MOF-SELECT: BENCHMARKING SELECTION AND CONFIDENCE IN CONDITIONAL MOF GENERATION

Anonymous authors
Paper under double-blind review

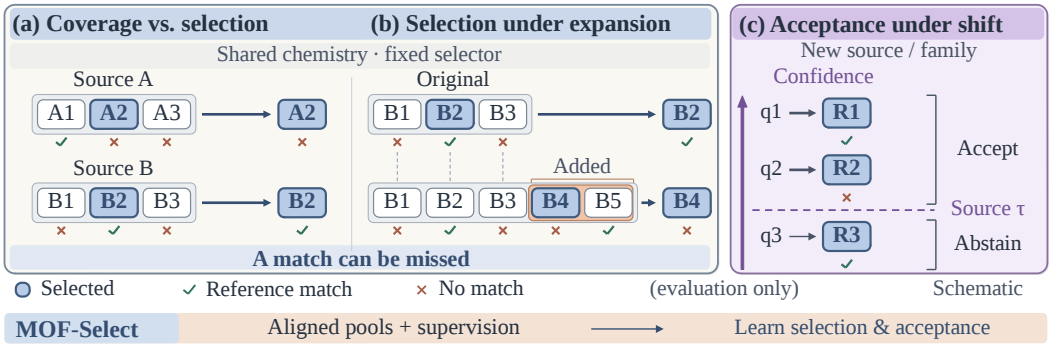

Figure 1: **From candidate coverage to reliable MOF recommendations.** (a) With shared chemistry and a fixed selector, pools containing matches can yield different recovery outcomes. (b) Pool expansion can add matches yet displace a correct choice. (c) After a source or family shift, the unchanged source-calibrated threshold τ can accept nonmatches and abstain on matches. Confidence ranks recommendations R_i for conditions q_i. Blue fill marks selections; checks/crosses denote evaluation-only reference matching in these schematic examples, not physical validity.

ABSTRACT

Conditional generative models predict metal–organic framework (MOF) structures from specified building blocks. Their value for materials design depends on converting candidate pools into reliable recommendations, a transition that candidate coverage alone cannot assess. We introduce MOF-Select, a dataset and benchmark that makes this transition a measurable learning problem. MOF-Select aligns outputs from different generators under shared chemical conditions and provides candidate-level reference-recovery supervision. Controlled protocols separate available coverage from successful selection and acceptance under changes in generator source, chemistry, and sampling budget. The resource comprises 245,760 generation attempts and supports comparisons between heuristic and learned decision rules. Across three generators, the best selector in each generator/test setting leaves 7.85–24.74 percentage points of reference recovery unused at ten attempts per condition. Learning improves selection and confidence ranking on MOF-LLM pools, while larger pools can displace correct choices. Better confidence ranking does not ensure certified acceptance, and source-calibrated rules can incur high acceptance error after a generator change. MOF-Select establishes a common framework for evaluating and improving how generative coverage translates into reliable recommendations, making downstream decision quality an explicit target for progress in MOF generation.

1 INTRODUCTION

Metal–organic frameworks (MOFs) combine metal nodes and organic linkers into periodic structures, offering a large design space for computational materials discovery (Fu et al., 2024). Conditional generative models predict these structures from specified building blocks (Kim et al., 2025a; 2026a; Pan et al., 2026). Benchmarks such as MGB enable comparisons across materials generation

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.

tasks and distribution shifts (Yan et al., 2025). Using these predictions for design requires choosing which generated structure to recommend and deciding when to accept that recommendation.

Candidate coverage measures whether a pool contains a match to a known reference, while selection must choose a structure without access to that reference. Even for the same chemistry, a fixed selector can succeed on one generator's pool and fail on another when both contain a match (Figure 1a). Adding candidates can expand coverage while displacing a correct choice (Figure 1b). Acceptance further depends on confidence that remains reliable after a change of generator or chemical family (Figure 1c).

These distinctions determine how progress in generation should be assessed. Improved recovery among recommended structures can come from greater candidate coverage or more effective selection. Accepting more recommendations may increase correct yield while admitting more errors. Joint evaluation reveals which gains translate into reliable use of a generator.

Reference-supervised candidate scoring is established in crystal prediction and molecular docking (Jiao et al., 2023; Corso et al., 2023). Across MOF generators, the evaluation challenge is to separate decision quality from differences in the candidate supply. Pairing generators on shared chemical conditions isolates changes in source from changes in chemistry; comparing decision rules on fixed candidate pools isolates their ability to use those candidates. Separate data roles keep training and calibration apart from final evaluation.

Shared candidate-level supervision lets researchers train models on matching and nonmatching candidates and test whether the learned distinctions transfer to another generator. Holding the label budget fixed separates the effect of changing training sources from the benefit of additional supervision. Reusing generated structures also supports comparisons of learning objectives and model representations without confounding them with new samples.

We introduce **MOF-Select**, a dataset and benchmark that makes the conversion from candidate coverage to accurate, selectively accepted MOF recommendations a reproducible learning problem. Its aligned supervision and controlled protocols provide a common testbed for improving decision quality across generators. Our contributions are:

Aligned data. We align outputs from three generators on shared building-block conditions from the BW hypothetical-MOF collection (Boyd et al., 2019), yielding 245,760 attempts across 7,168 conditions. Labeled candidate pools and explicit failure records support reusable training and evaluation of decision rules without rerunning generation.

