REJECT-TO-PROTECT: DYNAMIC DECISION-AWARE ADVERSARIAL TRAINING FOR SELECTIVE MOLECULAR PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Selective prediction pairs each model output with an accept–defer decision, but most pipelines attach rejection rules to predictors trained independently for accuracy. In molecular property prediction, where structurally similar compounds can have very different prediction harms, such post-hoc strategies still accept risky analogues or reject reliable ones. We introduce Reject-to-Protect (R2P), a decision-aware adversarial training framework that directly optimizes the composed prediction–acceptance policy. R2P builds a frozen, structure-only candidate graph of chemically plausible neighbors. This restricts optimization to the local neighborhoods where selective errors are most likely to occur, and the search stays finite and auditable. Within this graph, a discrete inner maximization searches for two complementary failure modes. Unsafe acceptance (UA) occurs when a harmful prediction is accepted despite its error, while over-rejection (OR) defers a reliable prediction to expert review. An outer minimization then uses these fixed endpoints to train a risk head, and optionally the predictor, under an explicit target-coverage constraint. The risk head is supervised by scaffold out-of-fold prediction harm, and retrieval, supervision, and thresholding are each confined to their own data splits to prevent leakage. On nine MoleculeNet benchmarks at 80% coverage, R2P achieves the lowest mean selective risk on five of six classification tasks and two of three regression tasks, with up to a 33.6% reduction relative to full acceptance.

1 INTRODUCTION

Molecular property prediction helps prioritize compounds before costly synthesis and experimental characterization (Li et al., 2025). However, strong average performance does not ensure that every prediction is reliable enough for automated screening (Kötter et al., 2025). Selective prediction pairs each output with an accept–defer decision, minimizing prediction risk among accepted molecules at a prescribed coverage (Geifman & El-Yaniv, 2017; Chow, 1970). Coverage denotes the fraction of predictions accepted for automated use. Deferral reserves expert or experimental review for selected compounds, making review-budget allocation part of the prediction problem.

Existing methods derive rejection decisions from confidence or uncertainty scores, jointly learn prediction and selection, or use conformal calibration (Hendrycks & Gimpel, 2017; Guo et al., 2017; Geifman & El-Yaniv, 2019; Angelopoulos et al., 2024). A useful acceptance policy must distinguish errors that warrant deferral from accurate predictions that can remain in the automated pathway. This motivates examining whether training on local decision failures can improve the learned acceptance policy.

Molecular uncertainty estimates do not consistently rank prediction errors, and their reliability can vary under distribution shift (Hirschfeld et al., 2020; Antoniuk et al., 2025; Parrondo-Pizarro et al., 2026). Within local chemical neighborhoods, structurally similar molecules can nevertheless have markedly different prediction errors. Structural proximity therefore defines a setting for examining decision differences, without implying that neighboring molecules should receive identical predictions or acceptance decisions. We call accepting a high-error prediction *unsafe acceptance* (UA), and deferring a low-error prediction *over-rejection* (OR). Lowering coverage can reduce unsafe

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.

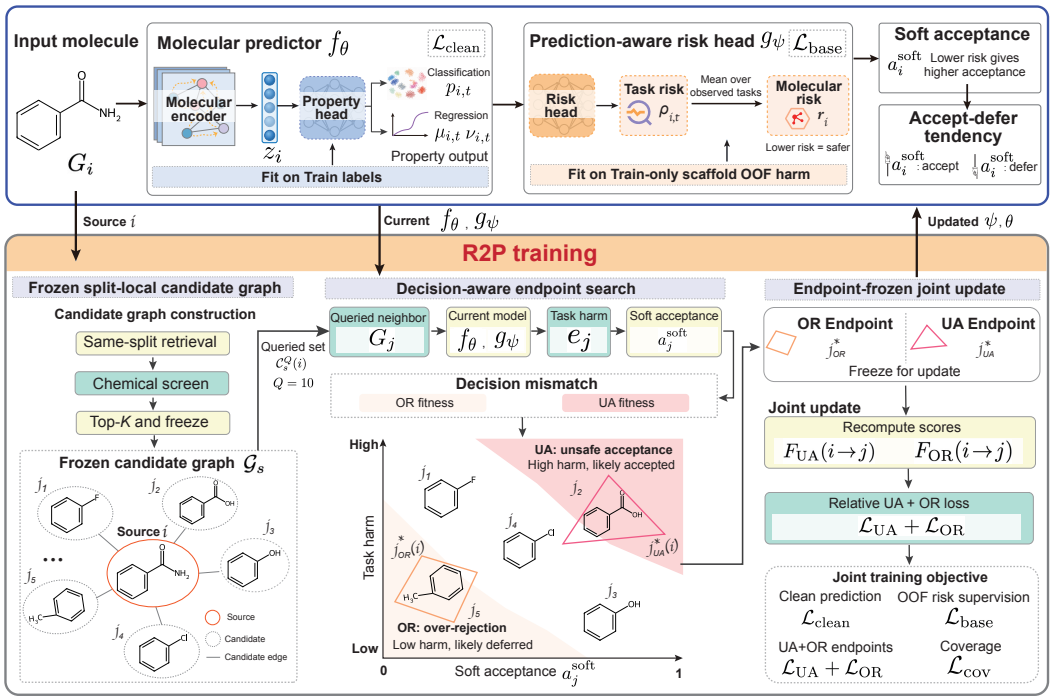

Figure 1: **Overview of R2P for selective molecular prediction.** R2P searches a frozen local candidate graph for unsafe acceptance (UA) and over-rejection (OR) endpoints based on task harm and soft acceptance. The selected endpoint identities are held fixed during the outer update, which jointly optimizes the molecular predictor and risk head under prediction, risk-supervision, and coverage objectives.

acceptance but increases deferral, motivating training that addresses both failures at a prescribed coverage.

We introduce **Reject-to-Protect (R2P)** to reduce prediction harm among accepted molecules at a prescribed coverage. In the configuration evaluated here, R2P jointly trains a molecular predictor and a risk head. Figure 1 summarizes the training procedure. An inner search identifies UA and OR endpoints within a frozen graph of structurally related, measured molecules. Holding the selected endpoint identities fixed, an outer update optimizes the predictor and acceptance policy under a target-coverage penalty. Endpoints are reselected during training to track the model's evolving decision failures. We use R2P for the full method and R2P(UA+OR) to distinguish it from the single-objective variants R2P(UA) and R2P(OR).

Our contributions are as follows.

- A decision-aware training objective combines local UA and OR endpoint losses with prediction, risk-supervision, and coverage terms.
- A structure-based candidate graph defines a finite search space, while Train-only scaffold out-of-fold (OOF) supervision separates risk learning from held-out threshold calibration.
- R2P achieves the lowest mean selective risk among the compared methods on seven of nine MoleculeNet benchmarks at nominal 80% coverage. Ablations reveal dataset-dependent benefits of UA and OR.

2 Related Work

Selective prediction. Confidence-based approaches construct rejection rules for pretrained predictors (Geifman & El-Yaniv, 2017; El-Yaniv & Wiener, 2010). SelectiveNet jointly optimizes prediction and selection under a coverage constraint, whereas Deep Gamblers learns abstention through

a gambling objective (Geifman & El-Yaniv, 2019; Liu et al., 2019). Conformalized quantile regression provides calibrated prediction intervals, while conformal risk control addresses more general monotone losses (Romano et al., 2019; Angelopoulos et al., 2024). R2P focuses on mining local decision failures to train the predictor and acceptance policy.

Uncertainty quantification for molecular property prediction. Molecular uncertainty methods use ensemble disagreement, Bayesian approximations, or learned predictive distributions (Frömbgen et al., 2026; Lakshminarayanan et al., 2017; Soleimany et al., 2021). Benchmark studies examine their calibration and error-ranking performance (Hirschfeld et al., 2020; Lanini et al., 2024; Tang et al., 2025). Under molecular data shifts, chemical distance and predictive variance capture different aspects of reliability (Parrondo-Pizarro et al., 2026; Kim & Vlachos, 2025). Atomic-level and graph-based approaches provide additional reliability signals (Qian et al., 2026; Chen & Li, 2025). R2P supervises a prediction-aware risk score with held-out errors and updates it using local acceptance and deferral failures.

Adversarial learning for molecular property prediction. AFSE and GapCL perturb molecular representations to improve generalization (Yin et al., 2022; Jin et al., 2025). AAIS selects samples for adversarial augmentation using influence functions, while GVAT incorporates unlabeled molecular graphs through virtual adversarial training (Cho et al., 2025; Lu et al., 2025). Uncertainty-guided local perturbations also target reliable molecular prediction (Zhao et al., 2026). R2P instead selects measured molecules from a frozen structural neighborhood and trains on failures of the composed prediction–acceptance policy.

