substrate combinations.

Catalyst extrapolation, with 381 training and 307 test reactions, where the chiral phosphoric acid catalyst scaffolds in the test set were unseen, was substantially more challenging. In this setting, the best model shifted to XGBoost, which achieved a test R^2 = 0.767 and RMSE = 0.338, representing a decrease of approximately 0.14 relative to substrate extrapolation. Joint substrate-catalyst extrapolation, with 341 training and 214 test reactions, where both the substrates and catalysts in the test set were unseen, was the most stringent setting. CatBoost was again optimal, yielding a test R^2 = 0.769 and RMSE = 0.331, a difficulty level comparable to catalyst extrapolation alone.

This hierarchy of extrapolation difficulty, substrate extrapolation being easier than catalyst extrapolation, with joint extrapolation similarly difficult to catalyst extrapolation, is consistent with chemical intuition. A plausible explanation is that structural variation in chiral phosphoric acid catalysts, including substituent identity, steric profile, and binaphthyl backbone modification, has a stronger influence on the enantioselective activation barrier than variation in substrate combinations. Once the catalyst structure is represented within the training set, the model can effectively learn the mapping between catalytic environment and substrate response. Extending this mapping to entirely new catalyst scaffolds, however, poses a substantially greater challenge. Notably, XGBoost showed highly variable performance across the three CPAC extrapolation protocols, with R^2 values of 0.469, 0.767, and 0.310 for substrate, catalyst, and joint extrapolation, respectively. Its marked underperformance relative to CatBoost in substrate and joint extrapolation highlights a strong model-split interaction and illustrates the value of comparing several model families. A single model chosen a priori can easily miss the algorithm that generalizes best under a specific extrapolation regime, although identifying that algorithm without test labels requires a validation split that mimics the extrapolation regime.

The SMC system provides a second set of targeted extrapolation tests (Fig. 2e): borane extrapolation and ligand extrapolation. In the borane extrapolation split, with 480 training and 320 test reactions, the boronic ester or boronate nucleophiles SM2 in the test set were absent from training. CatBoost achieved R^2 = 0.729 and RMSE = 15.68, substantially below the random-split result of 0.837 obtained on an 800-reaction subset of the same size. This suggests that cross-coupling yields are sensitive to the structure of the nucleophilic boron component. The electronic properties and steric profiles of boronic ester substituents, together with the hydrolytic stability of boronates, directly affect the efficiency of the transmetalation step, and these effects may be difficult to extrapolate from the boranes observed during training.

Ligand extrapolation was even more challenging, with 640 training and 160 test reactions, where the phosphine ligands in the test set were unseen during training. CatBoost achieved only R^2 = 0.493 and RMSE = 19.93, making this the most difficult extrapolation scenario among all reaction benchmarks. This result is chemically plausible. The electronic effects of phosphine ligands, including σ-donor ability and π-acidity, together with steric effects such as cone angle and bite angle, directly regulate the kinetics of

oxidative addition and reductive elimination, which are key rate-determining steps in the Suzuki coupling catalytic cycle. Even subtle changes in ligand structure can lead to large variations in reaction yield. Yield prediction for reactions involving unseen ligands is therefore expected to be among the most difficult settings in reaction prediction, and the present representation does not solve it.

External Computed Descriptor Modeling and Agent-Driven Autonomous Optimization

The tasks described above all rely on Chemia's built-in multimodal representation engine to automatically generate features from SMILES. However, many frontier modeling problems in chemistry and materials science, such as porous materials, electronic structures, and catalytic systems, depend on key descriptors derived from molecular dynamics simulations, density functional theory (DFT) calculations, or high-throughput characterization, which cannot be directly inferred from molecular graphs. To validate the effectiveness of Chemia's standardized external-descriptor interface and to demonstrate the autonomous closed-loop capability of Chemia-Agent, we selected a representative materials data set: the MOF-IL supercapacitor database reported by Wang et al.[25] This data set contains 510 metal-organic framework (MOF) and ionic-liquid electrolyte supercapacitor systems. Each system is characterized by six nanopore structural descriptors, including pore dimensionality, pore limiting diameter (PLD), largest cavity diameter (LCD), specific surface area (SSA), porosity, and density. The target property is gravimetric capacitance calculated from constant-potential molecular dynamics simulations. Because all descriptors are externally computed, this data set provides a suitable test case for Chemia's '*precomputed_features*' interface.