Controlled evaluation. Our protocols distinguish candidate coverage, selected recovery, and accepted yield across generator, chemistry, and budget changes. Controls isolate training-source effects at a fixed label budget and confidence quality at fixed recommendations.

Learning gains and gaps. The best selector in each generator/test setting leaves 7.85–24.74 percentage points of recovery unused at ten attempts per condition. Learning improves selection and confidence ranking on MOF-LLM, including a 3.11-point selection gain over structural consensus on 2,048 additional conditions. Pool expansion and generator transfer expose distinct challenges in preserving correct choices and reliable acceptance.

2 Related Work

MOF generation. MOF generators address both framework design and assembly from specified building blocks. MOFDiff uses coarse-grained diffusion and building-block assembly (Fu et al., 2024). PRO-MOF optimizes block proposal and assembly policies for property targets (Liu et al., 2026). For conditional structure prediction, MOFFlow models rigid-body assembly, while MOFFlow-2 incorporates torsional flexibility (Kim et al., 2025b;a). AtomMOF uses all-atom flow matching, and MOF-LLM predicts assembly transformations with a language model (Kim et al., 2026a; Pan et al., 2026). Our benchmark uses frozen outputs from AtomMOF, MOFFlow-2, and MOF-LLM to study downstream decisions under shared chemical conditions.

Materials generation benchmarks. MGB evaluates material generation across tasks, including MOF structure prediction and out-of-distribution settings (Yan et al., 2025). The work introducing METRe examines structural similarity, dataset redundancy, and polymorph-aware splits (Mar-

tirossyan et al., 2025). MADE evaluates adaptive materials discovery under limited oracle queries (Malik et al., 2026). These benchmarks and evaluation tools address generation quality, structural comparison, and discovery efficiency. More broadly, WILDS shows the value of explicit domain splits and standardized learning interfaces for distribution-shift evaluation (Koh et al., 2021). MOF-Select holds chemical requests fixed across generators and supplies candidate supervision to evaluate decisions within their output pools.

Candidate ranking and confidence. DiffCSP studies candidate scorers, including a reference-match classifier (Jiao et al., 2023). DiffDock learns confidence scores to rank docking poses and evaluates selective accuracy (Corso et al., 2023). Packora separates generation and ranking evaluation for molecular crystals, with a common relaxation and ranking pipeline (Kim et al., 2026b). These studies establish ranking as a component of generative structure prediction. Selective classification studies the risk–coverage trade-off (Geifman & El-Yaniv, 2017); Learn then Test supports finite-sample calibration of decision rules (Angelopoulos et al., 2021). Uncertainty benchmarks also show that calibration can deteriorate under dataset shift (Ovadia et al., 2019). MOF-Select connects these established ideas through condition-aligned pools, distinct learning roles, and controlled changes in source, family, and budget. This design compares learned decision rules and their supervision while keeping the generated candidates fixed.

3 MOF-Select Dataset

MOF-Select pairs chemical conditions with generator-specific candidate pools. Figure 2 connects this learning unit to selection and acceptance. Table 1 separates learning roles, repeated generation, and confirmation.

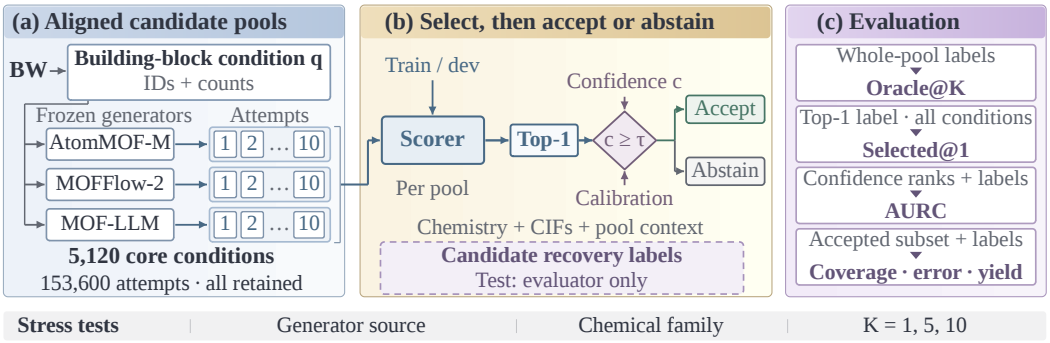

Figure 2: **MOF-Select data and decision protocol.** (a) Shared chemistry aligns three separate generator pools; numbers index attempts. (b) Scoring selects one candidate per pool; independent calibration sets confidence threshold τ. (c) Oracle@K measures pool coverage; Selected@1 measures recovery over all conditions. AURC evaluates confidence ordering. Accepted recommendations determine coverage; their labels give error and correct yield. Test labels enter only evaluation. Panel (a) shows the core; Table 1 includes all extensions.