3 METHOD

3.1 PROBLEM DEFINITION

Let $\mathcal{D} = \{(G_i, y_i, \mathcal{T}_i)\}_{i=1}^{N}$ contain standardized molecules, where $G_i = (V_i, E_i, X_i)$ is the atom–bond graph, y_i contains its measured labels, and $\mathcal{T}_i$ is its nonempty set of observed tasks. The index t denotes a task throughout. Whole Bemis–Murcko scaffold groups are assigned to disjoint Train, Calibration, Validation, and Test sets (Bemis & Murcko, 1996), denoted by $\mathcal{D}_{\mathrm{tr}}$, $\mathcal{D}_{\mathrm{cal}}$, $\mathcal{D}_{\mathrm{val}}$, and $\mathcal{D}_{\mathrm{te}}$. We write $N_s = |\mathcal{D}_s|$.

The predictor f_θ returns probabilities $p_{i,t}$ for binary classification. For regression, it returns a mean $\mu_{i,t}$ and log variance $\nu_{i,t}$. The risk head g_ψ produces a nonnegative score r_i, with lower values indicating lower predicted error. Here, harm denotes prediction error rather than a direct measure of clinical or toxicological harm. Task harm and its molecular average are

$$h_{i,t} = \begin{cases} (p_{i,t} - y_{i,t})^2, & \text{classification,} \\ |\mu_{i,t} - y_{i,t}|, & \text{regression,} \end{cases} \qquad h_i = \frac{1}{|\mathcal{T}_i|} \sum_{t \in \mathcal{T}_i} h_{i,t}. \tag{1}$$

Our objective is to reduce accepted-set harm at target coverage c. For hard acceptance decisions $a_i \in \{0, 1\}$ on n molecules with $\sum_i a_i > 0$, coverage and selective harm are

$$\mathrm{Cov} = \frac{1}{n} \sum_i a_i, \qquad \mathrm{SelRisk} = \frac{\sum_i a_i h_i}{\sum_i a_i}. \tag{2}$$

For classification, this definition gives Selective Brier: tasks are averaged within each molecule before accepted molecules are averaged. Regression training uses absolute error, while the main experiments report Selective RMSE computed from squared errors. Appendix B defines the evaluation metrics.

3.2 MODEL AND INITIALIZATION

All R2P variants start from a shared initialization, R2P(Base). We obtain R2P(Base) by fitting the predictor on Train and supervising the risk head with its out-of-fold prediction harm. Figure 1 summarizes the subsequent endpoint search and joint training procedure.

Prediction and risk. Each molecule is mapped to a latent representation z_i. The classification head uses binary cross-entropy. The regression head uses a Gaussian negative log-likelihood with $\nu_{i,t} = \log \sigma_{i,t}^2$. For a minibatch $\mathcal{B}$, the clean prediction loss is

$$\mathcal{L}_{\text{clean}} = \frac{\sum_{i\in\mathcal{B}} \sum_{t\in\mathcal{T}_i} \ell_{i,t}}{\sum_{i\in\mathcal{B}} |\mathcal{T}_i|}, \quad \ell_{i,t} = \begin{cases} \text{BCE}(p_{i,t}, y_{i,t}), & \text{classification,} \\ \frac{1}{2}\left[\nu_{i,t} + (y_{i,t} - \mu_{i,t})^2 e^{-\nu_{i,t}}\right], & \text{regression.} \end{cases} \tag{3}$$

Let $\xi_{i,t}$ contain the prediction features for task t: probability, entropy, and margin for classification, or mean and log variance for regression. The risk head combines these features with the molecular representation and a learned task embedding $\mathbf{v}_t$:

$$\rho_{i,t} = g_\psi(z_i, \xi_{i,t}, \mathbf{v}_t), \qquad r_i = \frac{1}{|\mathcal{T}_i|} \sum_{t\in\mathcal{T}_i} \rho_{i,t}. \tag{4}$$

Here, $\rho_{i,t}$ is the nonnegative task risk. The risk-head architecture and input features are described in Appendix A.2.

Train-only out-of-fold supervision. Train scaffolds are divided into three folds. For each fold k, a shadow predictor trained on the other folds supplies held-out predictions. Classification harm is normalized using a Train-only, task-aware percentile. For regression task t, the normalized out-of-fold error is

$$e_{i,t}^{\text{OOF}} = \frac{|y_{i,t} - \mu_{i,t}^{\text{OOF}}|}{\max(m_t^{\text{OOF}}, \epsilon)}, \qquad m_t^{\text{OOF}} = \text{median}_{j:\, t\in\mathcal{T}_j} |y_{j,t} - \mu_{j,t}^{\text{OOF}}|. \tag{5}$$

In both task types, $\widetilde{h}_i^{\text{OOF}}$ is the mean normalized harm over the observed tasks for molecule i. The risk head is fitted with

$$\mathcal{L}_{\text{base}} = \frac{1}{N_{\text{tr}}} \sum_{i\in\mathcal{D}_{\text{tr}}} \text{SmoothL1}(r_i, \widetilde{h}_i^{\text{OOF}}) + \frac{\lambda_{\text{rank}}}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} [\delta + r_i - r_j]_+, \tag{6}$$

where $\mathcal{P}$ contains pairs ordered from lower to higher out-of-fold harm, δ is the ranking margin, and $[x]_+ = \max(0, x)$. Targets, normalizers, and pair ordering are detached from the gradient. R2P(Base) is retained without further training for the experimental comparison.

3.3 Local decision-aware endpoint search

Frozen candidate graph. For each split s, we construct a graph of measured molecules,

$$\mathcal{G}_s = (\mathcal{V}_s, \mathcal{A}_s), \qquad \mathcal{C}_s(i) = \{j : (i,j) \in \mathcal{A}_s\}. \tag{7}$$

For each split, candidate molecules are ranked by split-local ECFP4 Tanimoto similarity and screened for chemical validity, configured edit constraints, endpoint compatibility, and observed-label availability. The top-K eligible candidates under this ranking form $\mathcal{C}_s(i)$, and the resulting graph is frozen before training. Message passing is performed on each molecule's atom–bond graph G_i; the candidate graph supplies molecules for endpoint search. During training, endpoint search evaluates a queried subset defined by

$$\mathcal{C}_s^Q(i) \subseteq \mathcal{C}_s(i), \qquad |\mathcal{C}_s^Q(i)| = \min\{Q, |\mathcal{C}_s(i)|\}. \tag{8}$$

Small candidate sets are fully enumerated, whereas larger sets follow a deterministic query order and retain the candidate with the highest endpoint score among those evaluated. Thus, top-K determines the frozen graph neighborhood, while Q controls the number of candidates considered during endpoint search. Appendix A.1 provides the complete retrieval and graph-construction protocol.

Soft acceptance. At training step u, we center the risks using a detached Train mean:

$$\bar{r}_u = \text{sg}\left(\frac{1}{N_{\text{tr}}} \sum_{i\in\mathcal{D}_{\text{tr}}} r_i^{(u)}\right), \qquad \widetilde{r}_i^{(u)} = r_i^{(u)} - \bar{r}_u. \tag{9}$$

Here, sg denotes stop-gradient and u indexes training steps. Let τ_u be the detached c-quantile of centered Train risks and s_u their detached median absolute deviation. Soft acceptance is

$$a_i^{\mathrm{soft},(u)} = \sigma\left(\frac{\tau_u - \widetilde{r}_i^{(u)}}{T_a \max(s_u, \epsilon)}\right), \qquad \mathcal{L}_{\mathrm{cov}} = \left(\frac{1}{N_{\mathrm{tr}}} \sum_{i \in \mathcal{D}_{\mathrm{tr}}} a_i^{\mathrm{soft},(u)} - c\right)^2. \tag{10}$$

Here, σ is the sigmoid function and $\epsilon > 0$ ensures numerical stability. We use target coverage $c = 0.80$ and temperature $T_a = 1$. We omit the step index below when all quantities refer to the same model state.

Unsafe acceptance and over-rejection. The adversary searches $\mathcal{C}_{\mathrm{tr}}(i)$ for local acceptance errors. Let e_j be the detached harm used for endpoint search: bounded Brier harm for classification or normalized absolute error for regression. The two scores are

$$F_{\mathrm{UA}}^{\mathrm{search}}(i \to j) = a_j^{\mathrm{soft}} \begin{cases} e_j, & \text{classification,} \\ \log(1 + e_j), & \text{regression,} \end{cases} \tag{11}$$

$$F_{\mathrm{OR}}^{\mathrm{search}}(i \to j) = (1 - a_j^{\mathrm{soft}}) \begin{cases} 1 - e_j, & \text{classification,} \\ \exp(-e_j), & \text{regression.} \end{cases} \tag{12}$$

UA favors high-harm candidates with high acceptance, whereas OR favors low-harm candidates with low acceptance. For each direction $q \in \{\mathrm{UA}, \mathrm{OR}\}$, we select

$$j_q^*(i) = \underset{j \in \mathcal{C}_{\mathrm{tr}}^Q(i)}{\arg\max} F_q^{\mathrm{search}}(i \to j). \tag{13}$$

Canonical molecule ID breaks ties. The selected identities are detached and held fixed during the next optimizer update.

Connection to selective risk. UA and OR jointly favor reliable acceptance at fixed coverage. For fixed classification harms $h_i \in [0, 1]$ and $m = \sum_i a_i > 0$ hard acceptances, their global scores satisfy

$$\sum_i \big[a_i h_i + (1 - a_i)(1 - h_i)\big] = 2m \ \mathrm{SelRisk} + \sum_i (1 - h_i) - m. \tag{14}$$

At fixed m, minimizing the left side therefore minimizes selective risk. This relation motivates the two decision scores. R2P concentrates supervision on graph-local endpoints and uses $\mathcal{L}_{\mathrm{cov}}$ to control mean soft acceptance. Appendix A.4 gives the derivation and the scope of this connection.

