# MAKING EVERY EXPERIMENT COUNT: ANALOGICAL MOLECULAR PROPERTY PREDICTION

**Tianang Leng***
Department of Bioengineering
University of Pennsylvania
Philadelphia, PA 19104, USA
ltianang@engineering.upenn.edu

**Joseph Lee*** **& Haydn Jones**
Department of Computer and
Information Science
University of Pennsylvania
Philadelphia, PA 19104, USA
{jojolee,haydnj}@seas.upenn.edu

**Cesar de la Fuente**
Departments of Psychiatry
and Microbiology
Perelman School of Medicine
University of Pennsylvania
Philadelphia, PA 19104, USA
cfuente@upenn.edu

**Jacob Gardner** **& Mark Yatskar**
Department of Computer and
Information Science
University of Pennsylvania
Philadelphia, PA 19104, USA
jacobrg@seas.upenn.edu
myatskar@cis.upenn.edu

## ABSTRACT

Evidence for therapeutically relevant molecular properties is diverse, spanning cell-based experiments, animal studies, and human trials. How can machine learning models utilize such indirect evidence—measurements on related molecules, often from different assay contexts—when predicting a property of a molecule that has never been tested? We introduce an analogical-reasoning approach that, rather than predicting solely from structure, aggregates such evidence from the literature and reasons about its relevance. The core component is a property-transfer model, trained on literature evidence, that scores how likely a measurement on one molecule is to carry over to another. A context optimization procedure then selects evidence maximizing diversity, relevance, and transfer likelihood, which frontier LLMs can use to predict by analogy. We assemble a dataset of direct and indirect safety and ADME evidence mined from a 25M-paper corpus and use the direct measurements to build literature-derived benchmarks for six tasks. With Gemini 3.8 Flash, our system reaches a mean macro-F1 of 0.736 on the literature set and 0.815 on the *de facto* TDC benchmark, exceeding the best ML baseline on each benchmark—including MiniMol trained multi-task on the same indirect evidence—by 11.9 and 11.2 percentage points, respectively. Against naive retrieval with the same backbone—the comparison that isolates our evidence selection—the gains are 2.4 and 4.5 points with Gemini, and 3.4 and 8.8 points averaged across the three backbones, the smaller Gemini margin reflecting its stronger naive baseline. Gains over the ML baselines hold for all three models. Adding indirect evidence improves mean performance in every backbone–benchmark combination, while supplying the same evidence to baselines as auxiliary multi-task targets lowers mean performance on both benchmarks. Ablations support the contributions of learned transfer scoring, informativeness, and diversity to evidence selection.

## 1 INTRODUCTION

Around 90% of drug candidates entering clinical trials fail, and roughly half of these failures are attributable to poor safety or ADME profiles (absorption, distribution, metabolism, and excretion) (Sun et al., 2022) that are challenging to predict. This is not entirely due to a lack of data, even if that data is rarely gathered in central repositories. The difficulty is that the evidence bearing on a property like oral bioavailability is multifaceted and mostly indirect. Our belief is shaped not just by direct

*Equal contribution

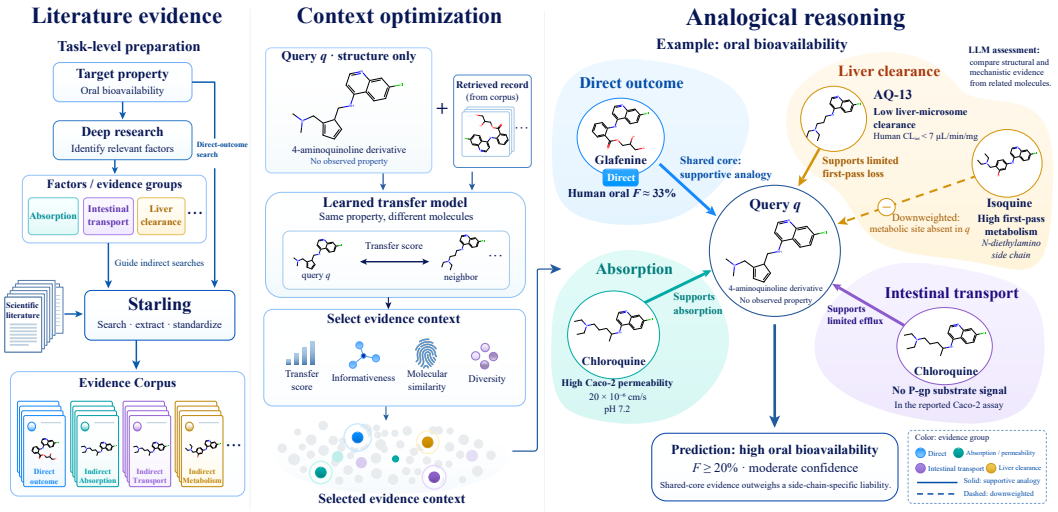


Figure 1: **Overview of the analogical reasoning system. Left:** Deep research identifies task-relevant factors that guide Starling's searches for indirect evidence. These records, together with direct measurements of the target property, form the evidence corpus. **Middle:** Context optimization balances learned transfer scores, endpoint informativeness, molecular similarity, and diversity to select evidence for the query; this stage is shown schematically. **Right:** The language model assesses which observations transfer and integrates their implications for the target property. In this recorded oral bioavailability example, shared-core evidence supports the prediction, while a metabolic liability associated with a different side chain is downweighted.

measurements, but also by related experiments, such as permeability, metabolic stability, and animal pharmacokinetics. Each measurement captures only part of a property and may be noisy or hard to weight, especially when run on related molecules instead of the one of interest. Evidence accumulates across protein-interaction assays, cell studies, and animal experiments before human trials, which are ethical only after preclinical safety support. But this evidence can conflict: a molecule may be orally toxic in rats while a Caco-2 assay suggests poor human absorption. How likely is human toxicity, and how much should each result count? For novel molecules, the challenge grows because the most relevant data often comes indirectly from related compounds tested in different settings. Integrating this heterogeneous, indirect evidence is a central machine-learning challenge in drug discovery.

Existing models for these critical properties largely ignore this indirect evidence. Most models compute features from molecular structure and are trained on small datasets of *direct* outcomes. For example, models for oral bioavailability are generally trained only on direct oral bioavailability measurements, often exclusively from human data (Huang et al., 2021; Kläser et al., 2024). As a result, most experimental biomedical evidence goes unused.

In this paper, we seek to address this gap and leverage indirect assay measurements to improve predictive performance on high-value ADME and safety properties. We hypothesize that generalist models like LLMs with broad background knowledge—knowledge of what a Caco-2 assay measures, or of when rat toxicity extrapolates to humans—can weigh indirect evidence in ways that are difficult to capture through multi-task supervised learning on molecular structures alone. We assemble a corpus of millions of indirect evidence records from the literature, retrieve from it for each query molecule, and perform in-context analogical reasoning over the selected records to make predictions. Rather than predicting from molecular structure alone, our approach connects evidence from structurally and experimentally related molecules and reasons about whether their observed properties should transfer and how they would steer the final prediction of a query molecule.

Our system, Ampere (**A**nalogical **M**olecular **P**roperty **E**stimation from **R**elated **E**vidence), Figure 1, has three components: a deep research pipeline that gathers evidence from the literature, a context optimizer that selects relevant evidence, and a language model that weighs it. The pipeline extends Starling, an agentic system that queries 25M PubMed papers to build evidence databases (Jones et al., 2026). Given a target property (e.g., oral bioavailability—the fraction of an active ingredient reaching the bloodstream after ingestion), an initial query identifies *factors*: measurement types in

the literature that may correlate with the target. Starling builds a table for each factor, producing a pool of indirect evidence. Because only a small subset is useful per query, we score candidates by molecular similarity, measurement informativeness, diversity, and a learned property-transfer model that predicts whether two molecules will have similar measurements. We combine these scores into a joint submodular objective and maximize it for each query. The language model then reasons over the selected evidence, upweighting or discounting it based on retrieved-analog quality (Figure 2); this reasoning emerges when the prompts and context encourage it.

We evaluate on six ADME/safety tasks from the Therapeutic Data Commons (TDC) (Huang et al., 2021) against three baseline families: fingerprint-based classical methods; RASAR Data Fusion (random forests over fixed indirect-data features) (Luechtefeld et al., 2018); and MiniMol, a physics-pretrained neural network model, evaluated both in its standard form and with multi-task indirect-data training (Kläser et al., 2024). We evaluate Ampere with three frontier language models: DeepSeek-V4-Flash, GPT-6 Luna, and Gemini 3.8 Flash. On TDC, the best of these, our Gemini-based system reaches 0.815 mean macro-F1, beating the strongest ML baseline by 11.2 points. Because TDC test sets are small, we also use a broader literature-based benchmark (Jones et al., 2026), where we score 0.736 and lead the strongest ML baseline by 11.9 points; gains over ML predictors also hold with DeepSeek-V4-Flash and GPT-6 Luna. Adding indirect evidence improves all three reasoning models on both benchmarks, and ablations confirm the value of transfer scoring, informativeness, and diversity. Ampere's evidence-integration method improves mean macro-F1 over naive similarity-based evidence retrieval by 3.4 on the literature-derived benchmarks and 8.8 points on TDC, on average across backbones, showing that evidence selection adds value beyond the backbone.

## 2 METHODOLOGY

Our goal is to use diverse and indirect experimental evidence about molecules to improve property prediction for entirely novel molecules. Given a query molecule $x$, we aim to predict $y$, a binary label of whether a target property holds for $x$, for example whether its oral bioavailability exceeds 20%. We will assume access to a corpus of evidence $\mathcal{E}$. The corpus contains, among other things, many descriptions of molecules and their properties. For a given target property $y$, the properties described in the corpus are either *direct evidence*, *indirect evidence*, or irrelevant.

*Direct evidence* reports measurements of the target property that could easily be used for direct supervised learning to predict $y$. *Indirect evidence* is all other evidence that describes related outcomes but not the property itself. For example, the corpus might describe the liver metabolism speed of some molecules, which negatively correlates with oral bioavailability. To use the direct and indirect evidence that exists in the corpus to make predictions, we need to do two things. First, we need to determine what kinds of experiments have been done in the literature that result in direct or indirect evidence. Then, for a given query molecule $x$, we need to retrieve the measurements from the literature that will be most helpful for predicting $y$.

Figure 1, described in detail below, summarizes how we construct and use evidence corpora. We first identify what kinds of evidence might be useful for predicting each property of interest through deep research. We use the factors identified to guide Starling's literature searches, collecting both direct measurements and the indirect measurements identified for the property of interest (Section 2.1). To apply these observations to a query molecule, a learned property-transfer model estimates the transferability of each measurement (Section 2.2). These estimates guide context optimization, which selects a complementary evidence set $E_{x,y} \subseteq \mathcal{E}$ to represent the task-relevant property landscape around the query (Section 2.3). Given $E_{x,y}$, a language model evaluates which observations transfer and weighs their implications to predict the target property, as seen in Figure 2.

### 2.1 EVIDENCE GATHERING

We need to gather evidence from the literature that is useful for predicting the property of interest, $y$. We begin with a deep research stage where we use OpenAI's deep research to propose a list of biological properties (which we will call factors) that (1) are commonly measured in the scientific literature and (2) either correlate with or directly measure the target property $y$. We provide the final set of factor definitions, which we manually curate on a per-task basis, in Appendix A.2.