3.1 Chemical conditions and aligned candidate pools

Conditions and references come from the BW hypothetical-MOF collection (Boyd et al., 2019), through AtomMOF's official test inventory. A condition specifies building-block chemistry and multiplicities. All references for a condition are grouped before assigning data roles.

Each of the 5,120 core conditions has ten attempts from each frozen generator: AtomMOF-M, MOFFlow-2, and MOF-LLM. Separate pools for the same chemical request enable paired comparisons across sources. Mixed-label pools provide direct ranking supervision: the 1,024-condition core test contains 533 such AtomMOF-M pools, 393 MOFFlow-2 pools, and 202 MOF-LLM pools. All-hit and all-miss pools remain in evaluation, preserving conditions with no reference match to select.

Table 1: **Data composition and learning roles.** Each stream uses ten attempts per condition and generator. Repeated streams reuse existing conditions; the total counts each chemical condition once.

Resource / role	Conditions	Attempts	Use
Core: training	2,048	61,440	Fit decision rules
Development	512	15,360	Choose model settings
Calibration	512	15,360	Set acceptance thresholds
Core test	1,024	30,720	Evaluate source population
Family adaptation	512	15,360	Supply target-family labels
Family test	512	15,360	Evaluate held-out skeletons
Two repeated streams	512 reused	30,720	Test resampling sensitivity
Additional conditions	2,048	61,440	Confirm fixed comparisons
Total	7,168 distinct	245,760	

3.2 Candidate supervision and data quality

Attempt records link condition IDs, generators, draw indices, status, and recovery labels. Model inputs include building-block graphs, candidate CIFs and periodic graphs, within-pool structural relations, and optional energy and force descriptors. Reference coordinates, target cell and topology, recovery labels, and reference-matching time are excluded from scoring inputs.

A positive label requires a match to any known same-condition reference. The core contains 5,161 reference files, with multiple references for 36 conditions. Labels use pymatgen's RMS-based periodic matcher (Ong et al., 2013) with cell-volume scaling, normalized site tolerance 0.5, lattice tolerance 0.3, and angle tolerance 10°. The strict label reduces site tolerance to 0.3. Recovery measures reference agreement; physical stability and synthesis are separate targets. A fixed 1,920-candidate audit checks geometric error, short contacts, and scaling sensitivity (Appendix F). Exact matching calls and finer tolerance checks appear in Appendix C.

3.3 Data splits and evaluation extensions

Chemical-condition groups do not cross the six core roles in Table 1. The family test spans 23 linker skeleton groups absent from training, development, calibration, and the core test. Skeletons use the generic heavy-atom graph 2-core, with the full heavy-atom graph as the acyclic fallback. Separate adaptation conditions from held-out families support comparisons with and without target-family supervision. Generator-training exposure is annotated separately from these decision-learning splits.

Two further generation streams reuse 256 core-test and 256 family-test conditions to test resampling at fixed chemistry. The additional 2,048 task-new conditions evaluate fixed models and comparisons. Repeated streams add no distinct conditions. Appendix A details grouping and confirmation history.

3.4 Learning interfaces and reusable assets

Role-aware readers separate inputs, learning supervision, and evaluation labels. Methods score a fixed candidate prefix and supply recommendation confidence. The evaluator joins labels after scoring. Pool relations use the requested prefix, and failure records preserve attempt counts.

Researchers can replace the scorer, pool representation, training-source allocation, or confidence function while reusing the generated structures. Tabular and structure-based interfaces, model configurations, and evaluation runtimes support these experiments without rerunning the generators. Section 4 defines the evaluation protocol.

4 Benchmark Tasks and Evaluation Protocol

The two tasks share candidate pools and use separate evaluation paths (Figure 2).

4.1 CANDIDATE SELECTION UNDER SAMPLING BUDGETS

For a fixed generator g and budget K, each condition q has an ordered pool $\mathcal{X}_{q,g}^{K} = (x_{q,1}, \ldots, x_{q,K})$ with recovery labels $y_{q,i} \in \{0, 1\}$. A selector uses the permitted inputs to choose index a_q, with a fixed label-independent tie rule. Write $s_q = y_{q,a_q}$, setting $s_q = 0$ when no candidate can be scored. Over all N evaluated conditions, we report

$$S_K = \frac{1}{N} \sum_q s_q, \qquad O_K = \frac{1}{N} \sum_q \max_{i \le K} y_{q,i}, \qquad R_K = \frac{1}{NK} \sum_q \sum_{i=1}^{K} y_{q,i}. \tag{1}$$

These are Selected@1, Oracle@K, and Random@1: delivered recovery, available reference coverage, and the exact expectation of a uniform attempted draw. The gap $O_K - S_K$ measures covered conditions whose correct candidate was missed.

Budgets $K \in \{1, 5, 10\}$ use fixed prefixes, with pool features recomputed at each K. Generation and parsing failures have label zero. All-failed pools remain in N. Equal attempt counts compare sampling budgets, not runtime; measured costs appear in Appendix G. For a fixed selector, the change in Selected@1 decomposes into new hits plus rescues minus lost correct recommendations (Appendix C). The main learned-model budget comparison evaluates the same $K = 10$-trained selectors at $K = 5$ and $K = 10$.