3.4 Endpoint-frozen loss and min–max update

R2P alternates between discrete maximization of the search scores and joint optimization of an outer surrogate. The selected molecule indices stay fixed for the outer update. Their predictions and risks are recomputed with gradients to obtain the outer scores F_q, matching the notation in Figure 1. We use F_q^{search} for the detached search scores and F_q for their outer-update counterparts. These scores differ in their gradient paths and, for OR, in the harm reference used.

UA supplies gradients to lower prediction harm and reduce acceptance of harmful endpoints. OR uses detached OOF harm as its reliability reference and supplies an acceptance gradient for reliable endpoints. Holding this reference fixed removes the error-increasing gradient of the OR harm factor. Both acceptance paths reach the risk head and predictor through their shared computation. Appendix A.3 defines the outer scores, and Appendix A.4 derives these gradient directions.

Let $\mathcal{S} = \{i \in \mathcal{D}_{\mathrm{tr}} : |\mathcal{C}_{\mathrm{tr}}(i)| > 0\}$ contain the sources with valid candidates. The relative endpoint loss is

$$\mathcal{L}_q = \frac{1}{|\mathcal{S}|} \sum_{i \in \mathcal{S}} \left[F_q(i \to j_q^*(i)) - \mathrm{sg}\left(F_q(i \to i)\right)\right]_+^2. \tag{15}$$

This squared hinge penalizes violations of the local requirement $F_q(i \to j_q^*) \le F_q(i \to i)$. The detached source score sets the reference for the current update. An active penalty differentiates through the endpoint score, with weight proportional to its excess above the reference. Appendix A.4 derives the gradient. The source is excluded from its candidate set, and $\mathcal{L}_q = 0$ when $\mathcal{S}$ is empty.

The full objective is

$$\mathcal{L}_{\mathrm{R2P}} = \lambda_{\mathrm{clean}}\mathcal{L}_{\mathrm{clean}} + \lambda_{\mathrm{base}}\mathcal{L}_{\mathrm{base}} + \lambda_{\mathrm{cov}}\mathcal{L}_{\mathrm{cov}} + \lambda_{\mathrm{UA}}\mathcal{L}_{\mathrm{UA}} + \lambda_{\mathrm{OR}}\mathcal{L}_{\mathrm{OR}}. \tag{16}$$

The full method uses unit weights. R2P(UA) sets $\lambda_{\mathrm{OR}} = 0$, and R2P(OR) sets $\lambda_{\mathrm{UA}} = 0$. The clean loss encourages accurate predictions, the base loss maintains supervision from held-out harm, and the coverage penalty discourages excessive rejection.

The outer update combines active loss gradients using equal-norm PCGrad and updates θ and ψ. The next search reevaluates candidates under the updated model. This min–max procedure couples discrete adversarial search with a differentiable outer surrogate. Algorithm 1 shows the computations and stop-gradient boundaries.

3.5 Training and evaluation

R2P learns from Train and selects checkpoints on Validation. The scaffold out-of-fold harm bank is refreshed every 10 epochs for classification and every 10 optimizer steps for regression. For each candidate checkpoint, we fit an acceptance threshold on Calibration before evaluating it on Validation. The selected model and policy are frozen for Test.

For deployment, Calibration sets an inclusive threshold from its ordered risks:

$$\tau_{\mathrm{cal}} = r^{\mathrm{cal}}_{(\lceil cN_{\mathrm{cal}} \rceil)}, \qquad a_i = \mathbb{I}[r_i \le \tau_{\mathrm{cal}}]. \tag{17}$$

The threshold depends on ordered Calibration risk scores. Its realized Test coverage varies with ties and distribution shift.

The main experiments compare methods at a matched acceptance count. They accept the $\lceil cN_{\mathrm{te}} \rceil$ Test molecules with the lowest risk scores. Appendix B defines the fixed-coverage evaluation metrics. This ranking-based evaluation differs from applying the frozen Calibration threshold at deployment.

4 Experiments

We evaluate R2P on nine molecular property prediction benchmarks. Our experiments examine selective risk at fixed coverage, the contributions of the two training objectives, and variation in performance across tasks.

4.1 Experimental Setup

Datasets and protocol. We evaluate six classification and three regression datasets from MoleculeNet (Wu et al., 2018), using Bemis–Murcko scaffold-disjoint Train, Calibration, Validation, and Test splits with target proportions of 60%, 15%, 10%, and 15% (Bemis & Murcko, 1996). Train fits the model, Calibration fits deployment thresholds, Validation selects checkpoints, and Test is reserved for final evaluation. All methods use the same communicative message-passing neural network (CMPNN) backbone (Song et al., 2020), splits, and seeds. For all selective methods, we set the nominal target coverage to 80% and accept the $\lceil 0.8N_{\mathrm{te}} \rceil$ test molecules with the lowest risk scores, ensuring the same acceptance count across methods. We use Selective Brier for classification and Selective RMSE for regression. For multitask classification, Brier harm is first averaged over the observed tasks within each molecule. Dataset statistics and implementation details are given in Appendix B.

Baselines. We compare R2P with several selective-prediction baselines. *Full* accepts all molecules and provides a reference at 100% coverage. *Random* assigns random ranks before rejection. *Embedding distance* ranks molecules by their distance from the Train-set centroid in representation space. *Split conformal* ranks predictions by calibrated set size for classification and interval width for regression. The latter uses conformalized quantile regression (Romano et al., 2019). *Confidence* (maximum class probability, or MCP) ranks classification predictions by their largest class probability; its regression analogue is a standardized-residual proxy based on the predicted mean. These are classical single-score selective-prediction baselines; none uses Test labels. Implementation details are given in Appendix B.

Table 1: Selective risk at 80% coverage, reported as mean ± sample standard deviation over three seeds. Classification uses Selective Brier and regression uses Selective RMSE. Lower is better. Bold marks the lowest mean among selective methods. Full provides the reference at 100% coverage.

Classification: Selective Brier ↓						
Dataset	**Full**	**Random**	**Emb. dist.**	**Conformal**	**Confidence**	**R2P**
BACE	0.1779 ± 0.0121	0.1785 ± 0.0120	0.1869 ± 0.0059	0.1611 ± 0.0111	**0.1589 ± 0.0139**	0.1637 ± 0.0124
BBBP	0.1151 ± 0.0220	0.1148 ± 0.0220	0.1397 ± 0.0275	0.1169 ± 0.0079	0.0903 ± 0.0210	**0.0887 ± 0.0194**
ClinTox	0.0678 ± 0.0039	0.0682 ± 0.0038	0.0482 ± 0.0025	0.0773 ± 0.0023	0.0465 ± 0.0017	**0.0450 ± 0.0018**
SIDER	0.1671 ± 0.0031	0.1669 ± 0.0031	0.1690 ± 0.0046	0.1654 ± 0.0044	0.1640 ± 0.0033	**0.1632 ± 0.0030**
Tox21	0.0901 ± 0.0019	0.0902 ± 0.0020	0.0869 ± 0.0180	0.1006 ± 0.0031	0.0743 ± 0.0009	**0.0737 ± 0.0024**
ToxCast	0.0629 ± 0.0015	0.0631 ± 0.0015	0.0626 ± 0.0069	0.0710 ± 0.0006	0.0556 ± 0.0024	**0.0549 ± 0.0043**
Regression: Selective RMSE ↓						
Dataset	**Full**	**Random**	**Emb. dist.**	**Conformal**	**R2P**	
ESOL	0.6888 ± 0.0054	0.6889 ± 0.0052	0.6290 ± 0.0111	0.6439 ± 0.0264	**0.6247 ± 0.0239**	
FreeSolv	1.3993 ± 0.1252	1.3948 ± 0.1220	1.2922 ± 0.1102	**1.1177 ± 0.0830**	1.2003 ± 0.1627	
Lipophilicity	0.6479 ± 0.0180	0.6466 ± 0.0180	0.6096 ± 0.0276	0.6060 ± 0.0173	**0.5992 ± 0.0211**	

Table 2: Component ablation at 80% coverage. Values are mean ± sample standard deviation over three seeds. Bold marks the lowest mean in each row, including ties at the displayed precision.