Chemia supports modeling with external descriptors without writing code. Users only need to specify the six descriptor columns in '*precomputed_features.feature_columns*' and assign the target property to '*target_col*' in the YAML configuration. No SMILES strings are required. Chemia then incorporates the external feature matrix into its unified training and optimization pipeline. For comparison with the original study, we reproduced its evaluation protocol as implemented in the released script ('train_xgboost.py'), namely a stratified 80/20 hold-out split based on target-value deciles with '*random_state = 42*', and reported performance on the fixed 20% test set. Under this protocol, 11 models from Chemia's algorithm library were compared, each with 300 trials of Bayesian hyperparameter search. When the final model was chosen by its validation score, the selected model was an ANN with a test R^2 of 0.893. Among all candidates, the highest test R^2 was obtained by kernel ridge regression (R^2 = 0.911), followed by support vector regression (R^2 = 0.904), ExtraTrees (R^2 = 0.898), and CatBoost (R^2 = 0.897); because this ranking uses the test set, we report it only as an upper reference (Table 4; Supplementary Information, Table S4). Both values are slightly below the machine-learning baseline reported in the original work for gravimetric capacitance (test R^2 = 0.92), showing that the external-descriptor interface can reproduce an expert-written workflow with similar, but not higher, accuracy from a declarative configuration file.

Gradient-boosted tree models such as XGBoost and GBDT reached training R^2 values above 0.99, indicating overfitting, whereas kernel ridge regression had a training R^2 = 0.929 and the smallest train-test gap. This illustrates why it is useful to compare several model families under one protocol rather than to choose a single algorithm a priori. At the same time, the validation-selected model was not the one with the highest test score for four of the five targets (Supplementary Information, Table S4), and with 102 test samples the differences among the leading models (test R^2 of 0.893–0.911) are small relative to the sampling uncertainty of a single hold-out split.

Predictions for two additional electrochemical targets further examined the generality of this interface (Table 4). Current density was modeled as log(I), strictly following the definition in the original study, because the raw current values span three orders of magnitude. The validation-selected model (ANN) achieved a test R^2 of 0.879, and the best candidate on the test set (random forest) achieved R^2 = 0.914 and MAE = 0.14; both exceed the original baseline of R^2 = 0.86 and MAE = 0.17. For volumetric capacitance (Cv), the comparison requires careful definition of scope. The original study reported only a mixed overall baseline that combined one-dimensional and three-dimensional pore systems (R^2 = 0.80, MAE = 7.74). We first modeled the same mixed data set under the identical evaluation scope. The validation-selected model (SVR) achieved a test R^2 of 0.609, and the best candidate on the test set (kernel ridge regression) achieved 0.630; both are clearly lower than the reported value of 0.80. This is the only directly comparable result for Cv.

In addition, we noted that comments in the original ‘train_xgboost.py’ script explicitly suggested that volumetric capacitance should be trained and tested separately for one-dimensional and three-dimensional pore systems. Following this recommendation, we performed an additional analysis in which one-dimensional and three-dimensional pore systems were modeled independently, while two-dimensional pore samples were excluded because only 12 examples were available. For the three-dimensional pore systems, the ANN model, which was also the validation-selected model, achieved R^2 = 0.831 and MAE = 7.41. For the one-dimensional pore systems, with 65 test samples, the best candidate on the test set (SVR) achieved R^2 = 0.640, whereas the validation-selected model (XGBoost) achieved 0.524. Importantly, the original study did not report separate baselines for different pore dimensionalities. Therefore, these subset-specific results are not directly comparable to the mixed R^2 = 0.80 baseline and should not be interpreted as a direct improvement over the original result. Their value instead lies in revealing a structure-property dependence that is obscured by mixed reporting: under pore-specific modeling, one-dimensional pore structures show substantially weaker explanatory power for volumetric capacitance than three-dimensional systems. This subset analysis illustrates the additional scientific value of a standardized workflow. For fair comparison with the original study, however, the directly matched mixed-data result for Cv (test R^2 of 0.609–0.630) does not reach the reported baseline.