We gather measurements from the literature for each factor identified. To do this, we submit them as Starling queries to generate tabular data from its 25M indexed PubMed papers (Jones et al., 2026). We

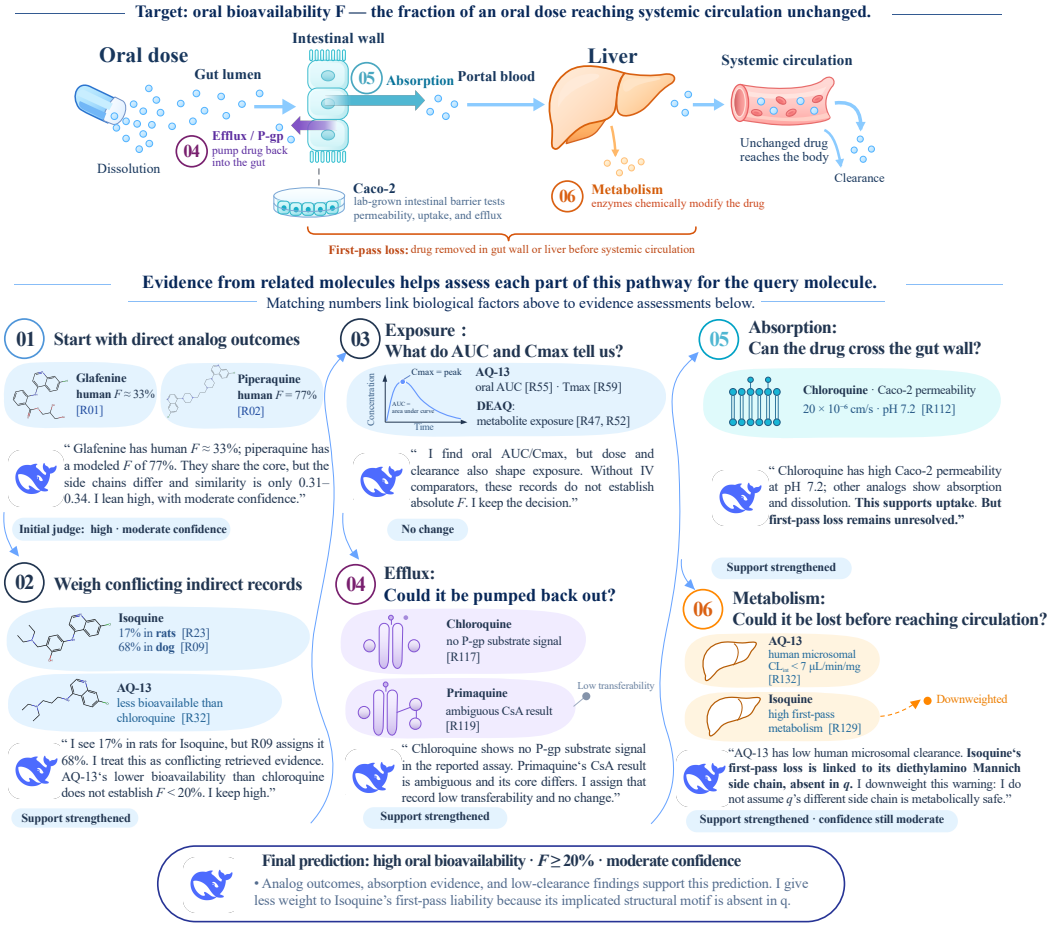


Figure 2: **An example of analogical reasoning for oral bioavailability. Top**: The biological pathway underlying oral bioavailability and the factors used to organize indirect evidence. **Bottom**: An inference trace for a query molecule. DeepSeek-V4-Flash compares direct analog outcomes and retrieved evidence on exposure, absorption, efflux, and metabolism, weighs their relevance and transferability, and combines them into a final prediction ($F \geq 20\%$). Numbered panels connect each reasoning step to the corresponding factor above; record IDs denote retrieved literature evidence.

execute $G$ total Starling queries for each factor $g$. Each query retrieves *records* $\{r_{g,i}\}_{i=1}^{n_g}$. A record contains a single measurement of the factor $g$ and the conditions under which that measurement was made. In our running example, a record for liver metabolism speed might measure the microsomal clearance rate, reported in $\mu$L/min/mg, of some molecule in mice. Note that at this stage, we collect data from the literature agnostic to whether the specific molecules will be useful for predicting the target property of specific query molecules. Rather, at this stage, our goal is to maximize data coverage so that—given a query molecule—we can *later* retrieve from the tabulated records.

### 2.2 Property-Transfer Model

Because LLMs have limited context windows, retrievers are crucial for selecting task-relevant information (Lewis et al., 2020). Morgan fingerprints are a useful heuristic for retrieving similar molecules (Rogers & Hahn, 2010), but they are static and not tailored to target properties. Instead, we train a property-aware retriever on millions of literature claims and experimental observations.

**Problem Formulation.** An evidence record $r$ records some measurement about a molecule from the literature. Our goal is to train a model that predicts whether *query* molecule $x$ would have a similar measurement if measured under the same conditions as the record. Using our example above, if $r$ records the microsomal clearance rate of a molecule in mice, we want a model $f_\theta$ with parameters $\theta$ that predicts whether $x$ would have a *similar* clearance rate if also measured in mice.

We train a property-transfer model $f_\theta(r, x) \mapsto [0, 1]$ that maps a record $r$ and a query molecule $x$ to a score representing the predicted similarity of the experimental outcome if we were to replace the molecule in $r$ with the query molecule $x$. Intuitively, this model allows us to transfer assay knowledge from measurements made about molecules in the literature to our query molecules: it asks how well we can use the analogous molecule's microsomal clearance rate (for example) as an estimated measurement for the query molecule.

This model can be used to determine the relevance of records when we make predictions about the query molecule $x$. Given $x$, we rank records for each property by $s(x, r)$ and use these scores later in our optimization. To train $f_\theta$, we process data in the corpus $\mathcal{E}$ according to the following procedure.

**Grouping Records into Training Examples.** We construct training data by grouping Starling records that contain roughly the same kind of measurements, made under the same conditions, about different molecules. For example, if the microsomal clearance rate of a molecule $m_1$ and a molecule $m_2$ were both measured in humans and report $\mu$L/min/mg, then the corresponding records can form a training example for our transfer model where we will try to predict the similarity of the clearance rate values. Crucially, we must *normalize* the various conditions under which measurements are taken. For example, if $m_1$ has a measured rate in "humans" and $m_2$ has a measured rate in "patients", we must still consider this a valid training pair because "patients" are typically humans. We also remove outliers and other poor-quality evidence. See Appendix B for more details.

**Training Data and Loss.** Given a training pair of records $(r, r')$ with observed real-valued measurements $(y, y')$, we will treat one of the molecules as the query and the other as a candidate for transfer in training $f_\theta$. We construct a label from the difference $\sigma\big(\frac{|y-y'|}{s_e}\big)$ where $s_e$ is the standard deviation across *all* records with measurements directly comparable to $y$ and $y'$ (i.e., made under the same conditions). $\sigma$ is a sigmoid function centered at $0.4$ with temperature $0.1$ (Appendix B). The output of the sigmoid is used as a target in a cross-entropy loss.

**Training.** We fine-tune a language model (Intern-S1-mini; Bai et al., 2025) on pairs sampled within created endpoints, limiting the total number that can be sampled per endpoint. We train separate retrievers for each task and for direct and indirect data (see Appendix B for details).

### 2.3 Context Optimization

The corpus of evidence is often far larger than the effective context window of a model, so for each query molecule we must select a subset of records to reason over. Here, we select from the direct ($\mathcal{E}_{\text{dir}}$) and indirect ($\mathcal{E}_{\text{ind}}$) evidence records separately, writing $\mathcal{E}_\bullet \in \{\mathcal{E}_{\text{dir}}, \mathcal{E}_{\text{ind}}\}$ to mean one of the two pools of evidence arbitrarily. We denote by $B$ the number of records we seek to select from $\mathcal{E}_\bullet$ to aid in-context learning. $E_{x,y}$ as described in Section 2 will then be the union of the two selections.

What direct and indirect records should we pick? Most straightforwardly, each record should be *relevant*, reporting a measurement that transfers well to the query molecule $x$, and *informative*: if that measurement *were* known about $x$, it would be useful for predicting the *target* property $y$. Moreover, a record should communicate something new with respect to the others that we pick, providing sufficient *coverage* of the relevant and informative records. Relevance and informativeness are properties of individual records, while coverage is a global property of the chosen set.

We encode these three criteria as a single set utility function, $F(S; x, y)$, defined over the remainder of this section from per-record and set-level utilities as stated in Equation (5). The selection of our subset is then a budgeted optimization of this utility.

$$S^\star \in \underset{S \subseteq E_\bullet,\ |S|=B}{\arg\max} F(S; x, y). \tag{1}$$

**Relevance.** Let $\sigma_i = \text{sim}(m_i, x) \in [0, 1]$ be the Morgan fingerprint Tanimoto similarity between molecule $m_i$ of record $r_i$ and the query $x$, and let $s_i = f_\theta(r_i, x) \in [0, 1]$ be the transfer score of Section 2.2, which estimates whether $x$ would yield a similar outcome to $m_i$ under $r_i$'s experiment and conditions. We combine relevance and structural similarity as

$$G_i = \frac{(1 + \lambda_{\text{transfer}} s_i)\, \sigma_i}{1 + \lambda_{\text{transfer}}}, \tag{2}$$

where $\lambda_{\text{transfer}}$ sets how much of the score the learned model accounts for relative to structural similarity alone. We multiply $s_i$ by $\sigma_i$ to avoid selections where the structural similarity is low but the transfer score is high, as the downstream model prefers structurally similar molecules.

**Informativeness.** Endpoints will differ in how informative they are for the target property $y$: a record with an intestinal permeability result will bear more directly on oral bioavailability than a record with a cytotoxicity readout. We estimate this informativeness using the prior of an LLM, which scores the informativeness of record $r_i$'s endpoint as $w_i \in [0, 1]$. Because the corpus contains many more endpoints than can be scored directly, we first cluster them into semantic groups $\mathcal{U}$ and score the groups. Details of this procedure can be found in Appendix C.1.

**Coverage.** Two records can both be relevant and informative while adding no value when considered jointly. We model coverage along one structural and several categorical axes. For structure, we define $\mathcal{B}_i$ to be the set bits of record $r_i$'s Morgan fingerprint; structural coverage is then the fraction of the pool's bits that $S$ represents:

$$D_{\text{struct}}(S) = \frac{\left|\bigcup_{i \in S} \mathcal{B}_i\right|}{\left|\bigcup_{j \in \mathcal{E}_\bullet} \mathcal{B}_j\right|}. \tag{3}$$

Since each bit position is counted once, records contribute only via bits not already set in $S$.

The categorical axes are all of one form. For a family of categories $\mathcal{X}$, we define the utility as

$$D_{\mathcal{X}}(S) = \frac{1}{Z_{\mathcal{X}}} \sum_{\chi \in \mathcal{X}} \sqrt{|S_\chi|} \qquad Z_{\mathcal{X}} = \max_{S \subseteq \mathcal{E}_\bullet,\ |S|=B} \sum_{\chi \in \mathcal{X}} \sqrt{|S_\chi|}. \tag{4}$$

The square root gives each category diminishing credit for additional records, which encourages spread without imposing a strict quota, while $Z_{\mathcal{X}}$ is the largest sum attainable at the budget $B$ and serves to put $D_{\mathcal{X}}$ in $[0, 1]$. For the family of categories, we consider the evidence factors $\mathcal{G}$ in Section 2.1, the semantic groups $\mathcal{U}$, the standardized parent molecules $\mathcal{P}$, and, for the direct pool, the label categories $\mathcal{L}$. Because factors and semantic groups are defined only for indirect evidence, that pool uses $\{\mathcal{G}, \mathcal{U}, \mathcal{P}\}$ and direct uses $\{\mathcal{P}, \mathcal{L}\}$.

**Objective.** Putting this all together, we define the objective as

$$F(S; x, p) = \underbrace{\frac{1}{B} \sum_{i \in S} G_i}_{\text{relevance}} + \underbrace{\lambda_{\text{info}} \frac{1}{B} \sum_{i \in S} w_i}_{\text{informativeness}} + \underbrace{\lambda_{\text{struct}} D_{\text{struct}}(S) + \sum_{\mathcal{X} \in \mathcal{F}(E_\bullet)} \lambda_{\mathcal{X}} D_{\mathcal{X}}(S)}_{\text{coverage}}, \tag{5}$$

where $\mathcal{F}(E_\bullet)$ is the set of category families for the pool, and all weights $\lambda_{(\cdot)}$ are nonnegative.