4.2 CONFIDENCE ESTIMATION AND SELECTIVE ACCEPTANCE

Confidence ranks recommendations across conditions, while selection ranks candidates within a pool. Acceptance follows the selective-prediction setting (Geifman & El-Yaniv, 2017). We evaluate both confidence functions for fixed Consensus recommendations and the selected scores of frozen learned selectors. For confidence c_q, threshold τ accepts $A_q(\tau) = \mathbf{1}[c_q \ge \tau]$. Coverage, accepted error rate, and correct yield are

$$C(\tau) = \frac{1}{N} \sum_q A_q(\tau), \quad \rho(\tau) = \frac{\sum_q A_q(\tau)(1 - s_q)}{\sum_q A_q(\tau)}, \quad J(\tau) = \frac{1}{N} \sum_q A_q(\tau) s_q. \tag{2}$$

Correct yield counts successful accepted recommendations per evaluated condition. Zero acceptance gives undefined risk and zero yield. Area under the risk–coverage curve (AURC; lower is better) measures confidence ordering, using expected random order within score ties.

We calibrate thresholds on 512 independent conditions for risk targets $\alpha \in \{0.05, 0.10, 0.20\}$ and failure probability $\delta = 0.05$. Threshold candidates come from a fixed probability grid, development-score quantiles, or one development-selected threshold. Raw ranking scores use development quantiles. We correct one-sided Clopper–Pearson upper bounds for threshold search within each scoring function. Among thresholds whose bound is at most α, we choose the one with greatest calibration coverage, otherwise abstaining on all conditions. This follows held-out risk control (Angelopoulos et al., 2021), which assumes a fixed scoring function and independent calibration and future examples from the same distribution. Generator and family shifts are evaluated empirically with the unchanged threshold. Appendix C specifies the calibration rules.

4.3 BASELINES AND TRAINING CONTROLS

First and Random measure selection without candidate discrimination. Consensus uses within-pool structural agreement. Physical heuristics rank UMA energy or force descriptors (Wood et al., 2025). Histogram gradient boosting (HGB) learns from candidate and pool scalars. A LightGBM classifier and LambdaMART ranker (Ke et al., 2017; Burges, 2010) share scalar and chemical features, isolating the learning objective under matched search budgets.

SchNet-pair uses a periodic SchNet encoder (Schütt et al., 2017), trained with binary classification and within-pool pairwise ranking. Full combines this structure representation with chemistry, physical descriptors, and pool mean and maximum embeddings. Attention-Full adds candidate self-attention. These models and feature ablations test which inputs and interactions improve choice. Full provides the complete input set and source-training variants for the main budget and transfer comparisons; it is a common reference across these analyses. An auxiliary chemical–structure contrastive control uses training reference structures, with this additional supervision marked separately.

Main learned selectors use 2,048 training conditions and three seeds. Neural selectors receive 5,120 updates, and each selector tree family searches six configurations. Model selection uses development data. Full and both LightGBM families compare source specialists, balanced three-source training, and leave-one-source-out training at 2,048 pools and 20,480 labels. All-source training uses three separate pools per condition: 6,144 pools and 61,440 labels. Source assignment is label-independent, and generator identity is excluded from features. Development and calibration each retain one pool per condition, with identical assignments for balanced and all-source training. Leave-one-source-out excludes the named generator from all three learning roles. Appendix B details architectures, confidence functions, and baseline coverage.

4.4 Statistical evaluation

Selection comparisons average the three fixed training seeds within condition before paired bootstrap resampling. Repeated generation streams remain grouped by condition. We use 10,000 replicates and pointwise 95% intervals for exploratory comparisons. Family sensitivity resamples the 23 skeleton groups with a condition-weighted mean. The two locked confirmation contrasts use nominal 97.5% intervals each. Acceptance results retain each seed's calibrated rule.

Main tables use corrected physical inputs and refitted models. The original frozen confirmation and corrected-input sensitivity on the same conditions remain distinct. Candidate CIFs and recovery labels are unchanged (Appendix B.4).

5 Results and Analysis

Shared conditions and fixed pools isolate the effects of decision learning, pool expansion, and supervision. We then test whether confidence supports reliable acceptance. Recovery gains are reported in percentage points (pp).

5.1 Selection gains depend on the generator

Selection learns useful distinctions within generated pools. Table 2 compares nine selectors on the same candidates. On MOF-LLM families, Full and SchNet-pair reach 40.17% and 41.28% recovery, compared with 36.33% for Consensus and 39.06% for the lowest-energy rule. The leading point estimate varies by generator and test role; Consensus leads both MOFFlow-2 tests. Even the highest-recovery selector in each column leaves 7.85–24.74 pp of Oracle coverage unused.

Table 2: **Selection performance by source and test role.** Selected@1 (%), primary label, $K = 10$; core $n = 1{,}024$, family $n = 512$. Learned specialists: three-seed means. Oracle gives coverage; bold marks the largest selected-recovery point estimate per column.