More importantly, using this data set as a testbed, we recorded an optimization trajectory of Chemia-Agent in which Claude Code (Anthropic; Claude Opus 4.8 model) served as the LLM agent. Its interaction

pattern is shown in Fig. 3. Throughout the entire process, the large language model agent drove Chemia only by reading and writing YAML configurations, without writing any Python code. The trajectory began from a deliberately naive configuration of the kind a novice user might write, consisting of a single XGBoost model, 10 hyperparameter-search trials, and one hold-out validation split. During execution, the agent first encountered two real errors: a missing runtime path and misuse of the data-interface mode. By parsing the structured error messages returned by the framework and the behavior of the data loader, it autonomously diagnosed the correct contract: externally computed descriptor data should be loaded through the '*single_file*' mode, together with an empty SMILES column and the '*precomputed_features*' interface. After correcting the configuration, the agent obtained the first usable baseline with a test R^2 = 0.904.

Table 4
Test-Set Performance of Chemia on Modeling Tasks in the MOF-IL Dataset

Target	**Subset**	**Validation-selected model (test R^2)**	**Best on test (post hoc)**	**Test R^2**	**Test MAE**	**Original R^2**	**Original MAE**
Gravimetric Capacitance Cg	Full Set	ANN (0.893)	KernelRidge	0.911	12.33	0.92	11.8
Current Density log(I)	Full Set	ANN (0.879)	RandomForest	0.914	0.14	0.86	0.17
Volumetric Capacitance Cv	Mixed (1D + 3D)	SVR (0.609)	KernelRidge	0.630	10.32	0.80	7.74
Volumetric Capacitance Cv	3D Pore	ANN (0.831)	ANN	0.831	7.41	-	
Volumetric Capacitance Cv	1D Pore	XGBoost (0.524)	SVR	0.640	11.39	-	

Note: "Best on test" is the candidate with the highest test R^2 among the 11 models and is therefore an optimistic (post hoc) estimate; "Validation-selected" is the candidate with the best validation score, which can be identified without test labels (Supplementary Information, Table S4). Test MAE refers to the best-on-test model. Original values are from the original study, which reported Cv only for the mixed (1D + 3D) set.

The agent then diagnosed the performance of this baseline and identified three actionable points for improvement. First, the training R^2 = 0.982 was substantially higher than the test value, suggesting overfitting. Second, only a single model had been evaluated, leaving the adaptive algorithm library underused. Third, a single hold-out split may have high variance for a small data set with N = 510. Based on this diagnosis, the agent autonomously upgraded the configuration to multi-model competition,

including XGBoost, LightGBM, CatBoost, and random forest, combined with a deeper tree-structured Parzen estimator search and cross-validation.

Crucially, before preparing the final report, the agent further inspected the original data source and discovered the baseline code released with the original study. This allowed it to identify a protocol mismatch: its own random cross-validation protocol was inconsistent with the stratified hold-out protocol used in the original report, meaning that the two sets of results were not directly comparable. The original study also explicitly required volumetric capacitance to be modeled separately for different pore dimensionalities. The agent therefore realigned the evaluation protocol exactly with the original study, using a stratified 80/20 hold-out split, a fixed test set, and pore-dimensionality grouping. Under this protocol, the agent selected CatBoost as the final model on the basis of test accuracy and overfitting diagnostics (test R^2 = 0.895, compared with 0.92 in the original study). Because this choice used test-set accuracy, 0.895 is not an unbiased estimate of generalization performance.

The trajectory was not free of human input. The author supplied the baseline values reported in the original study and pointed out that current density had been modeled as log(I) in that work, after which the agent rebuilt the target and the corresponding split. In a final step, a more specific instruction asked the agent to use the full model library and a deeper search without changing the data, split or test set; the resulting configuration (11 models, 300 trials) produced the results in Table 4. The value of this case study therefore lies in showing that an LLM agent can diagnose interface errors, overfitting and protocol mismatches and correct them through auditable YAML edits; it is a single trajectory with one LLM and not a quantitative evaluation of agent reliability.