**Optimization.** $F$ is a nonnegative weighted sum of modular per-record terms and monotone submodular coverage terms, and is thus itself monotone submodular. We maximize it by a lazy greedy selection, adding the record with the largest marginal gain until $|S| = B$, which attains a $(1 - \frac{1}{e})$ approximation of the optimum (Nemhauser et al., 1978; Minoux, 1978).

### 2.4 Analogical Reasoning

Given the optimized evidence context, a language model predicts the novel molecule's target property by analogy, comparing the query with retrieved molecules and weighing their structural differences and reported experimental conditions to judge whether each observation should transfer. Direct evidence informs the target property itself, whereas indirect evidence requires an additional argument linking the observed endpoint to the target. The model integrates arguments across molecules, weighs supporting and conflicting evidence, and outputs a prediction with a justification. As shown in Figure 2, the model distinguishes exposure measurements from absolute bioavailability and downweights a metabolic liability tied to a side chain absent from the query. We find that molecular context and a suitable prompt suffice to elicit this behavior across the model families we evaluated.

## 3 Experimental Setup

### 3.1 Benchmarks and Evaluation Protocol

We study six molecular property prediction tasks: blood–brain barrier penetration (BBB), oral bioavailability, skin sensitization, Ames mutagenicity, drug-induced liver injury (DILI), and car-

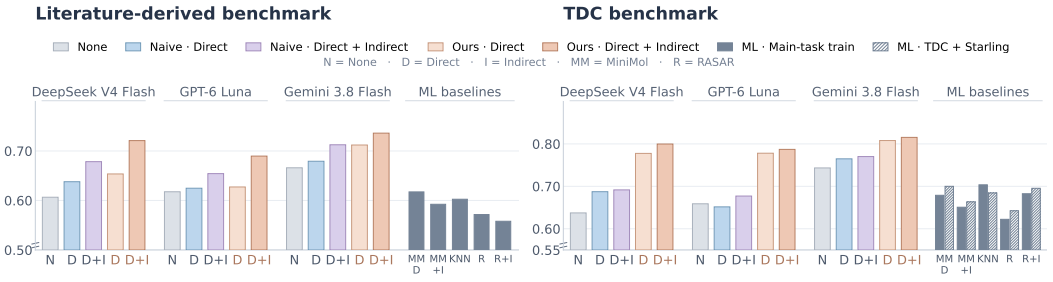


Figure 3: Average macro-F1 on the literature-based benchmark (left) and TDC (right).

cinogenicity, each represented as a binary prediction problem. We evaluate on two benchmark collections. The first is TDC (Huang et al., 2021). TDC provides established benchmarks for these tasks and allows for direct comparison with prior work; however, some of these datasets are quite small and have known issues: limited chemical-space coverage and label errors on the order of 5–10% of examples (Minerali et al., 2020; Jones et al., 2026). Further, TDC omits the experimental conditions under which a label was observed. Because we are interested in whether evidence transfers across molecules *and* across conditions, this motivates a complementary evaluation using additional molecules and literature-supported labels.

We therefore use direct evidence collected by Ampere to construct literature-derived benchmarks for all six tasks. These benchmarks retain experimental conditions that can affect the target outcome. For example, oral bioavailability can differ depending on whether a drug is taken with food or on an empty stomach. We associate labels with molecule–condition pairs, allowing the same molecule to have different labels under different conditions. We evaluate on both TDC and our literature-derived benchmarks. Table 3 compares their sizes and condition coverage.

We assign each molecule–condition pair a label by voting over qualified direct observations and retain labels that meet an agreement threshold. We construct scaffold-disjoint splits, and prioritize pairs supported by multiple observations for validation and testing. Label aggregation and split-construction details are provided in Appendix A.3.

No method may access a measurement of the query molecule itself: for both direct and indirect evidence, we exclude every record whose standardized parent molecule matches the query's, across all reported conditions, including salt, protonation, stereochemical, and tautomeric variants. Direct retrieval goes further and excludes any molecule sharing the query's—or any test-set molecule's—Bemis–Murcko scaffold, which would otherwise supply a near-duplicate measurement of the target property. Indirect retrieval keeps those neighbors, as an indirect record reports a different endpoint and cannot stand in for the label. Identity-matching rules are given in Appendix D.

### 3.2 Evaluation Metrics

AUROC is commonly used to evaluate molecular property predictions on TDC (Huang et al., 2021). For reasoning models, however, interpreting the final label probability requires accounting for the preceding reasoning. Given an input $x$ and a reasoning trajectory $R$, this probability is $p_\theta(y \mid x, R)$, whereas the marginal prediction probability is $p_\theta(y \mid x) = \mathbb{E}_{R\sim p_\theta(R|x)}\left[p_\theta(y \mid x, R)\right]$. A single rollout therefore yields a trajectory-conditioned probability, and AUROC computed over such probabilities is not comparable with one computed with a trained classifier's scores. Estimating the marginal probability requires averaging over multiple rollouts, which is computationally expensive. We therefore use a single rollout per query. We report macro-F1 on the binary prediction as our primary metric, which gives equal weight to both classes and is well defined for every method, and report AUROC when possible. Appendix E.2 reports macro-F1 and AUROC for all ML baselines.

### 3.3 Comparison Systems and Ablations

We compare Ampere with MiniMol, a molecular foundation model with strong performance on TDC ADMET benchmarks (Kläser et al., 2024). We evaluate a classifier trained on MiniMol embeddings, an end-to-end variant, and $k$-nearest-neighbor (KNN) baselines. On TDC, we evaluate baselines trained with and without additional data from our benchmark to distinguish increased supervision from the use of indirect evidence.

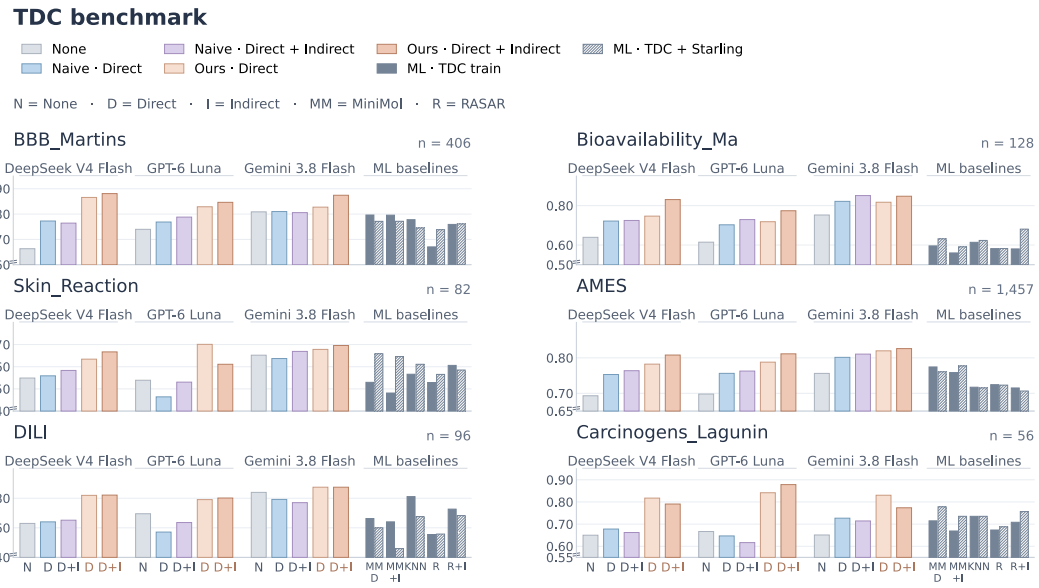


Figure 4: **Test macro-F1 on TDC.** Six-task means are shown separately in Figure 3. Solid and hatched ML bars denote TDC-only and TDC + Starling training, respectively. Naive LLM baselines retrieve 10 direct molecules (or molecule–condition pairs for our literature-based benchmark) or the same 10 plus 50 indirect records in decreasing order of molecular similarity. ML baselines show the maximum test score among their variants as descriptive upper bounds, separately for TDC-only and TDC + Starling training pools; complete variants and AUROC (if available) are reported in Appendix E. Complete scores for Ours are reported in Appendix G.

We construct two baselines that incorporate indirect measurements. *MiniMol + indirect* jointly trains the encoder on the target task and auxiliary tasks defined by binary or numerical indirect measurements. *RASAR + indirect* adapts the data-fusion approach of Luechtefeld et al. (2018), combining direct-neighbor labels, indirect measurements, and molecular similarities as inputs to a random forest. We include a matched *RASAR* control using only the three direct-neighbor label/similarity pairs. Training details, condition-prioritized variants, and complete results are provided in Appendix E.

Our LLM experiments vary two factors independently: how evidence is *selected*, and what evidence is *available*. Selection is either `Naive`, which ranks candidates by Morgan fingerprint similarity to the query and fills the budget in decreasing order, or `Ours`, the context optimization of Section 2.3. Availability is either `Direct` records only or `Direct + Indirect`. Direct and indirect evidence have separate budgets of 10 molecules and 50 records, respectively, and `Direct + Indirect` retains the same direct records as `Direct`. Each setting differs only in which records are in context.

We additionally include a `None` control that receives no retrieved evidence. `None` measures what the backbone already knows: a number of these molecules may be named drugs with properties described frequently in public literature, so some amount of the score will represent memorization rather than reasoning. We repeat all experiments with DeepSeek-V4-Flash, GPT-6 Luna, and Gemini 3.8 Flash.

## 4 Results and Ablations

Our main result, shown in Figures 3, 4 and 5, compares Ampere with existing methods for property prediction. Across model families and both benchmarks, LLM-based predictors using our evidence selection outperform fingerprint or manually engineered feature methods in KNN and random-forest classifiers. On the other hand, LLM-based predictors often underperform more traditional models under naive settings. For example, DeepSeek-V4-Flash and GPT-6 Luna both underperform KNN-based models without our method on the TDC benchmark. **Gemini 3.8 Flash with our selected indirect evidence outperforms the best ML baselines by roughly 11–12 points across both benchmarks.** Per task, our LLM-based approach outperforms traditional methods on 5 of 6 TDC tasks and on all literature-based evaluations. Gains on the bioavailability tasks are particularly strong, reaching 16.7 macro-F1 points on TDC and 18.4 on the literature-derived benchmark.

Table 1: Ablation independently removes the indicated score from the optimized objective. Scores are DeepSeek-V4-Flash's macro-F1 averaged across the six literature-derived tasks.

| Configuration | Mean Macro-F1 | Change |
|---|---|---|
| Our full method | 0.721 | |
| Without learned transfer score | 0.704 | $-0.017$ |
| Without informativeness | 0.714 | $-0.007$ |
| Without structural and molecular coverage | 0.711 | $-0.010$ |
| Without evidence-factor and semantic coverage | 0.714 | $-0.007$ |
| Without all terms except structural similarity (Morgan fingerprint) | 0.678 | $-0.043$ |

**Evaluating indirect evidence usage.** Adding indirect evidence improves our systems across every LLM in both TDC and literature benchmarks (Figure 3). In contrast, MiniMol given the same indirect measurements as auxiliary multi-task targets falls *below* its direct-only counterpart on both benchmarks, and RASAR gains on TDC only to lose ground on the literature-derived tasks. Our results show that integrating indirect evidence is difficult for traditional methods but easy with LLMs.

**System ablations.** In Figure 3, we ablate our evidence integration approach and find it outperforms naive similarity-based selection across all evaluated LLMs on average across evaluation datasets. In Table 1, we ablate parts of the optimization objective. Each term helps, indicating that all main system components matter, with the learned transfer score contributing the most (0.721 to 0.704).