	AtomMOF-M		MOFFlow-2		MOF-LLM	
Selector	Core	Family	Core	Family	Core	Family
Oracle@10	68.75	65.82	51.86	50.20	52.83	51.56
Random	39.07	35.37	30.08	27.05	41.97	36.60
Consensus	46.88	39.06	**34.38**	**33.59**	43.16	36.33
Lowest energy/atom	41.89	38.09	31.25	31.25	43.55	39.06
HGB	46.97	38.87	34.28	32.81	42.68	38.48
LightGBM	**47.66**	39.65	33.20	31.25	42.68	38.09
LambdaMART	44.24	37.30	32.71	29.49	43.75	39.65
SchNet-pair	43.68	37.83	32.42	29.56	43.52	**41.28**
Full	46.65	39.65	33.20	32.62	44.53	40.17
Attention-Full	46.09	**41.08**	32.68	29.95	**44.99**	39.26

Task-new conditions confirm a source-dependent benefit. Before generating pools for 2,048 additional conditions, we fixed two contrasts: Full minus Consensus on MOF-LLM, and its difference from the MOFFlow-2 gain (Table 3). Both have positive intervals in the frozen comparison and the corrected-input analysis of the same conditions (Appendix A). Corrected gains are 3.11 pp on

MOF-LLM and 0.65 pp on MOFFlow-2, a difference of 2.46 pp. The source effect concerns the size of the learning benefit.

Table 3: **Confirmation supports a source-dependent learning gain.** Full minus Consensus on 2,048 task-new conditions (pp), with nominal 97.5% intervals for the two fixed contrasts. Corrected sensitivity uses the same conditions.

Contrast	Frozen	Corrected sensitivity
MOF-LLM gain	**+3.34** [1.95, 4.72]	**+3.11** [1.76, 4.48]
MOF-LLM minus MOFFlow-2 gain	**+3.52** [1.66, 5.39]	**+2.46** [0.75, 4.17]

5.2 Larger pools can displace correct choices

Table 4 compares the same Full selector at $K = 5$ and $K = 10$. On MOF-LLM core conditions, coverage grows by 2.73 pp, but selected recovery gains 0.68 pp. New hits and rescues contribute 1.27 and 0.88 pp; losing previously correct recommendations offsets 1.46 pp. On AtomMOF-M families, coverage grows by 8.79 pp while Selected@1 changes by -0.13 pp (95% interval $[-2.67, 2.47]$).

Table 4: **Additional candidates create gains and losses.** Fixed Full, $K = 5 \rightarrow 10$, primary label; three-seed mean changes (pp). Bold net gain equals new hit plus rescue minus loss, up to rounding.

Generator	Role	ΔOracle	New hit	Rescue	Loss	ΔSelected
AtomMOF-M	Core	6.15	1.24	5.53	4.33	**+2.44**
AtomMOF-M	Family	8.79	1.50	4.23	5.86	**-0.13**
MOFFlow-2	Core	6.45	1.27	2.83	3.71	**+0.39**
MOFFlow-2	Family	6.05	1.76	3.78	3.26	**+2.28**
MOF-LLM	Core	2.73	1.27	0.88	1.46	**+0.68**
MOF-LLM	Family	4.30	1.63	1.43	1.82	**+1.24**

These losses identify a target for pool-aware selectors: exploit new candidates while preserving correct choices. The comparison measures prefix transfer of fixed $K = 10$-trained models.

5.3 Effects of model design and supervision

Useful inputs and interactions depend on the candidate source. Two-source input ablations isolate a physical-feature gain of 2.21 pp [0.39, 4.10] on MOFFlow-2 families; the other three source/role intervals cross zero (Appendix D). Attention-Full changes recovery relative to Full by $+1.43$ pp [0.33, 2.60] on AtomMOF-M families and -2.67 pp $[-4.10, -1.30]$ on MOFFlow-2 families; the other four intervals cross zero. With matched input features, LambdaMART has no consistent advantage over classification (Appendix D). These comparisons isolate candidate information, model interaction, and learning objective.

Additional supervision improves the smaller-budget evaluation. At $K = 10$, balanced multi-source training has no interval-supported gain over specialists at equal label count. Tripling source labels likewise gives no supported gain over balanced training. At $K = 5$, the same Full models show a benefit from additional source supervision (Table 5): core recovery improves by 0.90 pp [0.37, 1.42] and family recovery by 1.28 pp [0.46, 2.13]. Some two-source models also improve transfer to a held-out generator (Appendix D). The controls separate training-source allocation from additional supervision and evaluate each across candidate budgets.

5.4 Confidence ranking and acceptance reliability

Confidence can improve independently of candidate choice. Table 6 fixes Consensus recommendations and changes only confidence. Condition–pool learning lowers MOF-LLM AURC by 0.0515 on core and 0.0345 on family conditions, both with positive paired improvement intervals. It worsens AtomMOF-M family ordering; the other three intervals cross zero.

Table 5: **Extra source supervision helps at the smaller budget.** Full macro gains across three sources (pp), with pointwise 95% paired intervals. Training labels: specialist/balanced 20,480; all-source 61,440. Both budgets use the same $K = 10$-trained models. Teal/clay marks positive/negative intervals excluding zero.