Dataset	**R2P(Base)**	**R2P(UA)**	**R2P(OR)**	**R2P(UA+OR)**
Classification: Selective Brier ↓				
BACE	0.1812 ± 0.0145	0.1707 ± 0.0137	0.1691 ± 0.0201	**0.1637 ± 0.0124**
BBBP	0.0893 ± 0.0027	**0.0762 ± 0.0089**	0.0824 ± 0.0070	0.0887 ± 0.0194
ClinTox	**0.0450 ± 0.0018**	0.0466 ± 0.0036	0.0453 ± 0.0037	**0.0450 ± 0.0018**
SIDER	0.1653 ± 0.0019	0.1653 ± 0.0019	0.1661 ± 0.0037	**0.1632 ± 0.0030**
Tox21	0.0724 ± 0.0015	**0.0714 ± 0.0006**	0.0732 ± 0.0018	0.0737 ± 0.0024
ToxCast	0.0574 ± 0.0013	0.0574 ± 0.0013	0.0574 ± 0.0013	**0.0549 ± 0.0043**
Regression: Selective RMSE ↓				
ESOL	0.6544 ± 0.0291	**0.6233 ± 0.0427**	0.6245 ± 0.0138	0.6247 ± 0.0239
FreeSolv	1.2334 ± 0.1254	1.2334 ± 0.1254	1.2170 ± 0.1423	**1.2003 ± 0.1627**
Lipophilicity	0.6095 ± 0.0132	0.6022 ± 0.0173	0.6073 ± 0.0134	**0.5992 ± 0.0211**

4.2 Main Results

Table 1 shows that R2P reduces selective risk relative to Full acceptance on all nine benchmarks and relative to Random selection on every benchmark. The close values of Full and Random indicate that rejecting an arbitrary subset does not reliably target harmful predictions. Embedding-distance selection yields lower mean risk than Full on six datasets, but R2P yields lower mean risk than embedding-distance selection on all nine datasets. Compared with the score-based baselines, R2P has lower mean risk than Confidence on five of six classification benchmarks and lower mean risk than split conformal on seven of nine benchmarks. Confidence remains best on BACE, while split conformal remains best on FreeSolv. Overall, R2P achieves the lowest mean risk on five classification and two regression benchmarks, with reductions of up to 33.6% relative to Full acceptance, comparing 80% with 100% coverage.

4.3 Component Ablation

To examine the contributions of the two decision objectives, we compare four variants from the same R2P(Base) initialization described in Section 3.2. *R2P(Base)* retains the initial model without further training, whereas all dynamic variants retain the clean, risk-supervision, and coverage losses. *R2P(UA)* activates only the UA endpoint loss, and *R2P(OR)* activates only the OR endpoint loss. *R2P(UA+OR)* combines both objectives. All variants use the same data splits, candidate graphs, and random seeds.

Effect of UA and OR training. R2P(UA+OR) reduces mean selective risk relative to R2P(Base) on seven of the nine datasets, ties on ClinTox, and increases risk on Tox21. It achieves the lowest

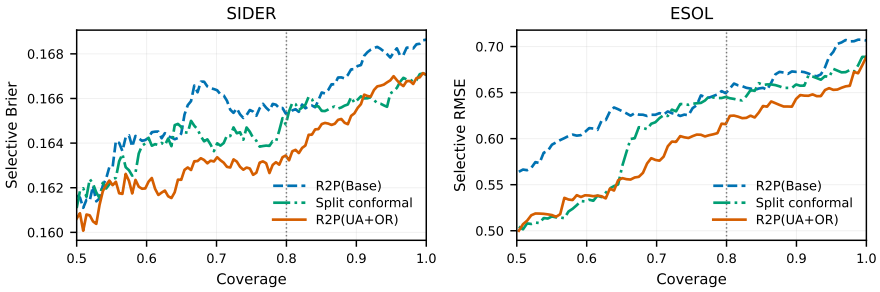

Figure 2: Risk–coverage curves on SIDER and ESOL. Lines show mean Test risk over three seeds; lower is better. The dashed blue curve denotes R2P(Base), the solid orange curve denotes joint UA+OR training, and the green curve denotes split conformal. Vertical dotted lines mark 80% coverage.

mean on six datasets, including a tie with R2P(Base) on ClinTox at the reported precision. R2P(UA) has the lowest mean on BBBP, Tox21, and ESOL. Adding OR to UA further reduces selective risk on six datasets, indicating that the two objectives can provide complementary training signals, although their relative benefit varies across tasks.

4.4 Risk–Coverage Analysis

We evaluate predictions across a range of coverage levels rather than only at the 80% operating point. Figure 2 shows one classification and one regression benchmark. For each method, we vary the number of accepted molecules from half of the Test set to the full set while keeping its risk ranking fixed. Each curve reports the mean risk over three seeds.

Joint training stays below R2P(Base) across the plotted range in both benchmarks. At 80% coverage, it reduces Selective Brier from 0.1653 to 0.1632 on SIDER and Selective RMSE from 0.6544 to 0.6247 on ESOL. The difference remains at full coverage, when every molecule is accepted, which shows that the joint objective also changes the underlying predictor and not only the ordering of Test molecules. Comparison with split conformal shows that R2P remains effective across a broad range of coverage levels. On SIDER, the joint model has lower risk over most of the plotted range and is overtaken only near full coverage. On ESOL, its advantage becomes clear above roughly 65% coverage. Averaged across the plotted coverage levels, R2P achieves lower risk than split conformal on both benchmarks, providing a more complete comparison of the methods across operating conditions.

4.5 Safety-Relevant Case Analysis

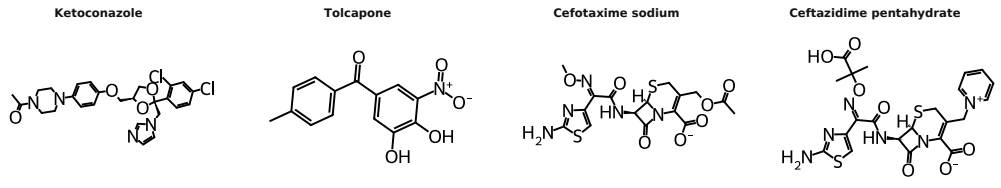

Figure 3: Structures of the four ToxCast Test molecules used in the safety-relevant audit and local analogue analysis.

Table 3: Retrospective ToxCast audit and local analogue analysis for Figure 3. Molecule Brier averages the observed assays for each molecule; lower is better. Risk rank orders predicted risk from low to high. At 80% coverage, ranks 1–1,030 are accepted. For Ketoconazole and Tolcapone, the observed-assay column reports the number of observed assays; for the local analogue pair, it reports positive calls over observed tasks.

Case	Method	Observed assays	Molecule Brier ↓	Risk rank	Decision
Ketoconazole	R2P(Base)	452	0.2111	888/1,287	Accept
	R2P(UA+OR)	452	0.2095	1,224/1,287	Defer
Tolcapone	R2P(Base)	249	0.1105	802/1,287	Accept
	R2P(UA+OR)	249	0.0981	1,047/1,287	Defer
Cefotaxime sodium	R2P(Base)	0/57	0.0253	1,229/1,287	Defer
	R2P(UA+OR)	0/57	0.0072	812/1,287	Accept
Ceftazidime pentahydrate	R2P(Base)	2/57	0.0558	1,180/1,287	Defer
	R2P(UA+OR)	2/57	0.0483	1,049/1,287	Defer

Safety-relevant held-out audit. Figure 3 and Table 3 show two exact-structure matches from the ToxCast seed-1 Test split. At the fixed 80% coverage, the joint R2P model moves both compounds out of the automatically accepted set. Ketoconazole moves from rank 888 to 1,224, and Tolcapone moves from rank 802 to 1,047. Their molecule-level Brier errors are high relative to the risks of the accepted sets. For Ketoconazole, R2P(Base) gives a Brier score of 0.2111, which is 3.65 times the R2P(Base) accepted-set mean. The joint R2P model gives 0.2095, which is 4.18 times its accepted-set mean. ToxCast records HepG2 cell-loss and cellular-stress signals for this compound, and LiverTox reports severe acute clinically apparent liver injury associated with oral ketoconazole (National Institute of Diabetes and Digestive and Kidney Diseases, 2017). For Tolcapone, R2P(Base) gives a Brier score of 0.1105, which is 1.91 times its accepted-set mean. The joint R2P model gives 0.0981, which is 1.96 times its accepted-set mean. LiverTox reports severe or fatal acute liver injury associated with tolcapone (National Institute of Diabetes and Digestive and Kidney Diseases, 2021). Appendix B.4 describes the post-hoc audit protocol.

Local analogue separation. The preceding audit examines unsafe acceptances that R2P moves to deferral. We next consider the complementary correction: accepting a low-harm target while retaining deferral of a structurally similar neighbor. Under the same candidate-construction rule, Cefotaxime sodium and Ceftazidime pentahydrate form a bidirectional local relation with ECFP4 similarity 0.6207, 11 atom edits, and 8 bond edits.

Figure 3 and Table 3 show that R2P(Base) defers both molecules, whereas R2P(UA+OR) separates their decisions. Cefotaxime sodium, with no positive label among 57 observed assays, moves from rank 1,229 to 812 and is accepted; its molecule-level Brier decreases from 0.0253 to 0.0072. Ceftazidime pentahydrate remains deferred at rank 1,049 and has the higher Brier value of 0.0483. Its two positive assay calls correspond to the Phase-2 p53 beta-lactamase channel-1 readout of uncleaved substrate and the ATP-based viability counter-screen. The resulting accept–defer split demonstrates that R2P can distinguish locally similar molecules using their prediction-risk profiles rather than assigning the same decision to both structures.