## 5 Related Work

**Scientific agents and literature processing.** ChemCrow and Coscientist use LLMs to coordinate chemical tools and experimental workflows, while Biomni supports a broader range of biomedical research tasks (M. Bran et al., 2024; Boiko et al., 2023; Huang et al., 2025). For literature-based research, PaperQA2 and OpenScholar retrieve and synthesize publications into evidence-supported answers, while Starling extracts structured biomedical datasets from the literature (Skarlinski et al., 2024; Asai et al., 2026; Jones et al., 2026).

**Language models for molecular prediction and reasoning.** TxGemma fine-tunes Gemma 2 for therapeutic property prediction and conversational explanation (Wang et al., 2025a). FutureHouse's ether0 uses reinforcement learning on experimentally grounded tasks to generate molecular structures through natural-language reasoning (Narayanan et al., 2025). Chem-R combines chemical foundation training, reasoning distillation, and multi-task reinforcement learning for molecular and reaction tasks (Wang et al., 2025b). Focusing on property prediction, MPPReasoner combines supervised reasoning traces with reinforcement learning guided by chemical principles (Zhuang et al., 2025).

**Analogical reasoning.** Yasunaga et al. (2024) introduces analogical prompting, in which language models generate relevant examples before solving a query. Lewis & Mitchell (2024) evaluates analogical reasoning through counterfactual tasks, while Musker et al. (2025) compares models' analogical behavior with human reasoning.

**Retrieval for molecular prediction.** Retrieval-based approaches use molecular examples or external scientific knowledge to support prediction. RASAR combines structural similarity with supervised learning for chemical read-across, incorporating information across multiple properties in its data-fusion variant (Luechtefeld et al., 2018). MolRAG retrieves structurally similar molecules and their known property values to guide LLM-based property prediction (Xian et al., 2025). ChemRAG benchmarks retrieval of textual knowledge from scientific literature, databases, and other resources across chemistry tasks (Zhong et al., 2025). Medex (Jones et al., 2025) distills facts about small molecules and proteins from the literature and trains CLIP-style models (Radford et al., 2021) to align text and structure for property prediction.

**Context selection for LLMs.** Beyond retrieving individually relevant examples, context-selection methods consider how examples can contribute jointly. CEIL formulates in-context example selection as a subset-selection problem, using determinantal point processes to model interactions among examples (Ye et al., 2023). Coverage-based example selection chooses sets that collectively cover relevant aspects of the input, rather than ranking examples independently (Gupta et al., 2023).

## 6 CONCLUSION

We presented an analogical reasoning framework for molecular property prediction that integrates direct and indirect evidence from the scientific literature. By combining learned property-transfer scores with context optimization, it selects relevant, informative, and diverse evidence for LLM reasoning. Across six ADME and safety tasks on two benchmark collections, the framework improves mean macro-F1 over supervised baselines and similarity-based retrieval, with consistent average gains from indirect evidence. These results demonstrate the potential of analogical reasoning to integrate heterogeneous experimental findings into effective molecular property predictions.

## AI USE STATEMENT

In this work, we used generative AI tools for implementing methods (as a coding assistant), extracting claims from literature, and cleaning and reformatting those claims. We have not used generative AI tools for developing theoretical models or conceptual frameworks, proposing or refining hypotheses, designing or providing feedback on research methodology or experiments, or interpreting results, and formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, assisting with translation, and supporting qualitative and thematic data analysis are not applicable to this work. Additionally, we used generative AI tools for creating and editing software code and editing the paper to improve readability (grammar and spelling). We did not use generative AI tools to brainstorm or generate research ideas. We have reviewed all AI-assisted work. All AI-assisted code was reviewed, tested, and verified for correctness by the authors; AI-assisted dataset cleaning steps were manually inspected. All grammar and spelling edits were reviewed to ensure they did not alter the technical meaning of the text. We take responsibility for the final content of this work, including text and artifacts produced with the aid of generative AI.

## ETHICS STATEMENT

This work does not involve human subjects, and no IRB approval was required. All data used in this study are derived from publicly available biomedical literature indexed in PubMed and contain no personally identifiable or patient information. We accessed this literature in accordance with NCBI's terms of use; any released data or artifacts will be distributed in compliance with the licensing terms of the source articles. Claims extracted by our method are produced automatically and may be inaccurate, incomplete, or stripped of important context, and the underlying literature may itself contain errors, retracted findings, or publication biases (e.g., toward positive results or well-studied populations). Our system is therefore intended to support research and literature exploration, not to inform clinical decision-making, and its outputs should be verified against the original sources by domain experts. We are not aware of any conflicts of interest or sponsorship that could bias this work.

## REPRODUCIBILITY STATEMENT

We provide the complete implementation and instructions for reproducing our experiments in an anonymized code repository at `https://anonymous.4open.science/r/TxAgent-E39F/`.

## REFERENCES

Akari Asai, Jacqueline He, Rulin Shao, Weijia Shi, Amanpreet Singh, Joseph Chee Chang, Kyle Lo, Luca Soldaini, Sergey Feldman, Mike D'Arcy, David Wadden, Matt Latzke, Jenna Sparks, Jena D. Hwang, Varsha Kishore, Minyang Tian, Pan Ji, Shengyan Liu, Hao Tong, Bohao Wu, Yanyu Xiong, Luke Zettlemoyer, Graham Neubig, Daniel S. Weld, Doug Downey, Wen-tau Yih, Pang Wei Koh, and Hannaneh Hajishirzi. Synthesizing scientific literature with retrieval-augmented language models. *Nature*, 650(8103):857–863, 2026. doi: 10.1038/s41586-025-10072-4. URL `http://dx.doi.org/10.1038/s41586-025-10072-4`.

Lei Bai, Zhongrui Cai, Yuhang Cao, Maosong Cao, Weihan Cao, Chiyu Chen, Haojiong Chen, Kai Chen, Pengcheng Chen, Ying Chen, et al. Intern-s1: A scientific multimodal foundation model. *arXiv preprint arXiv:2508.15763*, 2025.

Daniil A. Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. Autonomous chemical research with large language models. *Nature*, 624(7992):570–578, 2023. doi: 10.1038/s41586-023-06792-0. URL http://dx.doi.org/10.1038/s41586-023-06792-0.

Shivanshu Gupta, Matt Gardner, and Sameer Singh. Coverage-based Example Selection for In-Context Learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 13924–13950. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.findings-emnlp.930. URL https://aclanthology.org/2023.findings-emnlp.930/.

Kexin Huang, Tianfan Fu, Wenhao Gao, Yue Zhao, Yusuf Roohani, Jure Leskovec, Connor W. Coley, Cao Xiao, Jimeng Sun, and Marinka Zitnik. Therapeutics Data Commons: Machine Learning Datasets and Tasks for Drug Discovery and Development. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/file/4c56ff4ce4aaf9573aa5dff913df997a-Paper-round1.pdf.

Kexin Huang, Serena Zhang, Hanchen Wang, Yuanhao Qu, Yingzhou Lu, Yusuf Roohani, Ryan Li, Lin Qiu, Junze Zhang, Yin Di, et al. Biomni: A General-Purpose Biomedical AI Agent. *bioRxiv*, 2025. doi: 10.1101/2025.05.30.656746. URL https://doi.org/10.1101/2025.05.30.656746.

Haydn Jones, Natalie Maus, Josh Magnus Ludan, Maggie Ziyu Huan, Jiaming Liang, Marcelo Der Torossian Torres, Jiatao Liang, Zachary Ives, Yoseph Barash, Cesar de la Fuente-Nunez, Jacob R. Gardner, and Mark Yatskar. A Dataset for Distilling Knowledge Priors from Literature for Therapeutic Design. In *Advances in Neural Information Processing Systems*, volume 38. Curran Associates, Inc., 2025. doi: 10.52202/085713-1782. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/4d4e1e0ce7e201cfa738b21a48d38fdc-Abstract-Datasets_and_Benchmarks_Track.html.

Haydn Jones, Yimeng Zeng, Alden Rose, Li S. Yifei, Yining Huang, Kaiwen Wu, Jiaming Liang, Maggie Ziyu Huan, Yoseph Barash, Cesar de la Fuente-Nunez, Osbert Bastani, Zachary Ives, Mark Yatskar, and Jacob R. Gardner. Self-driving datasets: From 20 million papers to nuanced biomedical knowledge at scale, 2026. URL https://arxiv.org/abs/2605.07022v3. Version 3.

Kerstin Kläser, Błażej Banaszewski, Samuel Maddrell-Mander, Callum McLean, Luis Müller, Ali Parviz, Shenyang Huang, and Andrew Fitzgibbon. MiniMol: A parameter-efficient foundation model for molecular learning. In *ICML Workshop on Accessible and Efficient Foundation Models for Biological Discovery*, 2024. URL https://arxiv.org/abs/2404.14986.

Martha Lewis and Melanie Mitchell. Using Counterfactual Tasks to Evaluate the Generality of Analogical Reasoning in Large Language Models. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, 2024. URL https://escholarship.org/uc/item/58d9s666.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33: 9459–9474, 2020.

Thomas Luechtefeld, Dan Marsh, Craig Rowlands, and Thomas Hartung. Machine Learning of Toxicological Big Data Enables Read-Across Structure Activity Relationships (RASAR) Outperforming Animal Test Reproducibility. *Toxicological Sciences*, 165(1):198–212, 2018. doi: 10.1093/toxsci/kfy152. URL http://dx.doi.org/10.1093/toxsci/kfy152.

Andres M. Bran, Sam Cox, Oliver Schilter, Carlo Baldassari, Andrew D. White, and Philippe Schwaller. Augmenting large language models with chemistry tools. *Nature Machine Intelligence*, 6(5):525–535, 2024. doi: 10.1038/s42256-024-00832-8. URL http://dx.doi.org/10.1038/s42256-024-00832-8.

Eni Minerali, Daniel H. Foil, Kimberley M. Zorn, Thomas R. Lane, and Sean Ekins. Comparing machine learning algorithms for predicting drug-induced liver injury (DILI). *Molecular Pharmaceutics*, 17(7):2628–2637, 2020. doi: 10.1021/acs.molpharmaceut.0c00326. URL https://pmc.ncbi.nlm.nih.gov/articles/PMC7702310/.

Michel Minoux. Accelerated greedy algorithms for maximizing submodular set functions. In J. Stoer (ed.), *Optimization Techniques*, pp. 234–243, Berlin, Heidelberg, 1978. Springer. ISBN 978-3-540-35890-9. doi: 10.1007/BFb0006528.

Sam Musker, Alex Duchnowski, Raphaël Millière, and Ellie Pavlick. LLMs as models for analogical reasoning. *Journal of Memory and Language*, 145:104676, 2025. doi: 10.1016/j.jml.2025.104676. URL http://dx.doi.org/10.1016/j.jml.2025.104676.

Siddharth M. Narayanan, James D. Braza, Ryan-Rhys Griffiths, Albert Bou, Geemi Wellawatte, Mayk Caldas Ramos, Ludovico Mitchener, Samuel G. Rodriques, and Andrew D. White. Training a Scientific Reasoning Model for Chemistry. *arXiv preprint arXiv:2506.17238*, 2025. URL https://arxiv.org/abs/2506.17238.

G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming*, 14(1):265–294, December 1978. ISSN 1436-4646. doi: 10.1007/BF01588971. URL https://doi.org/10.1007/BF01588971.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 8748–8763. PMLR, July 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of chemical information and modeling*, 50(5):742–754, 2010.

Michael D. Skarlinski, Sam Cox, Jon M. Laurent, James D. Braza, Michaela Hinks, Michael J. Hammerling, Manvitha Ponnapati, Samuel G. Rodriques, and Andrew D. White. Language agents achieve superhuman synthesis of scientific knowledge. *arXiv preprint arXiv:2409.13740*, 2024. URL https://arxiv.org/abs/2409.13740.