Limitations

Several limitations should be considered when interpreting these results. First, most benchmarks were evaluated on a single data split with a single random seed (SHC-oAK being the exception), and confidence intervals were computed only for the representation ablation; with small test sets (for example, 93 IA records, 102 MOF-IL samples and 152 BACE molecules), differences of a few hundredths in R^2 or AUC are within the expected sampling variability. Second, where several candidate models were evaluated on the same test set, the reported best model is the one with the highest test score, which is an optimistic estimate; for the MOF-IL tasks we therefore also report the validation-selected models (Table 4), whose test R^2 was up to 0.12 lower. Third, comparisons with literature values are indicative rather than strictly matched, because preprocessing, retained molecule sets, split implementations and hyperparameter budgets differ between studies, and we did not retrain the original baseline models under identical conditions. Fourth, the experiments in this study used fixed-length representations with tabular learners; end-to-end fine-tuning of pretrained models and graph neural networks were not evaluated, and accuracy is bounded by the information content of the chosen features, as the SHC-oAK and ligand-extrapolation results suggest. Fifth, Chemia does not compute quantum-chemical or simulation descriptors itself, and this study did not evaluate uncertainty quantification or applicability-domain assessment, both of which are important for prospective use. Finally, the evidence for Chemia-

Agent consists of a single recorded trajectory on one dataset with one LLM, which included human input; systematic evaluation of success rates, failure modes and robustness across tasks and language models remains future work, and agent-generated configurations should be reviewed by users before their results are relied upon.

Conclusions

Chemia was designed to address a central question: can a full-stack chemical machine learning workflow, encompassing multimodal molecular representation, automated model optimization, and rigorous evaluation, be reduced to a declarative configuration file that is readable, writable, and version-controllable by non-AI scientists? For the molecular property, multicomponent reaction and external-descriptor materials benchmarks studied here, the answer is largely positive: a configuration-driven framework reached performance similar to reported baselines in most tasks without requiring users to write modeling code, while falling short in some settings, notably volumetric capacitance prediction and extrapolation to unseen ligands.

Our results also suggest that YAML can serve as a constrained and auditable interface between human scientists, machine learning engines, and LLM agents. Compared with free-form Python generation by LLMs, a schema-validated YAML interface constrains the action space, exposes every modeling decision as an explicit configuration field, and allows errors to be detected before execution. This design makes agent behavior more controllable and traceable, while preserving enough flexibility for model selection, feature construction, hyperparameter search, and evaluation protocol design. Quantifying how much this design improves agent reliability compared with free-form code generation will require systematic evaluation across tasks and models.

Looking forward, Chemia's declarative architecture can be extended toward uncertainty quantification, multi-objective optimization, active learning, and closed-loop experimental discovery. By combining declarative AutoML, pretrained chemical representations, and an agent-safe configuration protocol, Chemia provides a practical foundation for making advanced machine learning workflows more accessible to chemists and materials scientists.

Abbreviations

ANN
artificial neural network
AUC ROC
area under the receiver operating characteristic curve
AutoML
automated machine learning
BHC
Buchwald–Hartwig C–N cross-coupling

CPAC
chiral phosphoric acid catalysis
DFT
density functional theory
GBDT
gradient-boosted decision trees
GNN
graph neural network
IL
ionic liquid
LLM
large language model
MAE
mean absolute error
ML
machine learning
MOF
metal–organic framework
OOD
out-of-distribution
RMSE
root mean squared error
SMC
Suzuki–Miyaura cross-coupling
SMILES
simplified molecular-input line-entry system
SVR
support vector regression
TPE
tree-structured Parzen estimator
YAML
YAML Ain’t Markup Language

Declarations

Availability and requirements

Chemia is an open-source project available at https://github.com/flyben97/Chemia. It is platform independent (a Docker configuration is provided) and is implemented in Python (3.8–3.12). Its dependencies include RDKit (≥2022.03), scikit-learn, XGBoost, LightGBM, CatBoost, PyTorch, pytorch-tabnet, Optuna (≥3.0) and PyYAML, as well as Hugging Face Transformers and unimol-tools for

pretrained embeddings, with the full list given in requirements.txt and environment.yml in the repository. Chemia is released under the MIT license, with no restrictions on use by non-academics.