5 Conclusion

We introduced Reject-to-Protect (R2P), which searches a frozen, split-local graph for unsafe-acceptance and over-rejection endpoints. The selected endpoints guide joint training of a molecular predictor and a prediction-aware risk head with Train-only out-of-fold supervision and a target-coverage penalty. At nominal 80% coverage, R2P achieves the lowest mean selective risk on five of the six classification benchmarks and two of the three regression benchmarks among the compared methods. The largest reduction relative to full acceptance is 33.6%, comparing 80% with 100% coverage. The results show lower mean selective risk on most evaluated benchmarks, with benefits that vary across datasets. Future work will investigate adaptive, uncertainty-aware neighborhoods in sparsely sampled or previously unseen regions of chemical space.

Reproducibility Statement

Code and implementation details for reproducing the experiments are available at: https://anonymous.4open.science/r/R2P-C79A. All datasets used in this study are included in the repository and are also publicly available from the original sources described in the manuscript.

AI use statement

Generative AI tools were used solely to aid in language polishing and improving the readability of the manuscript. They were not used in the development of the methods, experimental design, data processing, analysis, or interpretation of results. The authors reviewed all AI-assisted edits and take full responsibility for the final manuscript.

Ethics statement

This work uses only publicly available molecular datasets. No new human, animal, or wet-lab experiments are conducted. The proposed method is intended for computational research and molecular property prediction, rather than direct clinical decision making.

References

Anastasios N. Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/f3549ef9b5ff520a7e41ff3cc306ab2b-Abstract-Conference.html.

Evan Antoniuk, Shehtab Zaman, Tal Ben-Nun, Peggy Li, James Diffenderfer, Busra Sahin, Obadiah Smolenski, Everett Grethel, Tim Hsu, Anna Hiszpanski, Kenneth Chiu, Bhavya Kailkhura, and Brian Van Essen. BOOM: Benchmarking out-of-distribution molecular property predictions of machine learning models. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-4245. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/b94263f7f98c9766ea9a09761ddd88ee-Abstract-Datasets_and_Benchmarks_Track.html.

Guy W. Bemis and Mark A. Murcko. The properties of known drugs. 1. molecular frameworks. *Journal of Medicinal Chemistry*, 39(15):2887–2893, 1996. doi: 10.1021/jm9602928. URL https://doi.org/10.1021/jm9602928.

Lung-Yi Chen and Yi-Pei Li. Uncertainty quantification with graph neural networks for efficient molecular design. *Nature Communications*, 16:3262, 2025. doi: 10.1038/s41467-025-58503-0. URL https://doi.org/10.1038/s41467-025-58503-0.

Soyoung Cho, Sungchul Hong, and Jong-June Jeon. Adaptive adversarial augmentation for molecular property prediction. *Expert Systems with Applications*, 270:126512, 2025. doi: 10.1016/j.eswa.2025.126512. URL https://doi.org/10.1016/j.eswa.2025.126512.

C. K. Chow. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1):41–46, 1970. doi: 10.1109/TIT.1970.1054406. URL https://doi.org/10.1109/TIT.1970.1054406.

John S. Delaney. Esol: Estimating aqueous solubility directly from molecular structure. *Journal of Chemical Information and Computer Sciences*, 44(3):1000–1005, 2004. doi: 10.1021/ci034243x. URL https://doi.org/10.1021/ci034243x.

Ran El-Yaniv and Yair Wiener. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11:1605–1641, 2010. URL https://www.jmlr.org/papers/v11/el-yaniv10a.html.

Tom Frömbgen, Elizaveta Surzhikova, Jürgen Dölz, Jonny Proppe, Barbara Kirchner, and Christoph R. Jacob. Uncertainty quantification for in silico chemistry. *Chemical Reviews*, 126(7): 4189–4236, 2026. doi: 10.1021/acs.chemrev.5c00931. URL https://doi.org/10.1021/acs.chemrev.5c00931.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/4a8423d5e91fda00bb7e46540e2b0cf1-Abstract.html.

Yonatan Geifman and Ran El-Yaniv. Selectivenet: A deep neural network with an integrated reject option. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pp. 2151–2159, 2019. URL https://proceedings.mlr.press/v97/geifman19a.html.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pp. 1321–1330, 2017. URL https://proceedings.mlr.press/v70/guo17a.html.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/pdf?id=VU4h3siRAw.

Lior Hirschfeld, Kyle Swanson, Kevin Yang, Regina Barzilay, and Connor W. Coley. Uncertainty quantification using neural networks for molecular property prediction. *Journal of Chemical Information and Modeling*, 60(8):3770–3780, 2020. doi: 10.1021/acs.jcim.0c00502. URL https://doi.org/10.1021/acs.jcim.0c00502.

Shuting Jin, Xiangrong Liu, Junlin Xu, Sisi Yuan, Hongxing Xiang, Lian Shen, Chunyan Li, Zhangming Niu, and Yinhui Jiang. Adaptive symmetry-based adversarial perturbation augmentation for molecular graph representations with dual-fusion attention information. *Information Fusion*, 120: 103062, 2025. doi: 10.1016/j.inffus.2025.103062. URL https://doi.org/10.1016/j.inffus.2025.103062.

Jae Young Kim and Dionisios G. Vlachos. Distance-aware molecular property prediction in nonlinear structure–property space. *Journal of Chemical Information and Modeling*, 65(13):6744–6756, 2025. doi: 10.1021/acs.jcim.5c01037. URL https://doi.org/10.1021/acs.jcim.5c01037.

Alex Kötter, Kanishka Singh, Hans Matter, Gerhard Hessler, and Christoph Grebner. Upgrading reliability in molecular property prediction by robust quantification of uncertainty from machine learning models. *Journal of Chemical Information and Modeling*, 65(20):10819–10831, 2025. doi: 10.1021/acs.jcim.5c00464. URL https://doi.org/10.1021/acs.jcim.5c00464.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/9ef2ed4b7fd2c810847ffa5fa85bce38-Abstract.html.

Jessica Lanini, Minh Tam Davide Huynh, Gaetano Scebba, Nadine Schneider, and Raquel Rodríguez-Pérez. UNIQUE: A framework for uncertainty quantification benchmarking. *Journal of Chemical Information and Modeling*, 64(22):8379–8386, 2024. doi: 10.1021/acs.jcim.4c01578. URL https://doi.org/10.1021/acs.jcim.4c01578.

Junyao Li, Jianmei Zhang, Rui Guo, Jiawei Dai, Zhiqiang Niu, Yan Wang, Taoyun Wang, Xiaojian Jiang, and Weicheng Hu. Progress of machine learning in the application of small molecule druggability prediction. *European Journal of Medicinal Chemistry*, 285:117269, 2025. doi: 10.1016/j.ejmech.2025.117269. URL https://doi.org/10.1016/j.ejmech.2025.117269.

Ziyin Liu, Zhikang Wang, Paul Pu Liang, Ruslan Salakhutdinov, Louis-Philippe Morency, and Masahito Ueda. Deep gamblers: Learning to abstain with portfolio theory. In *Advances in Neural Information Processing Systems*, volume 32,

2019. URL https://proceedings.neurips.cc/paper/2019/hash/0c4b1eeb45c90b52bfb9d07943d855ab-Abstract.html.

Yong Lu, Chenxu Wang, Ze Wang, Xukun Zhang, Guifei Zhou, and Chunyan Li. Semi-supervised learning-based virtual adversarial training on graph for molecular property prediction. *Alexandria Engineering Journal*, 115:491–500, 2025. doi: 10.1016/j.aej.2024.11.110. URL https://doi.org/10.1016/j.aej.2024.11.110.

David L. Mobley and J. Peter Guthrie. Freesolv: A database of experimental and calculated hydration free energies, with input files. *Journal of Computer-Aided Molecular Design*, 28(7): 711–720, 2014. doi: 10.1007/s10822-014-9747-x. URL https://doi.org/10.1007/s10822-014-9747-x.

National Institute of Diabetes and Digestive and Kidney Diseases. Ketoconazole. LiverTox: Clinical and Research Information on Drug-Induced Liver Injury, 2017. URL https://www.ncbi.nlm.nih.gov/books/NBK547869/.

National Institute of Diabetes and Digestive and Kidney Diseases. Tolcapone. LiverTox: Clinical and Research Information on Drug-Induced Liver Injury, 2021. URL https://www.ncbi.nlm.nih.gov/books/NBK548573/.

Raquel Parrondo-Pizarro, Jessica Lanini, and Raquel Rodríguez-Pérez. Uncertainty quantification in molecular machine learning for property predictions under data shifts. *Journal of Chemical Information and Modeling*, 66(2):923–935, 2026. doi: 10.1021/acs.jcim.5c02381. URL https://doi.org/10.1021/acs.jcim.5c02381.

Jiayu Qian, Yukun Luo, Qingping Zhou, and Xiaoqing Guan. Atomic uncertainty pinpoints critical failure structures for trustworthy molecular property prediction. *Journal of Chemical Information and Modeling*, 66(14):8076–8086, 2026. doi: 10.1021/acs.jcim.6c00973. URL https://doi.org/10.1021/acs.jcim.6c00973.

David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling*, 50(5):742–754, 2010. doi: 10.1021/ci100050t. URL https://doi.org/10.1021/ci100050t.