Duxin Sun, W. Gao, Hongxiang Hu, and Simon Yuji Zhou. Why 90% of clinical drug development fails and how to improve it? *Acta Pharmaceutica Sinica. B*, 12:3049 – 3062, 2022. URL https://api.semanticscholar.org/CorpusID:246783953.

Eric Wang, Samuel Schmidgall, Paul F. Jaeger, Fan Zhang, Rory Pilgrim, Yossi Matias, Joelle Barral, David Fleet, and Shekoofeh Azizi. TxGemma: Efficient and Agentic LLMs for Therapeutics. *arXiv preprint arXiv:2504.06196*, 2025a. URL https://arxiv.org/abs/2504.06196.

Weida Wang, Benteng Chen, Di Zhang, Wanhao Liu, Shuchen Pu, Ben Gao, Jin Zeng, Xiaoyong Wei, Tianshu Yu, Shuzhou Sun, Tianfan Fu, Wanli Ouyang, Lei Bai, Jiatong Li, Zifu Wang, Yuqiang Li, and Shufei Zhang. Chem-R: Learning to Reason as a Chemist. *arXiv preprint arXiv:2510.16880*, 2025b. URL https://arxiv.org/abs/2510.16880.

Ziting Xian, Jiawei Gu, Lingbo Li, and Shangsong Liang. MolRAG: Unlocking the Power of Large Language Models for Molecular Property Prediction. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15513–15531. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.755. URL https://aclanthology.org/2025.acl-long.755/.

Michihiro Yasunaga, Xinyun Chen, Yujia Li, Panupong Pasupat, Jure Leskovec, Percy Liang, Ed H. Chi, and Denny Zhou. Large Language Models as Analogical Reasoners. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=AgDICX1h50.

Jiacheng Ye, Zhiyong Wu, Jiangtao Feng, Tao Yu, and Lingpeng Kong. Compositional Exemplars for In-context Learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 39818–39833. PMLR, 2023. URL https://proceedings.mlr.press/v202/ye23c.html.

Xianrui Zhong, Bowen Jin, Siru Ouyang, Yanzhen Shen, Qiao Jin, Yin Fang, Zhiyong Lu, and Jiawei Han. Benchmarking Retrieval-Augmented Generation for Chemistry. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=qG4dL0bart.

Jiaxi Zhuang, Yaorui Shi, Jue Hou, Yunong He, Mingwei Ye, Mingjun Xu, Yuming Su, Linfeng Zhang, Ying Qian, Linfeng Zhang, Guolin Ke, and Hengxing Cai. Reasoning-Enhanced Large Language Models for Molecular Property Prediction. *arXiv preprint arXiv:2510.10248*, 2025. URL https://arxiv.org/abs/2510.10248.

# A Evidence Corpus and Benchmark Construction

## A.1 Evidence Corpus Cleaning

**Source-field normalization.** We map fields from different Starling runs to a shared format for the reported endpoint, result, unit, assay, and species. We also retain experimental conditions, supporting text, and source identifiers. This standardizes record organization while preserving what was measured or predicted and under which conditions.

**Molecular identity.** We use RDKit to standardize molecular structures and assign parent identifiers, allowing records for the same parent molecule to be linked across collections and kept in the same data split. We retain canonical isomeric SMILES and derive parent identifiers after structure cleanup, fragment-parent selection, and neutralization. Records with invalid structures or unresolved parent identities are excluded.

We additionally require the recorded structure to match the parent structure returned by a PubChem lookup of the molecule name. Candidates with unresolved name–structure assignments are excluded from label construction and retrieval. Indirect records without molecule names retain their source-provided structures and are marked as not verified by name.

**Task eligibility.** Filters inspect the reported endpoint, assay, conditions, and supporting text. For skin sensitization, they exclude photo-dependent effects, irritation-only records, and non-contact cutaneous reactions.

**Duplicate handling.** In the uncapped retrieval index, records within each assay–molecule group are deduplicated by their complete evidence representation: endpoint, result, unit, assay, species, qualifying conditions, supporting text, and evidence-category metadata. Differences in these fields preserve separate records. Retrieval deduplication therefore removes repeated representations, while study-level aggregation prevents repeated descriptions from multiplying label support.

**Reviewed corrections.** Targeted source reviews correct or exclude identified attribution errors. For example, six Bioavailability records naming nitrendipine are excluded because their attached structures represent a different molecule. In Skin, a review of one study corrected 17 molecule–record assignments while preserving endpoints, outcomes, doses, durations, and supporting text. Corrections are bound to the reviewed source identifiers and original content, and frozen releases retain change records and input/output hashes.

## A.2 Evidence Groups

Table 2 lists the evidence groups used for each task. Records are assigned individually according to their reported endpoints and supporting text. Direct groups contain observations eligible to supply target labels; related outcomes that do not meet these criteria remain in the corresponding indirect groups.

Table 2: Evidence groups and their scope. Oral BA denotes oral bioavailability; Skin denotes skin sensitization.

| Task | Evidence group | Type | Scope |
|---|---|---|---|
| BBB | Measured CNS access | Direct | Measured brain or central nervous system exposure after systemic administration. |
| | Additional CNS-access evidence | Indirect | Other CNS outcomes, predicted BBB outcomes, and functional observations suggesting brain access. |
| | Passive permeability | Indirect | Measured or predicted passage across membranes without active transport. |
| | Efflux transport | Indirect | Transport out of cells or brain, including transporter substrate and inhibition results. |
| | Influx transport | Indirect | Transporter-mediated uptake into cells or brain. |
| Oral BA | Absolute oral bioavailability | Direct | Measured fraction of an oral dose reaching systemic circulation. |
| | Additional bioavailability evidence | Indirect | Relative, comparative, or other bioavailability results that cannot supply a direct label. |
| | Oral drug exposure | Indirect | Blood concentration over time and peak concentration after oral dosing. |
| | Absorption and permeability | Indirect | Solubility, dissolution, membrane permeability, and intestinal absorption. |
| | Intestinal transport and metabolism | Indirect | Efflux and metabolism in the intestinal wall. |
| | Liver clearance and metabolic stability | Indirect | Removal or breakdown of compounds by the liver. |
| Skin | Measured sensitization | Direct | Measured skin sensitization or contact-allergy outcomes. |
| | Additional sensitization outcomes | Indirect | Other measured outcomes, predictions, and combined-assay classifications that cannot supply a direct label. |
| | Sensitization-related responses | Indirect | Protein reactivity and skin or immune-cell responses involved in sensitization. |
| Ames | Measured bacterial mutation | Direct | Measured bacterial reverse-mutation outcomes under the reported strain and activation conditions. |
| | Additional mutation outcomes | Indirect | Other bacterial or unspecified mutagenicity outcomes, predictions, and mixed reports that cannot supply a direct label. |
| | Other genetic damage | Indirect | Other gene mutations and chromosome damage. |
| | DNA damage and repair | Indirect | DNA lesions, repair, and cellular responses to DNA damage. |
| | Processes affecting genetic damage | Indirect | Metabolic activation, detoxification, DNA interactions, antioxidant defense, and chromosome segregation. |
| DILI | Human liver-injury outcomes | Direct | Human drug-induced liver-injury outcomes eligible for target labels. |
| | Additional liver-injury evidence | Indirect | Other liver-injury outcomes, clinical signals, predictions, and mixed reports that cannot supply a direct label. |
| | Liver-cell function | Indirect | Cell survival, membrane integrity, and liver-cell functional capacity. |
| | Bile transport and regulation | Indirect | Bile-acid transport, accumulation, and bile secretion. |
| | Reactive metabolites and cellular energy | Indirect | Formation of reactive metabolites and effects on mitochondrial energy production. |
| | Immune and molecular responses | Indirect | Inflammation, immune interactions, and changes in gene, protein, or metabolite profiles. |
| | Cellular stress | Indirect | Oxidative stress and disturbances in lipid balance, calcium regulation, or cellular recycling. |

Continued on next page

Table 2 (continued)

| Task | Evidence group | Type | Scope |
|---|---|---|---|
| Carcinogens | Measured carcinogenicity | Direct | Observed carcinogenicity outcomes under the reported species condition. |
| | Additional carcinogenicity evidence | Indirect | Other tumor outcomes, hazard assessments, predictions, and mixed reports that cannot supply a direct label. |
| | Cell transformation and growth | Indirect | Cell transformation, uncontrolled growth, survival, and altered cell-cycle control. |
| | Genetic damage and repair | Indirect | Mutations, chromosome damage, DNA lesions, and repair responses. |
| | Reactive metabolism and oxidative injury | Indirect | Reactive metabolites, chemical reactivity, and oxidative damage. |
| | Gene regulation and cell communication | Indirect | Epigenetic changes and disrupted communication that normally restrains cell growth. |
| | Regeneration and growth signaling | Indirect | Injury-associated regeneration, receptor-driven growth, and altered cell differentiation. |

### A.3 Conditioned Benchmark Construction

**Condition-specific labels.** We construct literature-derived benchmarks for BBB penetration, oral bioavailability, skin sensitization, Ames mutagenicity, DILI, and carcinogenicity. Task-specific rules identify observations that directly establish the target outcome and normalize their reported conditions into task-relevant categories. Observations are grouped by molecular parent and condition; missing conditions are explicitly marked as unreported, without assuming a particular experimental setting. Ames groups retain the reported strain panel and metabolic activation status. Results reported jointly for both activation settings remain pooled and are not duplicated into separate positive- and negative-S9 observations.

**Vote aggregation.** After task-specific duplicate handling (Appendix A.1), each accepted voting unit contributes one binary outcome. Within each parent–condition group, we assign the majority label if at least 60% of votes agree for explicitly reported conditions, or 70% when no external condition is reported; ties are excluded. All retained Ames and Carcinogens rows have explicit conditions and therefore use the 60% threshold. Source identifiers, vote counts, and minority outcomes are retained for auditing. Votes represent accepted source records or, for Ames, distinct study–parent–condition units; multiple votes need not represent independent studies in every task.

**Evaluation-set construction.** All conditions for a molecular parent remain in one split. Scaffold splits additionally keep Bemis–Murcko scaffolds disjoint; random splits group by parent. Subject to split-size and condition-coverage constraints, allocation first minimizes validation and test rows supported by only one vote, then balances these rows between the two sets, and finally balances labels. This prioritizes corroborated observations without treating vote count as a guarantee of correctness. Conditions retained in the benchmark must occur in training, validation, and test; rows with empty scaffolds are restricted to training in the scaffold split. These procedures construct the literature-derived benchmarks; the TDC evaluations retain their externally supplied labels.

## B Property-Transfer Model: Details

We provide further details of how literature observations become training examples for the property-transfer model. We construct separate direct and indirect training sets for each task.

### B.1 Direct and Indirect Labels

**Direct labels.** Direct evidence measures the target property itself, so records from a task's training set can be paired without reconciling different conditions. For literature-derived binary labels supported by several observations, we use the fraction of positive observations as the scalar outcome. For example, eight positive reports among ten qualifying reports give a value of $0.8$. This retains the degree of agreement among source records rather than reducing every such molecule to a hard label. For TDC, we do not have access to such data, so we assign a transfer label of 0.95 when two labels agree and 0.5 when they disagree.

Table 3: Statistics of our literature-derived benchmarks. Molecule counts refer to distinct molecular structures; samples in our benchmarks are labeled molecule–condition pairs. The condition count is the number of distinct reported experimental condition combinations.

| | TDC | | Our benchmarks | | |
|---|---|---|---|---|---|
| Task | Molecules | Samples | Molecules | Conditions | Samples |
| BBB | 1,975 | 2,030 | 3,732 | 8 | 3,843 |
| Oral bioavailability | 640 | 640 | 2,133 | 7 | 2,487 |
| Skin sensitization | 404 | 404 | 2,388 | 1 | 2,421 |
| Ames | 7,255 | 7,278 | 1,383 | 73 | 2,474 |
| DILI | 475 | 475 | 2,400 | 37 | 4,024 |
| Carcinogenicity | 278 | 280 | 3,484 | 5 | 4,692 |

**Indirect labels.** Indirect evidence instead contains measurements from heterogeneous experiments that may inform the target property. We form comparisons only among records that measure the same underlying quantity under compatible conditions. The resulting endpoint groups specify which outcomes can be compared. We estimate the standard deviation of the outcome with respect to the endpoint which is used to normalize the absolute differences when forming them into transfer labels.