Acknowledgements

The numerical calculations in this paper have been done on the supercomputing system in the Supercomputing Center of Wuhan University. Part of this work was conducted during an internship at the Shanghai Artificial Intelligence Laboratory.

Authors' contributions

B.G. designed the software framework and developed and implemented the code. P.L. and J.X. participated in software testing and validation. H.W. and D.Z. provided technical assistance with code development. G.Y. and Y.L. conceived and supervised the overall project and acquired funding. All authors discussed the results and reviewed the manuscript.

Research Funding

This work was supported by the National Natural Science Foundation of China (22122107) and the National Key R&D Program of China (2022ZD0160102).

Data Availability

The Chemia source code, example YAML configuration files and the molecular property datasets used in this study (FreeSolv, BACE, BBBP and the IA retention-time dataset) are available in the GitHub repository https://github.com/flyben97/Chemia. The reaction datasets (BHC, CPAC, SHC-oAK and SMC) and the MOF-IL dataset were obtained from the original publications cited in the Results section; the MOF-IL data are distributed by their authors under a CC BY-NC-SA 4.0 license.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Dual publication

This manuscript has not been published previously and is not under consideration elsewhere.

Third party material

The authors created all figures, tables and graphical elements in the manuscript and Supplementary Information. No third-party illustrations or other graphical material were reproduced or adapted. The sources of previously published datasets and benchmark values are cited in the manuscript and Supplementary Information.

References

1. Singh R, Barsainyan AA, Irfan R et al (2026) ChemBERTa-3: an open source training framework for chemical foundation models. Digit Discovery 5:662–685. https://doi.org/10.1039/D5DD00348B
2. Chithrananda S, Grand G, Ramsundar B (2020) ChemBERTa: Large-Scale Self-Supervised Pretraining for Molecular Property Prediction
3. Edwards C, Lai T, Ros K et al (2022) Translation between Molecules and Natural Language
4. Lu S, Gao Z, He D et al (2024) Data-driven quantum chemical property prediction leveraging 3D conformations with Uni-Mol+. Nat Commun 15:7104. https://doi.org/10.1038/s41467-024-51321-w
5. Zhou G, Gao Z, Ding Q et al (2023) Uni-Mol: a universal 3D molecular representation learning framework
6. Ji X, Wang Z, Gao Z et al (2024) Uni-Mol2. Exploring Molecular Pretraining Model at Scale
7. Butler KT, Davies DW, Cartwright H et al (2018) Machine learning for molecular and materials science. Nature 559:547–555. https://doi.org/10.1038/s41586-018-0337-2
8. Sanchez-Lengeling B, Aspuru-Guzik A (2018) Inverse molecular design using machine learning: Generative models for matter engineering. Science 361:360–365. https://doi.org/10.1126/science.aat2663
9. Vamathevan J, Clark D, Czodrowski P et al (2019) Applications of machine learning in drug discovery and development. Nat Rev Drug Discov 18:463–477. https://doi.org/10.1038/s41573-019-0024-5
10. Erickson N, Mueller J, Shirkov A et al (2020) AutoGluon-Tabular: Robust and Accurate AutoML for Structured Data
11. Bran M, Cox A, Schilter S O, et al (2024) Augmenting large language models with chemistry tools. Nat Mach Intell 6:525–535. https://doi.org/10.1038/s42256-024-00832-8
12. Gao Z, Ji X, Zhao G et al (2023) Uni-QSAR: an Auto-. ML Tool for Molecular Property Prediction
13. Mervin L, Voronov A, Kabeshov M, Engkvist O (2024) QSARtuna: An Automated QSAR Modeling Platform for Molecular Property Prediction in Drug Design. J Chem Inf Model 64:5365–5374. https://doi.org/10.1021/acs.jcim.4c00457
14. De Landsheere J, Zamyatin A, Karwounopoulos J, Heid E (2026) ChemTorch: A Deep Learning Framework for Benchmarking and Developing Chemical Reaction Property Prediction Models. J Chem Inf Model 66:2434–2442. https://doi.org/10.1021/acs.jcim.5c02645
15. Phan T-M, Phan T-L, Van-Nguyen P-C et al (2026) ProQSAR: A modular and reproducible framework for small-data QSAR modeling with fit-and-use models. J Cheminform 18:52.