K	Comparison	Role	Gain	95% interval
10	Balanced minus specialist	Core	**-0.82**	[-1.50, -0.16]
10	Balanced minus specialist	Family	-0.41	[-1.50, 0.65]
10	All-source minus balanced	Core	+0.23	[-0.37, 0.84]
10	All-source minus balanced	Family	+0.67	[-0.26, 1.61]
5	Balanced minus specialist	Core	**-0.63**	[-1.27, -0.02]
5	Balanced minus specialist	Family	-0.69	[-1.63, 0.24]
5	All-source minus balanced	Core	**+0.90**	[0.37, 1.42]
5	All-source minus balanced	Family	**+1.28**	[0.46, 2.13]

Table 6: **Confidence learning at fixed Consensus recommendations.** AURC ($[0, 1]$, lower is better), primary label, $K = 10$. Joint: three-seed condition–pool mean. Improvement: ECDF minus Joint, with 95% paired condition-bootstrap intervals; teal/clay marks intervals above/below zero.

Generator	Role	ECDF	Joint	Improvement	95% interval
AtomMOF-M	Core	0.2257	0.2331	-0.0074	[-0.0181, 0.0013]
AtomMOF-M	Family	0.3047	0.3197	**-0.0150**	[-0.0287, -0.0024]
MOFFlow-2	Core	0.3389	0.3401	-0.0012	[-0.0081, 0.0068]
MOFFlow-2	Family	0.3630	0.3676	-0.0046	[-0.0137, 0.0040]
MOF-LLM	Core	0.3409	0.2894	**+0.0515**	[0.0347, 0.0694]
MOF-LLM	Family	0.4137	0.3792	**+0.0345**	[0.0086, 0.0619]

Confidence ordering and risk certification test different properties. A seven-function diagnostic uses a 10% risk target, 512 calibration conditions, and one development-selected threshold per function. Four AtomMOF-M functions certify. The remaining three have empirical error below the target but insufficient evidence to bound risk at 10%. All seven MOFFlow-2 functions face this finite-sample obstacle. All seven MOF-LLM functions instead exceed the empirical target at the evaluated threshold. This separates empirical tail error from uncertainty in its estimate.

Certification also yields useful recommendations. The three AtomMOF-M Full seeds achieve 18.65–19.34% correct yield on the core test, with accepted error of 1.98–4.41%, below the 10% target. Appendix E reports coverage and counts for every certified function in this diagnostic.

Generator transfer can raise both coverage and accepted error. Figure 3 evaluates Full trained and calibrated without MOF-LLM.

Seed 1 certifies with 95 accepted calibration conditions and a 7.56% risk upper bound. On core tests, its unchanged threshold accepts 14.65–21.78% of conditions from the two source generators, with 2.69–6.67% accepted error. Held-out MOF-LLM has 52.05% coverage, 32.65% error, and 35.06% correct yield. Its family tests similarly give 46.68% coverage and 36.82% error. Broad acceptance thus coexists with error far above the 10% source target. Seeds 2–3 fail source certification and accept nothing on any generator, giving zero yield and undefined risk.

5.5 Robustness and label sensitivity

The MOF-LLM learning signal persists when generation is repeated at fixed chemistry. Averaging two new streams within each of 256 conditions per role, Full gains 4.30 pp [1.43, 7.23] on core and 5.01 pp [1.82, 8.27] on family tests. On the original family test, its 3.84-point gain retains a positive interval [1.43, 5.98] when resampling 23 skeleton groups. Excluding the six test conditions incident to reference-neighbor matches leaves core and family gains nearly unchanged. Gains also remain positive after excluding pools containing byte-identical candidate repeats.

Under strict matching, Full's MOF-LLM family gain over Consensus is 1.30 pp $[-0.88, 3.29]$ with family-group resampling. The energy rule leads both MOF-LLM tests in point estimates (Appendix D).

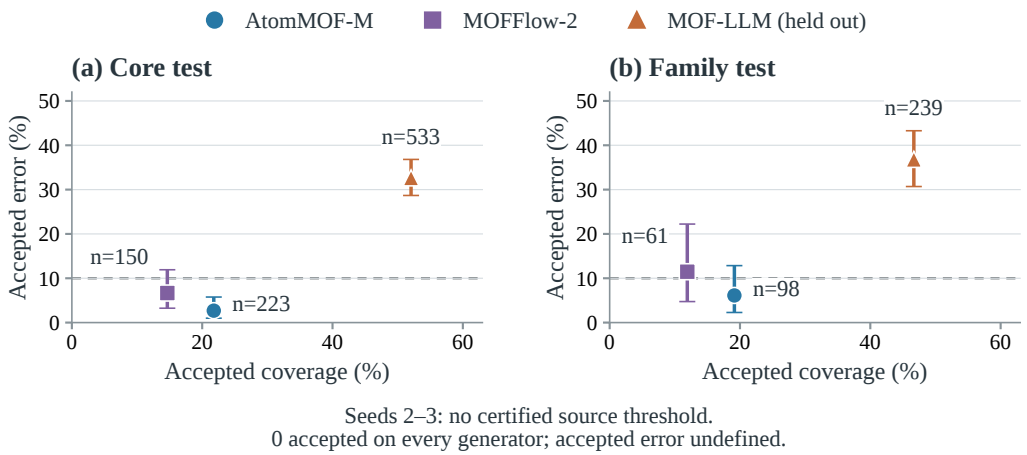

Figure 3: **Source-calibrated acceptance under generator transfer.** Full trained and calibrated on AtomMOF-M and MOFFlow-2; primary label, $K = 10$. All points use the same seed-1 threshold; n counts accepted conditions. Bars: descriptive 95% binomial intervals; dashed line: 10% source risk target. Seeds 2–3 have no certified threshold and abstain everywhere.