### B.2 Grouping Records into Training Examples

We first identify the fields in our literature extraction that define a measurement, including its assay system, unit, and relevant experimental conditions. Records often express the same field value in different ways; for example, a biological context may be described by a species name or by a more specific population within that species. We use language-model-assisted normalization to consolidate equivalent expressions while retaining distinctions that could change the interpretation of a measurement. Specifically, for a key subset of fields, we embed distinct values with all-MiniLM-L6-v2 and use $k$-means recursively until each cluster contains at most 50 values to assemble similar values for a language model to review and output a mapping of merged lists. We perform this process twice. Unit strings undergo a separate two-pass reconciliation: character-level text features are used to cluster recursively until each batch contains at most 50 values, and a language model maps equivalent units to canonical forms. All LLM-based steps in this appendix use GPT-5.4-mini.

**Endpoint cleaning.** Before comparing outcomes, we use an LLM to generate binary judgements that distinguish absolute measurements from relative ones, such as fold changes against a comparator, because these cannot generally be compared on the same scale. We also ask an LLM for binary judgments whether an endpoint's measurement unit should be understood in a logarithmic scale to obtain a more stable standard deviation, considering whether its measurements are multiplicative (e.g., fold changes) or additive (e.g., differences in means). To identify possible extraction or comparability errors, we flag observations more than five standard deviations from the endpoint mean and ask the LLM to judge whether each is valid. Observations judged invalid are excluded from training but remain available for retrieval. Records with only free-text statements do not contribute training pairs.

**Endpoint filtering.** We retain endpoints with at least 12 training observations from at least six distinct parent molecules for more reliable estimates of the standard deviation and sufficient molecular coverage. We also assess whether repeated measurements of one molecule are sufficiently consistent relative to differences between molecules. Where this ratio can be estimated, endpoints whose within-molecule variance exceeds $0.75$ times the variance of molecule-level means are excluded. This avoids training on groups for which experimental or extraction variation overwhelms the molecular differences the model is meant to detect, as it likely indicates a poorly constructed endpoint.

### B.3 Sampling transfer pairs

Within each eligible endpoint, we pair records from different parent molecules. We sample both pairs with relatively similar outcomes and pairs with larger outcome differences in a balanced manner, so the model sees examples of warranted and unwarranted transfer.

We cap the number of pairs drawn from an endpoint and the participation of individual records and molecules. These bounds keep large endpoints and heavily measured molecules from dominating

training, while allowing smaller endpoints to contribute. Records that overlap with the test set are excluded from training.

### B.4 TRAINING

**Loss.** For a numeric pair with outcomes $y$ and $y'$, let $s_e$ be the standard deviation of comparable training outcomes in endpoint $e$. The target probability of transfer is

$$t = \frac{1}{1 + e^{-10\left(0.4 - \frac{|y-y'|}{s_e}\right)}}. \tag{6}$$

Thus, smaller differences yield larger transfer targets. The reference record exposes its observed outcome, whereas the query molecule's outcome is withheld from the model. The task is formulated as multiple choice in the prompt of the LM, with $A$ the transfer token and $B$ the non-transfer token, and we minimize the soft cross-entropy $-t \log P(A) - (1-t) \log P(B)$ over the model's probabilities.

**Hyperparameters.** We fine-tune the Intern-S1-mini (Qwen-8B with science-specific pre-training) with a base learning rate of $2 \times 10^{-5}$. We use a batch size of 256 sequences of 4096 tokens long. The learning rate for the SMILES-token embeddings is multiplied by $1.5$ to adapt the molecular representation more quickly. The selected checkpoint is chosen using validation data.

## C CONTEXTUALIZATION MODEL: DETAILS

### C.1 INFORMATIVENESS

**Construction of semantic groups.** Here the question is whether two kinds of evidence address the same scientific aspect of a target property, even if their individual measurements are not numerically interchangeable. Within each corpus of evidence, we start from constructed endpoints and, first, group across columns that are less relevant for semantics (e.g., units), and second, further cluster across descriptions with respect to the endpoint's relation to the target property.

**Scoring of semantic groups.** We ask a language model to assign each group an informativeness weight in $[0, 1]$ for the target property, using representative descriptions and records with a common scoring rubric. Groups are first scored in small batches. We sort them by these initial scores and recalibrate them in overlapping windows, carrying previously scored groups as anchors so that weights remain comparable across windows. We repeat independent judgments 3 times to reduce sensitivity to an individual response. The resulting weight describes the relevance of a kind of evidence to the task, independently of how many records happen to belong to that group.

## D EXPERIMENTAL DETAILS

**Molecular identity exclusions.** We standardize structures with RDKit, remove counterions through fragment-parent selection, and normalize charge states. Auxiliary training excludes any record whose standardized parent SMILES or parent InChIKey connectivity block matches a molecule in the validation or test set. The connectivity check also excludes stereochemical and isotopic variants. This exclusion is applied across all auxiliary measurement groups before target aggregation and normalization.

For retrieval, DILI and Carcinogens additionally exclude their task-specific tautomer identity groups. Invalid or unresolved molecular identities are excluded during candidate preparation. These checks supplement the task-specific source-eligibility and held-out direct-label filters.

**Auxiliary targets.** The auxiliary source export groups records by comparable endpoints, measurement types, and experimental descriptors. Groups are retained when they contain nonmissing binary or numerical measurements for at least ten distinct standardized molecules. This minimum is applied before validation/test identity exclusion; groups that subsequently fall below ten molecules are retained. Within each molecule–group pair, continuous observations are aggregated by their median, while binary observations are represented by the fraction of positive outcomes. Continuous targets are standardized using the mean and standard deviation of the aggregated auxiliary training targets within each group. Constant groups use a scale of one, and no logarithmic transformation

is applied. Continuous heads use mean squared error and binary heads use binary cross-entropy; only observed targets contribute to the loss. Each main-task batch is paired with an auxiliary batch sampled uniformly over groups and then molecules within each group, with replacement.

**MiniMol training and selection.** The frozen, fine-tuned, and auxiliary-supervised variants use the same task-specific classification head and condition features on the literature-derived benchmarks. Both fine-tuned variants update the complete encoder; auxiliary heads are linear projections of its 512-dimensional embedding. Training uses five seeds and 25 epochs on the training split only. For each seed, the checkpoint is selected by target-task validation binary cross-entropy. Auxiliary loss weights are selected from $\{0.1, 0.6, 1\}$ by the five-seed ensemble's target-task validation binary cross-entropy. The classification threshold is then selected by ensemble validation macro-F1. There is no training-plus-validation refit. Selected hyper-parameters are in Table 4.

Table 4: Task-head hyperparameters and auxiliary loss weights selected on validation. Lit.: literature-derived benchmark; Mixed: TDC + Starling training labels. The fine-tuned main-only control uses $\lambda = 0$.

| | | | | Selected $\lambda$ | | |
|---|---|---|---|---|---|---|
| Task | Width | Depth | Head LR | Lit. | TDC | Mixed |
| BBB | 2048 | 3 | $10^{-4}$ | 1 | 0.6 | 0.6 |
| Oral | 512 | 3 | $3 \times 10^{-4}$ | 0.1 | 0.1 | 0.1 |
| Skin | 512 | 3 | $3 \times 10^{-4}$ | 1 | 0.6 | 1 |
| Ames | 512 | 3 | $10^{-4}$ | 0.6 | 0.1 | 0.1 |
| DILI | 512 | 4 | $5 \times 10^{-4}$ | 0.1 | 0.6 | 0.1 |
| Carcinogens | 512 | 3 | $3 \times 10^{-4}$ | 0.6 | 1 | 1 |

**Nearest-neighbor prediction.** Morgan fingerprints use radius two and 2,048 bits; MiniMol neighbors use frozen embeddings. References come from the designated training split, with each molecular identity contributing at most once. We fix $k = 3$ for every task and variant after doing validation search. Condition-prioritized variants first use matching conditions, followed by unreported conditions; any additional fallback follows the recorded task-specific protocol.

# E COMPLETE ML BASELINE TEST RESULTS

We report all eleven ML baseline variants for six tasks on the literature-derived benchmark and on TDC, with two TDC training pools. Tables 5 to 7 contain all 198 task–method entries. Each table gives macro-F1 and AUROC to four decimal places; the same test examples are used for every method within a task and benchmark. These tables use the final validation-selected MiniMol protocol and KNN baselines.

## E.1 BASELINE SETTINGS

**Training pools and condition features.** The literature-derived MiniMol models use the designated scaffold training split and training-derived condition one-hot features. TDC-only MiniMol models use the original TDC training labels and no condition features. The mixed pool adds the existing Starling target labels to TDC training, with equal weight per row and the source-specific condition vocabulary used by the frozen-head control. Both TDC pools share exactly the same validation and test molecules and labels. The remaining tasks retain their existing external test cohorts, including supplied ADMET benchmark test sets where available; their scaffold train/validation split does not imply that every external test cohort was reconstructed to be scaffold-disjoint.

Added Starling target labels exclude the TDC validation/test parent and scaffold union. Indirect-assay training instead uses the strict parent/connectivity exclusion in Appendix D. Original TDC main-training rows are preserved.

**MiniMol variants and optimization.** MiniMol (frozen) trains only a classification head on the pretrained 512-dimensional representation. MiniMol (fine-tuned) updates the encoder and the same head using target labels. MiniMol + indirect additionally trains one linear auxiliary output per measurement group, using the aggregation, losses, and sampling described in Appendix D. The objective is target-task binary cross-entropy plus $\lambda$ times the sampled auxiliary loss. Skin auxiliary measurements are restricted to assay-transfer-eligible source records present in the active release level map. Source groups must contain at least ten distinct parent molecules before heldout exclusion, yielding 3,287 rows and 67 groups. MiniMol + indirect and RASAR + indirect use this same pool. The level-map filter is not a strict L2+ restriction. Other tasks retain their existing auxiliary pools. Small or constant groups surviving heldout exclusion are retained; the ten-parent requirement is not reapplied.

All three variants use seeds 1–5, 25 training epochs, main-task batch size 32, Adam with weight decay $10^{-4}$, head dropout 0.1, and five warmup epochs followed by cosine decay. Encoder fine-tuning uses learning rate $10^{-5}$, gradient-norm clipping at 5, auxiliary batch size 32 and evaluation batch size 128. Auxiliary heads use the task-head learning rate. Table 4 gives the task-specific head settings and selected auxiliary weights; head depth counts the output layer. Skin and Carcinogens use the Bioavailability head configuration.

Each seed selects its checkpoint by minimum target-task validation binary cross-entropy. For auxiliary training, $\lambda \in \{0.1, 0.6, 1\}$ is selected separately for each task and training pool by the binary cross-entropy of the five-seed mean validation probability. The ensemble decision threshold maximizes validation macro-F1 and is then fixed for test evaluation. Training never includes validation rows, and there is no train-plus-validation refit. AUROC uses the mean test probabilities before thresholding. The reported metrics describe the ensemble, not a mean of five separately thresholded seed metrics.