https://doi.org/10.1186/s13321-026-01175-9

16. Mobley DL, Guthrie JP (2014) FreeSolv: a database of experimental and calculated hydration free energies, with input files. J Comput Aided Mol Des 28:711–720. https://doi.org/10.1007/s10822-014-9747-x
17. Xu H, Lin J, Zhang D, Mo F (2023) Retention time prediction for chromatographic enantioseparation by quantile geometry-enhanced graph neural network. Nat Commun 14:3095. https://doi.org/10.1038/s41467-023-38853-3
18. Wu Z, Ramsundar B, Feinberg EN et al (2018) MoleculeNet: a benchmark for molecular machine learning. Chem Sci 9:513–530. https://doi.org/10.1039/C7SC02664A
19. Akiba T, Sano S, Yanase T et al (2019) Optuna: A Next-generation Hyperparameter Optimization Framework. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, Anchorage AK USA, pp 2623–2631
20. Rong Y, Bian Y, Xu T et al Self-Supervised Graph Transformer on Large-Scale Molecular Data
21. Ahneman DT, Estrada JG, Lin S et al (2018) Predicting reaction performance in C–N cross-coupling using machine learning. Science 360:186–190. https://doi.org/10.1126/science.aar5169
22. Zahrt AF, Henle JJ, Rose BT et al (2019) Prediction of higher-selectivity catalysts by computer-driven workflow and machine learning. Science 363:eaau5631. https://doi.org/10.1126/science.aau5631
23. Qiu J, Xie J, Su S et al (2022) Selective functionalization of hindered meta-C–H bond of o-alkylaryl ketones promoted by automation and deep learning. Chem 8:3275–3287. https://doi.org/10.1016/j.chempr.2022.08.015
24. Perera D, Tucker JW, Brahmbhatt S et al (2018) A platform for automated nanomole-scale reaction screening and micromole-scale synthesis in flow. Science 359:429–434. https://doi.org/10.1126/science.aap9112
25. Wang Z, Wu T, Zeng L et al (2025) Machine Learning Relationships Between Nanoporous Structures and Electrochemical Performance in MOF Supercapacitors. Adv Mater 37:2500943. https://doi.org/10.1002/adma.202500943