6 Discussion and Limitations

Learning to use generative coverage. MOF-Select provides reusable supervision for improving decisions after generation. The expansion analysis identifies a concrete objective: retain correct choices while exploiting new candidates. Joint selection and confidence models can target both recovery and accepted yield. Fixed pools support comparisons of representations and objectives; source-allocation controls and family adaptation roles distinguish method changes from added supervision. Separate calibration then tests whether confidence supports reliable acceptance. These capabilities let researchers develop decision models on the same generated structures, with artifact use and measured costs documented in Appendix G.

What reference recovery measures. Known-reference matching supplies consistent supervision. An unrecorded plausible arrangement can be a nonmatch, while RMS agreement can coexist with local defects. Primary and strict labels and the structural audit expose these differences. Physical stability and synthesizability require energetic or experimental evidence. Connecting these criteria to decision evaluation extends the task toward materials discovery.

Transfer across chemistry and candidate distributions. The resource aligns three generators within one hypothetical-MOF population. Experimental structures and new generator versions would test broader transfer. The family split measures decision-learning generalization under the stated skeleton grouping; generator-training exposure is separately annotated and incompletely documented. Calibration certifies a rule under its source-distribution assumptions, while target-generator and family acceptance are empirical outcomes.

7 Conclusion

MOF-Select establishes a common testbed for learning to turn generated MOF pools into accurate, selectively accepted recommendations. Aligned conditions and fixed candidate pools separate decision quality from candidate supply. Learning improves selection and confidence ranking on MOF-LLM pools, while the leading selector varies across generators. Pool expansion can displace correct choices, and source-calibrated acceptance rules can incur high error after generator transfer. These findings show why coverage, selected recovery, and accepted yield must be evaluated together: gains in candidate supply need not translate into reliable recommendations. By making these stages separately measurable on reusable candidates, MOF-Select provides a foundation for improving the decision models that connect conditional generation to materials screening.

AI USE STATEMENT

We used generative AI to assist with manuscript writing and language polishing, including improving the clarity and readability of the text. We also used it to support literature retrieval and discovery, helping identify relevant studies and related work.

REPRODUCIBILITY STATEMENT

Sections 3–4 specify the data roles, allowed inputs, labels, baselines, and evaluation protocol. Appendices A–G provide construction, training, calibration, and artifact details. A checksum-verified review supplement combines the complete tabular dataset, role manifests, training, scoring, and evaluation runtimes, and a saved model example. Structure payloads and full checkpoints are indexed separately. Appendix G records package-consumption checks and distribution status.

ETHICS STATEMENT

The benchmark uses existing computational materials resources and evaluates reference recovery. Its scores concern structural agreement and do not certify synthesis or application safety. Source attribution and data redistribution terms are tracked separately from software licenses. Access arrangements for upstream assets must respect their original distribution terms.

REFERENCES

Anastasios N. Angelopoulos, Stephen Bates, Emmanuel J. Candès, Michael I. Jordan, and Lihua Lei. Learn then test: Calibrating predictive algorithms to achieve risk control, 2021. URL https://arxiv.org/abs/2110.01052. arXiv preprint arXiv:2110.01052.

Peter G. Boyd, Arunraj Chidambaram, Enrique García-Díez, Christopher P. Ireland, Thomas D. Daff, Richard Bounds, Andrzej Gładysiak, Pascal Schouwink, Seyed Mohamad Moosavi, M. Mercedes Maroto-Valer, Jeffrey A. Reimer, Jorge A. R. Navarro, Tom K. Woo, Susana Garcia, Kyriakos C. Stylianou, and Berend Smit. Data-driven design of metal–organic frameworks for wet flue gas CO_2 capture. *Nature*, 576(7786):253–256, 2019. doi: 10.1038/s41586-019-1798-7.

Christopher J. C. Burges. From RankNet to LambdaRank to LambdaMART: An overview. Technical Report MSR-TR-2010-82, Microsoft Research, 2010. URL https://www.microsoft.com/en-us/research/publication/from-ranknet-to-lambdarank-to-lambdamart-an-overview/.

Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. DiffDock: Diffusion steps, twists, and turns for molecular docking. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=kKF8_K-mBbS.

Xiang Fu, Tian Xie, Andrew S. Rosen, Tommi Jaakkola, and Jake Smith. MOFDiff: Coarse-grained diffusion for metal-organic framework design. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/47547ee84e3fbbcbbbbad7c1fd9a973b-Abstract-Conference.html.

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

Rui Jiao, Wenbing Huang, Peijia Lin, Jiaqi Han, Pin Chen, Yutong Lu, and Yang Liu. Crystal structure prediction by joint equivariant diffusion. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-0767.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural*

Information Processing Systems, volume 30, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html.