**Nearest-neighbor variants.** MiniMol KNN ranks frozen embeddings by cosine similarity; Morgan KNN uses Tanimoto similarity over radius-two, 2,048-bit fingerprints, with bond types and without feature or chirality encoding. All variants fix $k = 3$, use one reference per molecular identity, and predict with unweighted neighbor votes at threshold 0.5; AUROC uses the positive-vote fraction. All-train KNN ignores conditions. The variants marked "condition" prioritize matching conditions and then unreported conditions. Only the literature-derived DILI and Carcinogens variants additionally fill from other conditions if needed; the remaining condition-prioritized variants use the available matching/unreported neighbors without that fallback. On TDC-only data, the condition-prioritized and all-train policies are identical, so the repeated rows document equivalent policies rather than independent models.

**RASAR and RASAR + indirect.** Our RASAR adaptation (Luechtefeld et al., 2018) uses six features: the label and Morgan similarity of each of the three direct neighbors. RASAR + indirect

extends this same vector with one nearest-neighbor measurement/similarity pair for each of the $B$ indirect measurement groups, giving $6 + 2B$ features. The indirect targets use the same aggregation and standardization as MiniMol + indirect. The original data-fusion RASAR uses similarities to positive and negative analogs for each binary hazard, together with available query-hazard labels; our adaptation uses neighbor labels or measurements and similarities and requires no experimental labels for the query.

Both methods use all-train or condition-prioritized direct-neighbor membership, followed by decreasing similarity order. Indirect neighbors are selected independently within each group. RASAR reuses exactly the first six features of its matched RASAR + indirect input. Neither method adds condition one-hot features. Missing pairs remain NaN and use the forest's native missing-value handling. For every training, validation, and test query, donors exclude its own standardized parent/connectivity; the indirect donor pool also excludes the full validation/test identity union. This prevents training features from containing the query's own label and may change donors for the pre-existing TDC overlaps described above.

Each candidate averages probabilities from five forests (seeds 1–5), each with 100 trees, Gini splits, bootstrap sampling, unrestricted depth, and no class weighting. For input dimension $d$, we search maximum feature counts of $\lfloor\sqrt{d}\rfloor$ or $\max(1, \lfloor 0.3d \rfloor)$, crossed with minimum leaf sizes $\{1, 5, 20\}$. Here $d = 6$ for RASAR, giving two or one features per split, and $d = 6 + 2B$ for RASAR + indirect. Each method independently selects the candidate minimizing ensemble validation binary cross-entropy and the threshold maximizing validation macro-F1, then evaluates the selected candidate on test. Only target-task training labels fit the forest; there is no train-plus-validation refit. Within each method, the two TDC-only policies have identical features on every split and share a fitted model suite.

### E.2 Complete test performance

We retain all baseline results regardless of their relative test performance. Macro-F1 gives equal weight to the two classes; AUROC is computed from unthresholded prediction scores on the same test rows. No model is selected by test macro-F1 or AUROC. These cohorts were evaluated in earlier experiments and are not presented as untouched test sets. "Oral" denotes oral bioavailability and "Skin" denotes skin sensitization. Each table has two panels of three tasks; $n$ is the number of evaluated examples.

Table 5: Literature-derived benchmark: complete ML test results. Each task reports macro-F1 (F1) and AUROC (AUC). All methods use the settings in Appendix E.1.

| Method | BBB ($n = 393$) | | Oral ($n = 269$) | | Skin ($n = 241$) | |
|---|---|---|---|---|---|---|
| | F1 | AUC | F1 | AUC | F1 | AUC |
| MiniMol (frozen) | 0.6309 | 0.7279 | 0.6276 | 0.7179 | 0.5838 | 0.6127 |
| MiniMol (fine-tuned) | 0.6448 | 0.7069 | 0.6851 | 0.7126 | 0.5346 | 0.6097 |
| MiniMol + indirect | 0.5990 | 0.7036 | 0.6909 | 0.7123 | 0.5709 | 0.6193 |
| MiniMol KNN | 0.5949 | 0.6081 | 0.6153 | 0.6620 | 0.5410 | 0.5617 |
| MiniMol KNN (condition) | 0.6010 | 0.6106 | 0.6184 | 0.6645 | 0.5410 | 0.5617 |
| Morgan KNN | 0.5495 | 0.5892 | 0.6278 | 0.6946 | 0.5108 | 0.5278 |
| Morgan KNN (condition) | 0.5479 | 0.5982 | 0.6542 | 0.7238 | 0.5138 | 0.5355 |
| RASAR | 0.5199 | 0.5490 | 0.6169 | 0.7084 | 0.5146 | 0.5484 |
| RASAR (condition) | 0.5235 | 0.5723 | 0.6398 | 0.7366 | 0.5114 | 0.5503 |
| RASAR + indirect | 0.5110 | 0.5965 | 0.6169 | 0.7100 | 0.5049 | 0.5615 |
| RASAR + indirect (condition) | 0.5540 | 0.6260 | 0.6420 | 0.7162 | 0.5544 | 0.5647 |

| Method | Ames ($n = 274$) | | DILI ($n = 402$) | | Carcinogens ($n = 469$) | |
|---|---|---|---|---|---|---|
| | F1 | AUC | F1 | AUC | F1 | AUC |
| MiniMol (frozen) | 0.6473 | 0.7382 | 0.5999 | 0.6869 | 0.5431 | 0.5904 |
| MiniMol (fine-tuned) | 0.6315 | 0.7486 | 0.5686 | 0.6577 | 0.4737 | 0.5526 |
| MiniMol + indirect | 0.6449 | 0.7556 | 0.5932 | 0.6639 | 0.4552 | 0.5735 |
| MiniMol KNN | 0.5452 | 0.5483 | 0.6005 | 0.6186 | 0.5335 | 0.5511 |
| MiniMol KNN (condition) | 0.6328 | 0.6789 | 0.6334 | 0.6585 | 0.5528 | 0.5817 |
| Morgan KNN | 0.5255 | 0.5665 | 0.5710 | 0.6056 | 0.5101 | 0.5286 |
| Morgan KNN (condition) | 0.5823 | 0.6347 | 0.6215 | 0.6518 | 0.5393 | 0.5417 |
| RASAR | 0.5400 | 0.5726 | 0.5423 | 0.6217 | 0.5402 | 0.5491 |
| RASAR (condition) | 0.6111 | 0.6833 | 0.6003 | 0.6482 | 0.4562 | 0.4960 |
| RASAR + indirect | 0.5398 | 0.6321 | 0.4923 | 0.6109 | 0.4311 | 0.5299 |
| RASAR + indirect (condition) | 0.6102 | 0.6908 | 0.5513 | 0.6272 | 0.4363 | 0.5117 |

Table 6: TDC with TDC training labels: complete ML test results. Each task reports macro-F1 (F1) and AUROC (AUC). All methods use the settings in Appendix E.1.

| Method | BBB ($n = 406$) | | Oral ($n = 128$) | | Skin ($n = 82$) | |
|---|---|---|---|---|---|---|
| | F1 | AUC | F1 | AUC | F1 | AUC |
| MiniMol (frozen) | 0.7930 | 0.9230 | 0.5956 | 0.7140 | 0.5296 | 0.5787 |
| MiniMol (fine-tuned) | 0.7963 | 0.9090 | 0.5657 | 0.6398 | 0.5030 | 0.6006 |
| MiniMol + indirect | 0.7957 | 0.9160 | 0.5599 | 0.6332 | 0.4813 | 0.6067 |
| MiniMol KNN | 0.7778 | 0.8583 | 0.6139 | 0.5961 | 0.5030 | 0.5474 |
| MiniMol KNN (condition) | 0.7778 | 0.8583 | 0.6139 | 0.5961 | 0.5030 | 0.5474 |
| Morgan KNN | 0.7495 | 0.7828 | 0.5541 | 0.6358 | 0.5661 | 0.5784 |
| Morgan KNN (condition) | 0.7495 | 0.7828 | 0.5541 | 0.6358 | 0.5661 | 0.5784 |
| RASAR | 0.6705 | 0.7874 | 0.5818 | 0.6488 | 0.5287 | 0.5930 |
| RASAR (condition) | 0.6705 | 0.7874 | 0.5818 | 0.6488 | 0.5287 | 0.5930 |
| RASAR + indirect | 0.7592 | 0.8455 | 0.5803 | 0.7097 | 0.6060 | 0.6122 |
| RASAR + indirect (condition) | 0.7592 | 0.8455 | 0.5803 | 0.7097 | 0.6060 | 0.6122 |
| Method | Ames ($n = 1457$) | | DILI ($n = 96$) | | Carcinogens ($n = 56$) | |
| | F1 | AUC | F1 | AUC | F1 | AUC |
| MiniMol (frozen) | 0.7546 | 0.8433 | 0.6628 | 0.9535 | 0.6879 | 0.7394 |
| MiniMol (fine-tuned) | 0.7744 | 0.8609 | 0.6403 | 0.9278 | 0.7148 | 0.7687 |
| MiniMol + indirect | 0.7585 | 0.8587 | 0.6403 | 0.9343 | 0.6690 | 0.7616 |
| MiniMol KNN | 0.7073 | 0.7668 | 0.8112 | 0.8883 | 0.7148 | 0.7616 |
| MiniMol KNN (condition) | 0.7073 | 0.7668 | 0.8112 | 0.8883 | 0.7148 | 0.7616 |
| Morgan KNN | 0.7173 | 0.7636 | 0.7292 | 0.8122 | 0.7352 | 0.7303 |
| Morgan KNN (condition) | 0.7173 | 0.7636 | 0.7292 | 0.8122 | 0.7352 | 0.7303 |
| RASAR | 0.7244 | 0.7802 | 0.5547 | 0.8339 | 0.6735 | 0.6717 |
| RASAR (condition) | 0.7244 | 0.7802 | 0.5547 | 0.8339 | 0.6735 | 0.6717 |
| RASAR + indirect | 0.7151 | 0.8038 | 0.7265 | 0.8713 | 0.7083 | 0.8303 |
| RASAR + indirect (condition) | 0.7151 | 0.8038 | 0.7265 | 0.8713 | 0.7083 | 0.8303 |

Table 7: TDC with TDC + Starling training labels: complete ML test results. Each task reports macro-F1 (F1) and AUROC (AUC). All methods use the settings in Appendix E.1.

| Method | BBB ($n = 406$) | | Oral ($n = 128$) | | Skin ($n = 82$) | |
|---|---|---|---|---|---|---|
| | F1 | AUC | F1 | AUC | F1 | AUC |
| MiniMol (frozen) | 0.7645 | 0.8876 | 0.6179 | 0.7313 | 0.6095 | 0.7362 |
| MiniMol (fine-tuned) | 0.7713 | 0.8753 | 0.6315 | 0.7359 | 0.6585 | 0.7502 |
| MiniMol + indirect | 0.7716 | 0.8791 | 0.5908 | 0.7167 | 0.6459 | 0.7532 |
| MiniMol KNN | 0.7452 | 0.8069 | 0.6234 | 0.6787 | 0.4325 | 0.4824 |
| MiniMol KNN (condition) | 0.7402 | 0.8072 | 0.6234 | 0.6787 | 0.4325 | 0.4824 |
| Morgan KNN | 0.7329 | 0.7805 | 0.6082 | 0.6841 | 0.6110 | 0.6286 |
| Morgan KNN (condition) | 0.7437 | 0.7864 | 0.6018 | 0.6801 | 0.6110 | 0.6286 |
| RASAR | 0.6780 | 0.7682 | 0.5821 | 0.6525 | 0.5285 | 0.5404 |
| RASAR (condition) | 0.7383 | 0.7821 | 0.5533 | 0.6425 | 0.5653 | 0.5489 |
| RASAR + indirect | 0.7527 | 0.8322 | 0.6807 | 0.7739 | 0.5851 | 0.5976 |
| RASAR + indirect (condition) | 0.7610 | 0.8214 | 0.5253 | 0.7137 | 0.5791 | 0.6055 |
| Method | Ames ($n = 1457$) | | DILI ($n = 96$) | | Carcinogens ($n = 56$) | |
| | F1 | AUC | F1 | AUC | F1 | AUC |
| MiniMol (frozen) | 0.7606 | 0.8525 | 0.5362 | 0.7157 | 0.6957 | 0.8475 |
| MiniMol (fine-tuned) | 0.7585 | 0.8613 | 0.6000 | 0.7635 | 0.7782 | 0.7929 |
| MiniMol + indirect | 0.7773 | 0.8616 | 0.4594 | 0.7248 | 0.7352 | 0.7848 |
| MiniMol KNN | 0.7052 | 0.7694 | 0.6222 | 0.7074 | 0.5575 | 0.7899 |
| MiniMol KNN (condition) | 0.7145 | 0.7736 | 0.6312 | 0.7239 | 0.7148 | 0.7616 |
| Morgan KNN | 0.7090 | 0.7608 | 0.6759 | 0.7041 | 0.5497 | 0.6970 |
| Morgan KNN (condition) | 0.7152 | 0.7682 | 0.6759 | 0.6930 | 0.7352 | 0.7303 |
| RASAR | 0.6943 | 0.7650 | 0.5577 | 0.6413 | 0.6510 | 0.7566 |
| RASAR (condition) | 0.7231 | 0.7679 | 0.5279 | 0.6400 | 0.6879 | 0.6293 |
| RASAR + indirect | 0.7065 | 0.8013 | 0.6825 | 0.7657 | 0.6879 | 0.8778 |
| RASAR + indirect (condition) | 0.7014 | 0.8024 | 0.6515 | 0.7874 | 0.7565 | 0.8202 |