Yaniv Romano, Evan Patterson, and Emmanuel Candès. Conformalized quantile regression. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/5103c3584b063c431bd1268e9b5e76fb-Abstract.html.

Ava P. Soleimany, Alexander Amini, Samuel Goldman, Daniela Rus, Sangeeta N. Bhatia, and Connor W. Coley. Evidential deep learning for guided molecular property prediction and discovery. *ACS Central Science*, 7(8):1356–1367, 2021. doi: 10.1021/acscentsci.1c00546. URL https://doi.org/10.1021/acscentsci.1c00546.

Ying Song, Shuangjia Zheng, Zhangming Niu, Zhang-hua Fu, Yutong Lu, and Yuedong Yang. Communicative representation learning on attributed molecular graphs. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, pp. 2831–2838, 2020. doi: 10.24963/ijcai.2020/392. URL https://www.ijcai.org/proceedings/2020/392.

Hao Tang, Tianle Yue, and Ying Li. Assessing uncertainty in machine learning for polymer property prediction: A benchmark study. *Journal of Chemical Information and Modeling*, 65(13):6585–6598, 2025. doi: 10.1021/acs.jcim.5c00550. URL https://doi.org/10.1021/acs.jcim.5c00550.

Zhenqin Wu, Bharath Ramsundar, Evan N. Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S. Pappu, Karl Leswing, and Vijay Pande. Moleculenet: A benchmark for molecular machine learning. *Chemical Science*, 9(2):513–530, 2018. doi: 10.1039/C7SC02664A. URL https://doi.org/10.1039/C7SC02664A.

Yueming Yin, Haifeng Hu, Zhen Yang, Feihu Jiang, Yihe Huang, and Jiansheng Wu. Afse: towards improving model generalization of deep graph learning of ligand bioactivities targeting gpcr proteins. *Briefings in Bioinformatics*, 23(3):bbac077, 2022. doi: 10.1093/bib/bbac077. URL https://doi.org/10.1093/bib/bbac077.

Xuxin Zhao, Wen Zhang, and Shichao Liu. Uncertainty-guided adaptive local adversarial perturbation for reliable molecular property prediction. In *Proceedings of the Thirty-Fifth International Joint Conference on Artificial Intelligence*, pp. 5387–5395, 2026. doi: 10.24963/ijcai.2026/600. URL https://www.ijcai.org/proceedings/2026/600.

A Additional Method Details

This appendix gives implementation details for R2P and its UA and OR ablations. All dynamic variants jointly update the predictor and risk head and periodically refresh Train-only scaffold out-of-fold harm. Algorithm 1 gives the training computations. Section A.4 derives the connection to selective risk and the gradients of the endpoint objectives.

A.1 Standardization, splitting, and candidate graph

Molecular standardization. Input SMILES are sanitized with RDKit, valence-checked, reduced to a deterministic parent after salt and fragment removal, and assigned deterministic charge, aromaticity, tautomer, and stereochemistry conventions. Canonical isomeric SMILES and molecule IDs are generated before compatible duplicate measurements are aggregated. Invalid structures retain an explicit audit reason. A molecule without a valid Bemis–Murcko scaffold is excluded from the scaffold-disjoint analysis. Canonical molecular graphs are cached after standardization; cache keys include the molecule ID and featurizer and configuration hashes.

The outer split is deterministic and scaffold-disjoint, with target Train/Calibration/Validation/Test proportions $60/15/10/15$. Realized counts, scaffold hashes, group sizes, and the split seed are stored in the run manifest.

Structure-only retrieval and eligibility. Candidate edges are constructed independently inside each locked split before training. For a source molecule i, the source itself is excluded. The experimental configuration retrieves and ranks molecules by split-local ECFP4 Tanimoto similarity (Rogers & Hahn, 2010). Molecular validity, the configured chemical-edit constraints, endpoint and label-scale compatibility, and availability of an observed label determine whether a retrieved pair is eligible. These checks do not alter the structure-only ranking, and label values, predictions, harms, out-of-fold harms, and risk scores are never used for retrieval or ranking.

Eligible candidates are ordered by structural similarity, with canonical molecule ID as the deterministic tie-break, and the configured top-K neighbors are retained. The resulting graph $\mathcal{G}_s$ and candidate sets $\mathcal{C}_s(i)$ are frozen before training. Sources with no eligible neighbor do not receive endpoint supervision. During R2P training, the separate query budget $Q = 10$ limits how many members of the frozen set are evaluated: neighborhoods of size at most Q are fully enumerated, while larger neighborhoods follow the recorded deterministic query order. Thus the construction top-K controls graph degree, whereas Q controls endpoint search and does not set the construction value of K.

The graph audit records candidate availability, degree and component distributions, scaffold coverage, deterministic ordering, and source, configuration, data, and graph hashes. Each retained edge stores its structural score and eligibility metadata.

A.2 Backbone, task heads, and risk initialization

The molecular backbone has depth three, hidden width 300, and dropout 0.1. For binary classification, the task head returns $p_{i,t}$ and uses binary cross-entropy over observed labels. For regression, it returns $\mu_{i,t}$ and $\nu_{i,t} = \log \sigma^2_{i,t}$ and uses the Gaussian negative log-likelihood in Equation 3. Clean task losses are averaged over observed molecule–task pairs.

For task t, the prediction-aware risk head receives the molecular representation z_i, prediction features $\xi_{i,t}$, and learned task embedding $\mathbf{v}_t$. The classification features are probability, entropy, and margin; the regression features are mean and log variance. Nonnegative task risks $\rho_{i,t}$ are averaged over $t \in \mathcal{T}_i$ to obtain r_i, as in Equation 4. In the reported configuration, the frozen candidate graph

is used only for endpoint search; it is not used for backbone message passing or as an additional risk-head input.

Train scaffolds are divided into three balanced folds. For $i \in \mathcal{F}_k$, a shadow predictor trained without any molecule from $\mathcal{F}_k$ produces a held-out prediction. Classification Brier harm is converted to a Train-only, task-aware percentile. For regression, define

$$e_{i,t}^{\mathrm{OOF}} = \frac{|y_{i,t} - \mu_{i,t}^{\mathrm{OOF}}|}{\max(m_t^{\mathrm{OOF}}, \epsilon)}, \qquad m_t^{\mathrm{OOF}} = \mathrm{median}_{j:\, t \in \mathcal{T}_j} |y_{j,t} - \mu_{j,t}^{\mathrm{OOF}}|. \tag{18}$$

All out-of-fold targets, normalizers, and rankings are detached before risk-head training.

Let $\widetilde{h}_i^{\mathrm{OOF}}$ be the normalized observed-task mean harm. The prediction-aware risk head is initialized with

$$\mathcal{L}_{\mathrm{base}} = \frac{1}{|\mathcal{D}_{\mathrm{tr}}|} \sum_{i \in \mathcal{D}_{\mathrm{tr}}} \mathrm{SmoothL1}(r_i, \widetilde{h}_i^{\mathrm{OOF}}) + \lambda_{\mathrm{rank}} \mathcal{L}_{\mathrm{pair}}, \qquad \mathcal{L}_{\mathrm{pair}} = \frac{1}{|\mathcal{P}|} \sum_{(i,j) \in \mathcal{P}} [\delta + r_i - r_j]_+, \tag{19}$$

where (i, j) is ordered from lower to higher out-of-fold harm. The margin, pair sampling, and task balancing are fixed in the manifest. The shared starting point is the R2P(Base) initialization. It stores predictor and risk states, the task specification, the out-of-fold normalizer, split and graph hashes, and all configuration hashes.

A.3 Centered acceptance and endpoint losses

At training step u, define centered risk and the detached scale

$$\bar{r}_u = \mathrm{sg}\left(\frac{1}{N_{\mathrm{tr}}} \sum_{i \in \mathcal{D}_{\mathrm{tr}}} r_i^{(u)} \right), \quad \widetilde{r}_i^{(u)} = r_i^{(u)} - \bar{r}_u, \quad s_u = \mathrm{sg}\left(\mathrm{MAD}_{i \in \mathcal{D}_{\mathrm{tr}}}(\widetilde{r}_i^{(u)}) \right). \tag{20}$$

With $\tau_u = \mathrm{sg}(\mathrm{Quantile}_c(\{\widetilde{r}_i^{(u)} : i \in \mathcal{D}_{\mathrm{tr}}\}))$, $c = 0.80$, and $T_a = 1$, training uses

$$a_i^{\mathrm{soft},(u)} = \sigma\left(\frac{\tau_u - \widetilde{r}_i^{(u)}}{T_a \max(s_u, \epsilon)} \right), \qquad \mathcal{L}_{\mathrm{cov}} = \left(\frac{1}{N_{\mathrm{tr}}} \sum_{i \in \mathcal{D}_{\mathrm{tr}}} a_i^{\mathrm{soft},(u)} - c \right)^2. \tag{21}$$

For each trained model checkpoint, the deployment threshold is fitted from its Calibration risks:

$$\tau_{\mathrm{cal}} = r^{\mathrm{cal}}_{(\lceil c N_{\mathrm{cal}} \rceil)}, \qquad a_i = \mathbb{I}[r_i \le \tau_{\mathrm{cal}}]. \tag{22}$$

Each variant and seed receives a threshold from its own Calibration risk scores. The order statistic depends on score ranks, and ties can increase realized coverage. Validation evaluates checkpoints with these thresholds and selects the checkpoint for the final Test replay. Endpoint audits use the evaluation labels after the model, graph, and threshold have been frozen.