Figures

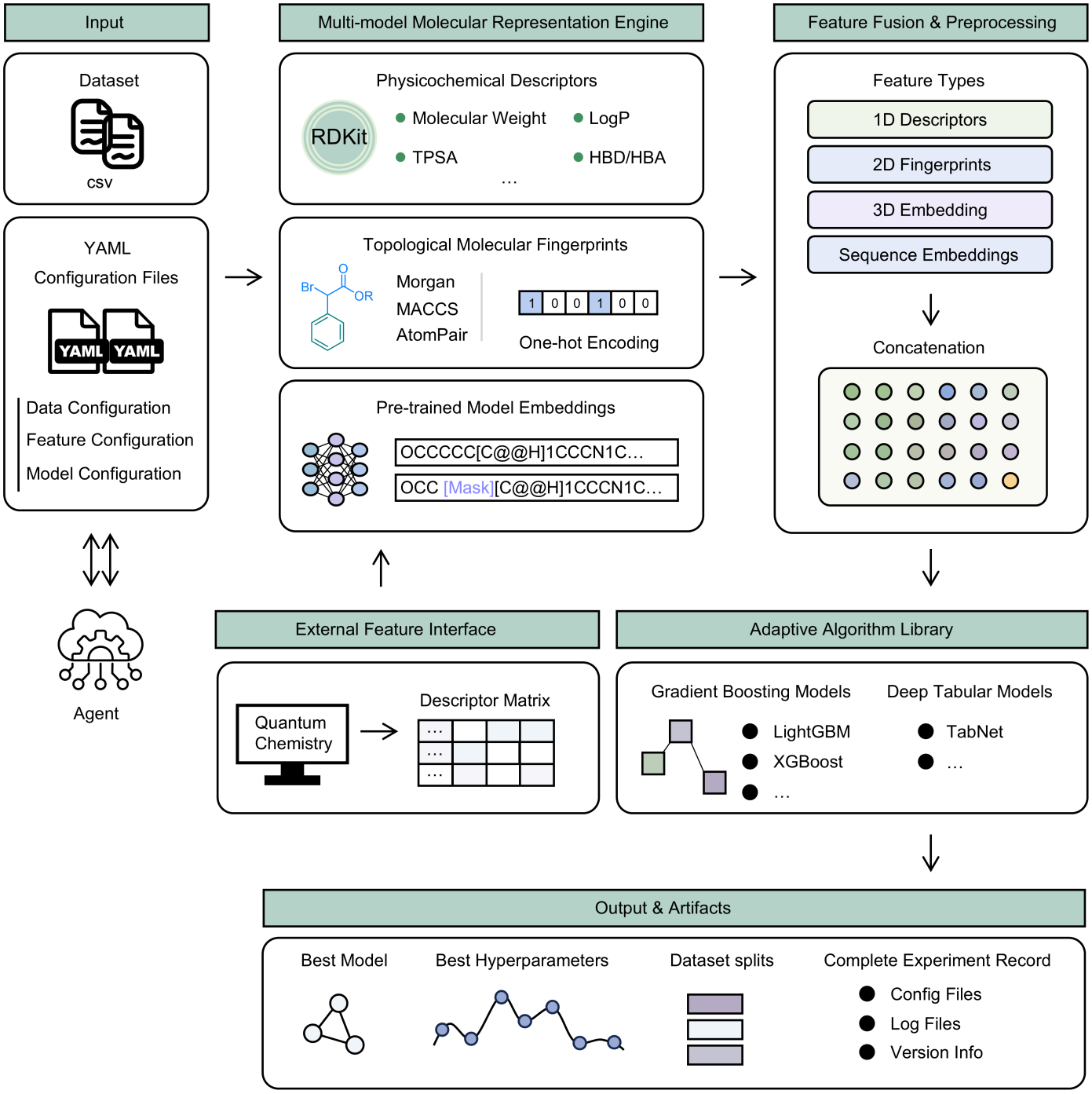

Figure 1

System-Level Architecture and Standardized Workflow of Chemia

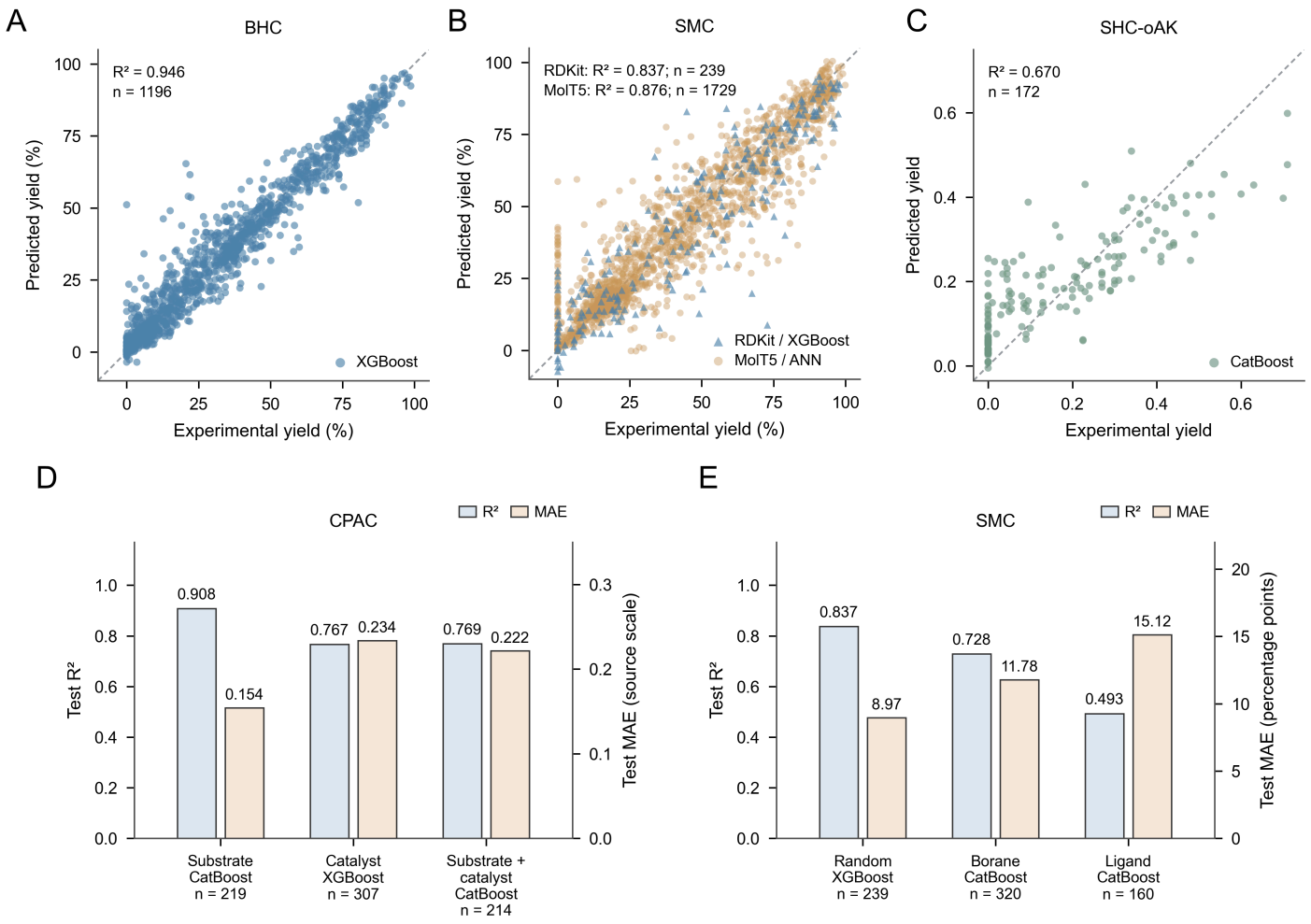

Figure 2

Application of Chemia to Reaction Prediction. Random Split: (A) BHC; (B) SMC; (C) SHC-oAK; OOD Split: (D) CPAC; (E) SMC.