Table 8: Mean macro-F1 of the baselines in Figure 3; the corresponding means for our system are given in Table 10. Each mean assigns equal weight to all six tasks. LLM results use a fixed backbone and evidence setting. For each ML group, we first take the maximum test macro-F1 among its candidate methods within each task and training pool, then average across tasks. These ML values are descriptive group-level upper bounds, rather than the performance of a single fixed or validation-selected method. Dashes denote inapplicable training-pool combinations.

| Baseline / setting | Literature-derived | TDC |
|---|---|---|
| *Naive LLM baselines* | | |
| DeepSeek-V4-Flash: None | 0.6066 | 0.6374 |
| DeepSeek-V4-Flash: Direct | 0.6379 | 0.6874 |
| DeepSeek-V4-Flash: Direct + Indirect | 0.6784 | 0.6916 |
| GPT-6 Luna: None | 0.6176 | 0.6589 |
| GPT-6 Luna: Direct | 0.6248 | 0.6516 |
| GPT-6 Luna: Direct + Indirect | 0.6543 | 0.6771 |
| Gemini 3.8 Flash: None | 0.6660 | 0.7434 |
| Gemini 3.8 Flash: Direct | 0.6792 | 0.7648 |
| Gemini 3.8 Flash: Direct + Indirect | 0.7125 | 0.7701 |
| *ML: each benchmark's main-task training pool* | | |
| MiniMol Direct | 0.6173 | 0.6789 |
| MiniMol + Indirect | 0.5923 | 0.6508 |
| KNN | 0.6025 | 0.7036 |
| RASAR | 0.5716 | 0.6223 |
| RASAR + Indirect | 0.5580 | 0.6826 |
| *ML: TDC + Starling training pool* | | |
| MiniMol Direct | – | 0.7000 |
| MiniMol + Indirect | – | 0.6634 |
| KNN | – | 0.6843 |
| RASAR | – | 0.6424 |
| RASAR + Indirect | – | 0.6954 |

## F Complete Naive LLM Baseline Results

Table 9 reports all 108 task-level macro-F1 scores for DeepSeek-V4-Flash, GPT-6 Luna, and Gemini 3.8 Flash in the main baseline comparison. Each model is evaluated on both benchmark collections with None, Direct, and Direct + Indirect evidence. None uses no retrieved records. Direct uses 10 direct molecules; Direct + Indirect uses the same direct budget plus 50 indirect records. Both evidence pools use Neighbor-fill: records are accumulated in decreasing molecular similarity order until the respective budget is filled. Each setting uses one full-flat inference with the matched query condition and cached single-molecule analysis.

The six-task mean is computed from unrounded task scores with equal task weights. The None scores are also the shared no-retrieval reference for the corresponding backbone in Appendix G.

Table 9: Complete test macro-F1 for the naive LLM baselines. DeepSeek: DeepSeek-V4-Flash; Luna: GPT-6 Luna; Gemini: Gemini 3.8 Flash. D: Neighbor-fill Direct 10 molecules; D+I: Neighbor-fill Direct 10 molecules + Indirect 50 records. Oral denotes oral bioavailability; Carc. denotes carcinogenicity. Scores and six-task means are rounded to four decimal places.

| Model | Evidence | BBB | Oral | Skin | Ames | DILI | Carc. | Mean |
|---|---|---|---|---|---|---|---|---|
| *Literature-derived benchmark* | | | | | | | | |
| DeepSeek | None | 0.5816 | 0.6303 | 0.6243 | 0.7066 | 0.6072 | 0.4894 | 0.6066 |
| | D | 0.6669 | 0.7466 | 0.5791 | 0.7212 | 0.5858 | 0.5278 | 0.6379 |
| | D+I | 0.7039 | 0.8014 | 0.6467 | 0.7288 | 0.6114 | 0.5783 | 0.6784 |
| Luna | None | 0.6262 | 0.6489 | 0.6308 | 0.6972 | 0.6200 | 0.4825 | 0.6176 |
| | D | 0.6444 | 0.7577 | 0.5836 | 0.6863 | 0.5979 | 0.4787 | 0.6248 |
| | D+I | 0.6878 | 0.7508 | 0.6193 | 0.7002 | 0.6177 | 0.5498 | 0.6543 |
| Gemini | None | 0.6707 | 0.7390 | 0.6488 | 0.7165 | 0.6472 | 0.5740 | 0.6660 |
| | D | 0.6925 | 0.7992 | 0.6539 | 0.7174 | 0.6240 | 0.5883 | 0.6792 |
| | D+I | 0.7406 | 0.8198 | 0.6456 | 0.7501 | 0.6928 | 0.6262 | 0.7125 |
| *TDC benchmark* | | | | | | | | |
| DeepSeek | None | 0.6633 | 0.6391 | 0.5490 | 0.6931 | 0.6298 | 0.6500 | 0.6374 |
| | D | 0.7726 | 0.7216 | 0.5589 | 0.7531 | 0.6403 | 0.6778 | 0.6874 |
| | D+I | 0.7643 | 0.7247 | 0.5832 | 0.7638 | 0.6515 | 0.6621 | 0.6916 |
| Luna | None | 0.7398 | 0.6146 | 0.5390 | 0.6979 | 0.6952 | 0.6667 | 0.6589 |
| | D | 0.7686 | 0.7027 | 0.4641 | 0.7563 | 0.5712 | 0.6468 | 0.6516 |
| | D+I | 0.7880 | 0.7292 | 0.5305 | 0.7627 | 0.6357 | 0.6166 | 0.6771 |
| Gemini | None | 0.8088 | 0.7524 | 0.6521 | 0.7562 | 0.8398 | 0.6510 | 0.7434 |
| | D | 0.8100 | 0.8217 | 0.6370 | 0.8014 | 0.7921 | 0.7268 | 0.7648 |
| | D+I | 0.8057 | 0.8510 | 0.6691 | 0.8106 | 0.7703 | 0.7143 | 0.7701 |

## G Complete Results of Our System

Table 10 reports the task-level scores underlying Figure 4, using DeepSeek-V4-Flash, GPT-6 Luna, and Gemini 3.8 Flash as reasoning backbones.

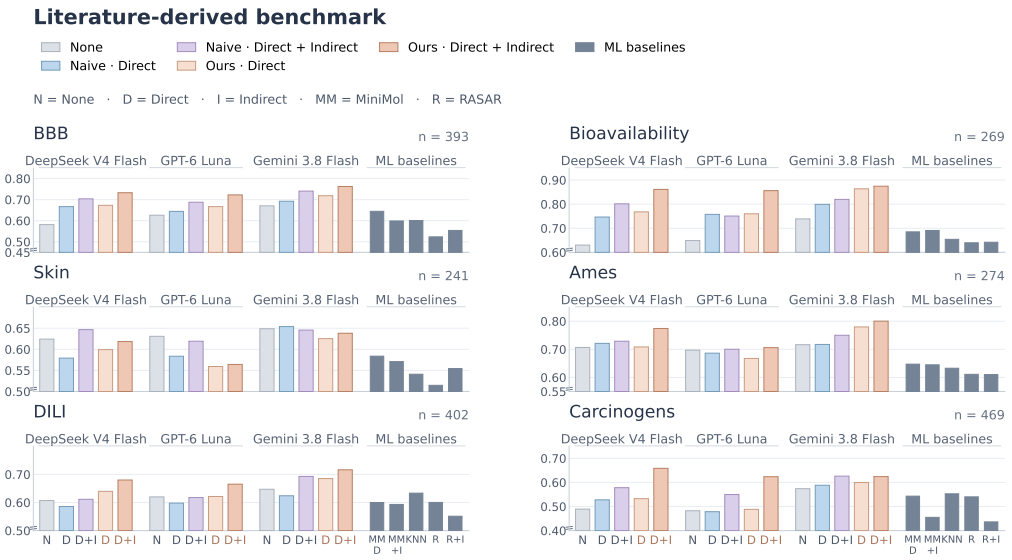


Figure 5: **Test macro-F1 on the literature-derived benchmark.** Six-task means are shown separately in Figure 3. Naive LLM baselines retrieve 10 direct molecule–condition pairs or the same 10 plus 50 indirect records in decreasing order of molecular similarity. Complete ML baseline variants and AUROC (if available) are reported in Appendix E. Complete scores for Ours are reported in Table 10.

Table 10: Test macro-F1 of our system on both benchmark collections. D: direct evidence; D+I: direct and indirect evidence. Means weight the six tasks equally. All scores are rounded to four decimal places.

| | DeepSeek-V4-Flash | | GPT-6 Luna | | Gemini 3.8 Flash | |
|---|---|---|---|---|---|---|
| Task | D | D+I | D | D+I | D | D+I |
| *Literature-derived benchmark* | | | | | | |
| BBB | 0.6732 | 0.7330 | 0.6669 | 0.7226 | 0.7186 | 0.7626 |
| Oral bioavailability | 0.7681 | 0.8613 | 0.7601 | 0.8561 | 0.8638 | 0.8745 |
| Skin | 0.5993 | 0.6185 | 0.5593 | 0.5642 | 0.6254 | 0.6382 |
| Ames | 0.7086 | 0.7742 | 0.6677 | 0.7060 | 0.7797 | 0.8003 |
| DILI | 0.6398 | 0.6801 | 0.6215 | 0.6653 | 0.6853 | 0.7164 |
| Carcinogens | 0.5327 | 0.6586 | 0.4883 | 0.6238 | 0.5996 | 0.6242 |
| Mean | 0.6536 | 0.7209 | 0.6273 | 0.6897 | 0.7121 | 0.7360 |
| *TDC benchmark* | | | | | | |
| BBB | 0.8657 | 0.8808 | 0.8288 | 0.8463 | 0.8274 | 0.8743 |
| Oral bioavailability | 0.7467 | 0.8307 | 0.7184 | 0.7740 | 0.8171 | 0.8476 |
| Skin | 0.6341 | 0.6669 | 0.7009 | 0.6110 | 0.6781 | 0.6957 |
| Ames | 0.7823 | 0.8079 | 0.7879 | 0.8113 | 0.8198 | 0.8261 |
| DILI | 0.8196 | 0.8213 | 0.7909 | 0.8015 | 0.8748 | 0.8748 |
| Carcinogens | 0.8174 | 0.7906 | 0.8415 | 0.8783 | 0.8303 | 0.7737 |
| Mean | 0.7776 | 0.7997 | 0.7781 | 0.7871 | 0.8079 | 0.8154 |