At each dynamic step, compute task-native detached endpoint fitness over the frozen graph. Let e_j denote bounded Brier harm for classification or normalized absolute error for regression. The search scores are

$$\begin{aligned} F_{\mathrm{UA}}^{\mathrm{search}}(i \to j) &= a_j^{\mathrm{soft}} \begin{cases} e_j, & \text{classification,} \\ \log(1 + e_j), & \text{regression,} \end{cases} \\ F_{\mathrm{OR}}^{\mathrm{search}}(i \to j) &= (1 - a_j^{\mathrm{soft}}) \begin{cases} 1 - e_j, & \text{classification,} \\ \exp(-e_j), & \text{regression.} \end{cases} \end{aligned} \tag{23}$$

The endpoint is the canonical-ID tie-broken argmax over the queried set $\mathcal{C}_s^Q(i)$. This set equals $\mathcal{C}_s(i)$ when its size is at most Q.

For the outer loss, endpoint harm is recomputed after identities are frozen. The scores F_q below retain gradients through soft acceptance. For UA, $h_j^{\mathrm{Brier,cur}}$ is current molecular Brier harm and $\widetilde{e}_j = |y_j - \mu_j| / \max(m_{\mathrm{OOF}}, \epsilon)$ is current regression error, where m_{OOF} is the median absolute

Algorithm 1 R2P training with discrete endpoint search

Require: Train data, fixed graph $\mathcal{G}_{\mathrm{tr}}$, R2P(Base) parameters θ, ψ,
OOF harm bank, search-harm evaluator, coverage c, budget Q
Ensure: Trained parameters θ, ψ

$\mathcal{S} \leftarrow \{i \in \mathcal{D}_{\mathrm{tr}} : |\mathcal{C}_{\mathrm{tr}}(i)| > 0\}$
$\mathcal{K} \leftarrow \{\mathrm{clean}, \mathrm{base}, \mathrm{cov}, \mathrm{UA}, \mathrm{OR}\}$
for each training step u **do**
  Compute Train predictions and risks r_i ▷ Equation 4
  Detach $\bar{r}_u, \tau_u, s_u$; compute a_i^{soft} ▷ Equations 9–10
  Evaluate and detach search harms e_i
  for each $i \in \mathcal{S}$ and $q \in \{\mathrm{UA}, \mathrm{OR}\}$ **do**
    Evaluate $F_q^{\mathrm{search}}(i \to j)$ on $\mathcal{C}_{\mathrm{tr}}^Q(i)$ with detached inputs
    $j_q^*(i) \leftarrow \arg\max_{j \in \mathcal{C}_{\mathrm{tr}}^Q(i)} F_q^{\mathrm{search}}(i \to j)$
    Recompute $F_q(i \to j_q^*(i))$ with gradients, holding $j_q^*(i)$ fixed
    $b_{i,q} \leftarrow \mathrm{sg}(F_q(i \to i))$
  end for
  $\mathcal{L}_{\mathrm{UA}}, \mathcal{L}_{\mathrm{OR}} \leftarrow 0, 0$
  if $\mathcal{S} \neq \emptyset$ **then**
    $\mathcal{L}_q \leftarrow |\mathcal{S}|^{-1} \sum_{i \in \mathcal{S}} [F_q(i \to j_q^*(i)) - b_{i,q}]_+^2$ for each q
  end if
  Compute $\mathcal{L}_{\mathrm{clean}}, \mathcal{L}_{\mathrm{base}}, \mathcal{L}_{\mathrm{cov}}$ ▷ Equations 3, 6, 10
  $d_k \leftarrow \nabla_{\theta,\psi}(\lambda_k \mathcal{L}_k)$ for active $k \in \mathcal{K}$
  $d \leftarrow \mathrm{PCGrad}_{\text{equal-norm}}(\{d_k\})$
  $(\theta, \psi) \leftarrow \mathrm{OptimizerStep}((\theta, \psi), d)$
  if OOF refresh is due **then**
    Refresh scaffold OOF predictions; detach harm targets and normalizers
  end if
end for
return θ, ψ

error in the current OOF bank. For OR, the harm reference comes from detached OOF predictions:

$$F_{\mathrm{UA}}(i \to j) = a_j^{\mathrm{soft}} \begin{cases} h_j^{\mathrm{Brier,cur}}, & \text{classification,} \\ \log(1 + \widetilde{e}_j), & \text{regression,} \end{cases}$$
$$F_{\mathrm{OR}}(i \to j) = (1 - a_j^{\mathrm{soft}}) \begin{cases} 1 - \mathrm{sg}(h_j^{\mathrm{Brier,OOF}}), & \text{classification,} \\ \exp(-\mathrm{sg}(e_j^{\mathrm{OOF}})), & \text{regression.} \end{cases} \tag{24}$$

Here $h_j^{\mathrm{Brier,OOF}}$ is bounded molecular OOF Brier harm. The UA harm term retains its predictor gradient. The OR harm term is detached.

Let $\mathcal{S}$ contain Train sources with valid candidates. After selecting endpoints, freeze their identities and use the relative loss

$$\mathcal{L}_q = \frac{1}{|\mathcal{S}|} \sum_{i \in \mathcal{S}} \left[F_q(i \to j_q^*(i)) - \mathrm{sg}(F_q(i \to i)) \right]_+^2 . \tag{25}$$

If $\mathcal{S} = \emptyset$, $\mathcal{L}_q = 0$. Candidate availability and query counts are recorded separately.

A.4 Analysis of the decision objectives

Fixed-coverage selective risk. The classification scores have an exact relation to selective risk when prediction harms and acceptance mass are fixed. Let $h_i \in [0, 1]$ be the fixed harm of molecule i, let $a_i \in [0, 1]$, and let $m = \sum_{i=1}^n a_i > 0$. For soft decisions, define selective harm as $\sum_i a_i h_i / m$.

Expanding the two score sums gives

$$\sum_{i=1}^{n}\big[a_i h_i + (1-a_i)(1-h_i)\big] = \sum_{i=1}^{n}\big[2a_i h_i + 1 - h_i - a_i\big]$$
$$= 2m\ \mathrm{SelRisk} + \sum_{i=1}^{n}(1-h_i) - m. \tag{26}$$

The final two terms are constant over acceptance rules with the same mass. Thus the summed UA and OR scores and selective risk have the same minimizers. For hard decisions, replacing an accepted molecule i by a deferred molecule j changes selective risk by

$$\Delta\,\mathrm{SelRisk} = \frac{h_j - h_i}{m}. \tag{27}$$

The replacement improves risk whenever $h_j < h_i$.

Decision gradients. The acceptance derivative determines how each branch changes risk scores. Within one outer update, hold the endpoint index, risk center, quantile, and scale fixed. Write $a = a_j^{\mathrm{soft}}$ and $\kappa = T_a \max(s_u, \epsilon) > 0$. Equation 10 gives

$$\frac{\partial a}{\partial r_j} = -\frac{a(1-a)}{\kappa}. \tag{28}$$

For classification, let h be current Brier harm and h_0 the detached OOF reference. Treating harm and risk as separate inputs to the score, the chain rule gives

$$\frac{\partial F_{\mathrm{UA}}}{\partial r_j} = -\frac{h\,a(1-a)}{\kappa}, \qquad \frac{\partial F_{\mathrm{UA}}}{\partial h} = a,$$
$$\frac{\partial F_{\mathrm{OR}}}{\partial r_j} = \frac{(1-h_0)a(1-a)}{\kappa}. \tag{29}$$

UA supplies a direction that increases risk and lowers prediction harm. OR supplies a direction that lowers risk for endpoints with low reference harm. These are the two local corrections to the acceptance boundary.

Detaching the OR reference removes an error-increasing gradient path. For a differentiable current harm h, the derivative of $(1-a)(1-h)$ with respect to h is $-(1-a)$. Holding the harm reference fixed leaves the acceptance path as the branch's training signal. That path updates both ψ and θ through Equation 4.

For regression, let e be current absolute error divided by the fixed OOF normalizer and let e_0 be the detached normalized OOF error. The corresponding derivatives are

$$\frac{\partial F_{\mathrm{UA}}}{\partial r_j} = -\frac{\log(1+e)a(1-a)}{\kappa}, \qquad \frac{\partial F_{\mathrm{UA}}}{\partial e} = \frac{a}{1+e},$$
$$\frac{\partial F_{\mathrm{OR}}}{\partial r_j} = \frac{\exp(-e_0)a(1-a)}{\kappa}. \tag{30}$$

The same signs hold for both task types. The regression transforms damp the effect of large errors in UA and concentrate OR supervision on low-error endpoints.