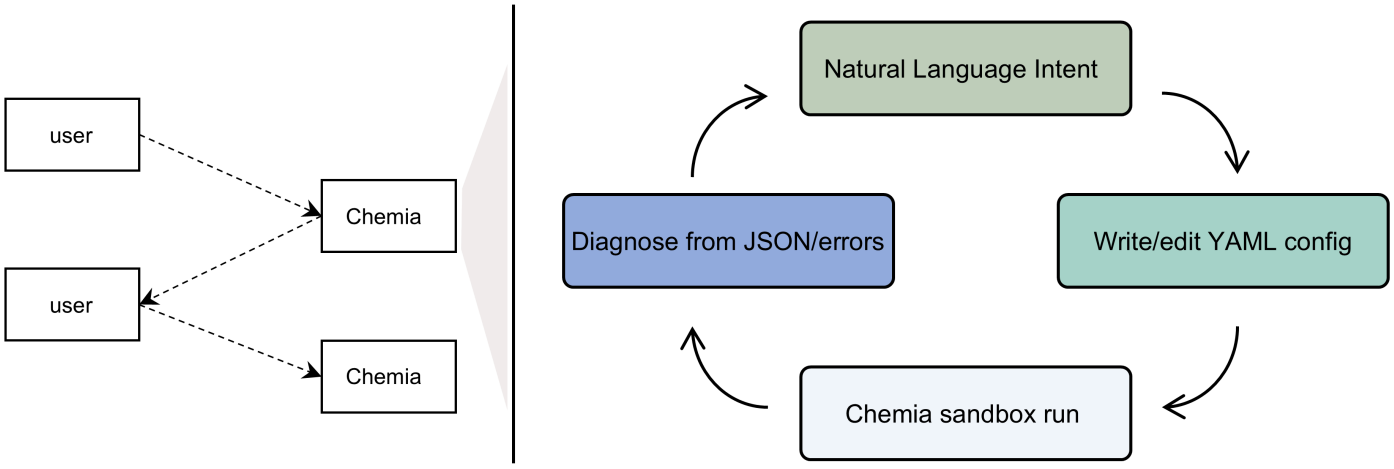

Figure 3

Schematic Illustration of Chemia Interactively Completing a Task.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- SupplementaryInfo.docx
- SupplementaryInformation.pdf