Relative endpoint penalty. The squared hinge implements a local comparison with the source. Define $b_{i,q} = \mathrm{sg}(F_q(i \to i))$ for the current update and let

$$\ell_{i,q} = \big[F_q(i \to j_q^*(i)) - b_{i,q}\big]_+^2. \tag{31}$$

This penalty is zero exactly when the selected endpoint score is at most the source reference. Differentiating with the index and reference fixed gives

$$\nabla_{\theta,\psi}\ell_{i,q} = 2\left[F_q(i \to j_q^*(i)) - b_{i,q}\right]_+ \nabla_{\theta,\psi} F_q(i \to j_q^*(i)). \tag{32}$$

The positive excess scales the endpoint gradient. Averaging these terms over $\mathcal{S}$ yields the gradient of $\mathcal{L}_q$. The detached reference is recomputed at the next search step; source predictions can change as the shared model parameters are updated.

Table 4: Objectives and parameter updates for the matched component study. Coverage regularization is active in all dynamic variants.

Method	Clean	Risk supervision	UA	OR	Predictor
R2P(Base)	frozen	frozen	0	0	frozen
R2P(UA)	1	1	1	0	trainable
R2P(OR)	1	1	0	1	trainable
R2P(UA+OR)	1	1	1	1	trainable

Scope of the analysis. Equation 26 assumes fixed classification harms, fixed acceptance mass, and global score sums. R2P uses local endpoint supervision, updates prediction harms, and penalizes soft coverage deviations. Its regression scores use nonlinear transforms of absolute error, while the main regression metric is RMSE. The derivative results describe individual score paths and the relative penalty at a fixed training state. The inner search maximizes F_q^{search} on the queried set, and the outer update combines gradients of the surrogate losses using PCGrad. The empirical evaluation measures the resulting changes in selective risk.

A.5 Matched variants and checkpoint selection

Table 4 specifies the objectives for the configurations in the component study. All dynamic variants use Equation 16 and begin from the same R2P(Base) initialization.

The configurations share splits, candidate graphs, target coverage, query budget, and seeds. R2P(Base) receives no further training. The dynamic variants refresh the Train-only scaffold OOF bank every 10 epochs for classification and every 10 optimizer steps for regression. Refresh state, shadow-model seeds, and training budgets are recorded in the run manifest. The runs underlying the current experimental comparison select checkpoints on Validation. Acceptance thresholds are fitted on Calibration before Validation evaluation; the selected checkpoint and threshold are frozen for Test. Each endpoint query record contains the source, candidate, split, direction, query index, fitness, and graph and model-state hashes.

A.6 Evaluation and reproducibility

The main comparisons report Selective Brier for classification and Selective RMSE for regression at fixed 80% coverage. Classification first averages Brier harm over observed tasks within each molecule, then averages over accepted molecules. Regression RMSE is computed directly from accepted squared errors. Appendix B defines both metrics. Endpoint audit records retain candidate availability, query counts, endpoint reuse, and whether search was exact or budget-limited. Results are means and sample standard deviations over seeds 0, 1, and 2. The fixed-coverage comparison accepts the lowest-risk $\lceil cN_{\mathrm{te}} \rceil$ molecules within the Test split. The deployment policy instead applies the frozen Calibration threshold, so the two procedures can yield different accepted sets. For every benchmark, the manifest records the dataset name, standardized molecule count, task count or class count, and realized split sizes. Dataset summaries use these recorded counts.

Every frozen run stores standardized molecules and node IDs, split/scaffold assignments, candidate edges and structure attributes, task specification and masks, predictor/risk/OOF states, endpoint maps and query ledger, Calibration policy and threshold, checkpoint selector and epoch, code/dependency/seed metadata, and manifest/graph/artifact/output hashes. The run manifest records the actual GPU model and count, software versions, and wall-clock budget.

The reported runs use the repository-locked Python, PyTorch, CUDA, and RDKit dependencies. Resource records store the accelerator, software versions, seed, and runtime for each run. The implementation records identify the backbone adapter, task-aware harm and normalization, prediction-aware risk, and finite endpoint selection components used to produce each manifest.

Table 5: Default model and training configuration.

Component	Parameter	Value
Outer split	grouping / proportions	Bemis–Murcko / 60:15:10:15
OOF	grouped folds	3 scaffold folds
Backbone	depth / hidden / dropout	3 / 300 / 0.1
Candidate graph	retrieval / freeze	ECFP4 top-K / before training
Endpoint search	query budget Q	10 per source
Acceptance	coverage c / temperature T_a	0.80 / 1.0
Numerical stability	ϵ	10^{-6}
Risk head	hidden width / aggregation	64 / task-aware mean
Warm start	optimizer / learning rate	Adam / 10^{-3}
Dynamic update	optimizer / learning rate	Adam / 10^{-4}
Training	batch size / dynamic epochs	256 / 100
Training	warm-up / shadow-OOF epochs	20 / 30
OOF refresh	classification / regression	10 epochs / 10 optimizer steps
Loss weights	clean/base/cov/UA/OR	1/1/1/1/1
Regularization	weight decay / LR schedule	0 / none
Gradient merger	method	equal-norm PCGrad

Table 6: Benchmark statistics after molecular standardization. N and N_{te} denote the total and scaffold-disjoint Test-set sizes.

Dataset	**Tasks**	$\boldsymbol{N}$	$\boldsymbol{N_{\mathrm{te}}}$
Classification			
BACE	1	1,513	227
BBBP	1	1,965	295
ClinTox	2	1,441	216
SIDER	27	1,427	214
Tox21	12	7,831	1,175
ToxCast	617	8,577	1,287
Regression			
ESOL (Delaney, 2004)	1	1,128	169
FreeSolv (Mobley & Guthrie, 2014)	1	642	97
Lipophilicity	1	4,200	630

B Additional Experimental Details

B.1 Benchmarks and data splits

Table 6 reports the number of standardized molecules, tasks, and Test molecules in each benchmark. Whole Bemis–Murcko scaffold groups are assigned to mutually disjoint Train, Calibration, Validation, and Test splits with target proportions of 60/15/10/15. Train is the only split used for parameter updates and endpoint supervision. Calibration fits acceptance thresholds or conformal scores, Validation selects checkpoints, and Test is accessed only for the final evaluation.

B.2 Implementation and comparison methods

All predictors use the shared three-layer molecular backbone described in Appendix A.2, with hidden width 300 and dropout 0.1. R2P uses a frozen split-local, structure-only candidate graph, query budget $Q = 10$, and target coverage $c = 0.80$. Its Train-only scaffold out-of-fold harm bank is refreshed every 10 epochs for classification and every 10 optimizer steps for regression. The selected model and its Calibration-fitted acceptance threshold are frozen before Test evaluation. Appendix A gives the remaining optimization settings.

Full acceptance accepts every Test molecule and provides the non-selective reference. *Random* uniformly selects 80% of Test molecules and tests whether rejecting an arbitrary subset is sufficient to reduce risk. *Embedding distance* ranks a molecule by the distance between its molecular representation z_i and the Train-set representation centroid. *Split conformal* uses a Calibration-fitted prediction-set size for classification and conformalized quantile regression interval width for regres-

sion. *Confidence* (maximum class probability, or MCP) ranks classification predictions by their largest class probability; its regression analogue is a standardized-residual proxy based on the predicted mean. *R2P* denotes the joint R2P(UA+OR) model. None of these selection scores uses Test labels.

B.3 METRICS AND REPORTING

Let $a_i \in \{0, 1\}$ indicate acceptance, $\mathcal{T}_i$ be the observed tasks for molecule i, and $p_{i,t}$ and $y_{i,t}$ be its predicted probability and binary label. The reported selective metrics are

$$b_i = \frac{1}{|\mathcal{T}_i|} \sum_{t \in \mathcal{T}_i} (p_{i,t} - y_{i,t})^2, \quad \text{SelBrier} = \frac{\sum_i a_i b_i}{\sum_i a_i}, \quad \text{SelRMSE} = \left(\frac{\sum_i a_i (\mu_i - y_i)^2}{\sum_i a_i} \right)^{1/2}. \tag{33}$$

Selective Brier averages over observed tasks within each molecule. All fixed-coverage selectors accept exactly $\lceil cN_{\text{te}} \rceil$ molecules with $c = 0.80$; Full accepts all molecules. We report the mean and sample standard deviation over seeds 0, 1, and 2. Random is additionally averaged over 256 deterministic rankings per seed. Boldface marks the lowest selective-risk mean among variants evaluated at 80% coverage, including ties at the displayed precision.

B.4 SAFETY-RELEVANT CASE AUDIT

The case study in Section 4.5 is a post-hoc audit of the frozen ToxCast seed-1 Test replay. Model parameters, the candidate graph, risk scores, and the 80% acceptance boundary were fixed before inspecting the Test labels. The two exact-structure cases were then selected from the held-out replay for their safety relevance; Test labels are used only to compute the molecule-level Brier values reported in Table 3. They do not enter training, threshold fitting, checkpoint selection, or risk ranking. ToxCast has $N_{\text{te}} = 1{,}287$, so $\lceil 0.8 N_{\text{te}} \rceil = 1{,}030$ molecules are accepted. The reported assay counts (452 for Ketoconazole and 249 for Tolcapone) are the observed, non-missing assays used in each molecule-level average. The clinical context is used only to motivate the retrospective audit and is not an additional model input or evaluation label.