Nayoung Kim, Seongsu Kim, and Sungsoo Ahn. Flexible MOF generation with torsion-aware flow matching. In *Advances in Neural Information Processing Systems*, volume 38, 2025a. doi: 10.52202/085713-0464.

Nayoung Kim, Seongsu Kim, Minsu Kim, Jinkyoo Park, and Sungsoo Ahn. MOFFlow: Flow matching for structure prediction of metal-organic frameworks. In *International Conference on Learning Representations*, 2025b. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/f392c6bbb14548df50092f10c9db440f-Abstract-Conference.html.

Nayoung Kim, Honghui Kim, Sihyun Yu, Minkyu Kim, Seongsu Kim, and Sungsoo Ahn. AtomMOF: All-atom flow matching for MOF-adsorbate structure prediction, 2026a. URL https://arxiv.org/abs/2602.07351. arXiv preprint arXiv:2602.07351.

Nayoung Kim, Kiyoung Seong, and Sungsoo Ahn. Packora: Systematic design for generative molecular crystal structure prediction, 2026b. URL https://arxiv.org/abs/2608.26962. arXiv preprint arXiv:2608.26962.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton Earnshaw, Imran Haque, Sara M Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. WILDS: A benchmark of in-the-wild distribution shifts. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5637–5664. PMLR, 2021. URL https://proceedings.mlr.press/v139/koh21a.html.

Zicheng Liu, Ben Fei, and Di Huang. PRO-MOF: Policy optimization with universal atomistic models for controllable MOF generation. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/f7f47a73d631c0410cbc2748a8015241-Abstract-Conference.html.

Shreshth A Malik, Tiarnan Doherty, Panagiotis Tigas, Muhammed Razzak, Stephen J. Roberts, Aron Walsh, and Yarin Gal. MADE: Benchmark environments for closed-loop materials discovery. In *International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2601.20996.

Maya M. Martirossyan, Thomas Egg, Philipp Höllmer, George Karypis, Mark Transtrum, Adrian Roitberg, Mingjie Liu, Richard G. Hennig, Ellad B. Tadmor, and Stefano Martiniani. All that structure matches does not glitter. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-1331. Datasets and Benchmarks Track.

Shyue Ping Ong, William Davidson Richards, Anubhav Jain, Geoffroy Hautier, Michael Kocher, Shreyas Cholia, Dan Gunter, Vincent L. Chevrier, Kristin A. Persson, and Gerbrand Ceder. Python materials genomics (pymatgen): A robust, open-source python library for materials analysis. *Computational Materials Science*, 68:314–319, 2013. doi: 10.1016/j.commatsci.2012.10.028.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, D. Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/hash/8558cb408c1d76621371888657d2eb1d-Abstract.html.

Mianzhi Pan, Jianfei Li, Peishuo Liu, Botian Wang, Yawen Ouyang, Yiming Rong, Wei-Ying Ma, Hao Zhou, and Jianbing Zhang. Enhancing spatial reasoning in large language models for metal-organic frameworks structure prediction. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, pp. 11750–11761. Association for Computing Machinery, 2026. doi: 10.1145/3770855.3818846.

Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. SchNet: A continuous-filter convolutional neural network for modeling quantum interactions. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/303ed4c69846ab36c2904d3ba8573050-Abstract.html.

Brandon M. Wood, Misko Dzamba, Xiang Fu, Meng Gao, Muhammed Shuaibi, Luis Barroso-Luque, Kareem Abdelmaqsoud, Vahe Gharakhanyan, John R. Kitchin, Daniel S. Levine, Kyle Michel, Anuroop Sriram, Taco Cohen, Abhishek Das, Ammar Rizvi, Sushree Jagriti Sahoo, Zachary W. Ulissi, and C. Lawrence Zitnick. UMA: A family of universal models for atoms, 2025. URL https://arxiv.org/abs/2506.23971. arXiv preprint arXiv:2506.23971.

Liang Yan, Beom Seok Kang, Maurice D. Hanisch, Jian Ma, and Anima Anandkumar. MGB: The material generation benchmark. In *NeurIPS Workshop on AI for Accelerated Materials Design*, 2025. URL https://openreview.net/forum?id=K15Dqxm0ge.

A Data Construction and Split Definitions

The appendix follows the benchmark's use path: data construction (A), model implementation (B), evaluation (C), selection and confidence results (D–E), structural interpretation (F), and artifact usage (G).

A.1 Condition grouping and family assignment

Chemical-condition identity groups the supplied building-block graphs and multiplicities before role assignment. Multiple reference files for one condition remain together; a candidate recovers that condition if it matches any member of its reference set. The core reference structures have 26–876 atoms per cell. Known reference members need not exhaust a condition's possible polymorphs. The six core role counts sum to 5,120, and their attempts sum to $5{,}120 \times 3 \times 10 = 153{,}600$. Two repeated streams contribute $512 \times 3 \times 10 \times 2 = 30{,}720$ attempts on existing conditions. Confirmation adds $2{,}048 \times 3 \times 10 = 61{,}440$ attempts on additional conditions.

Family grouping uses a generic heavy-atom graph 2-core with a complete-heavy-graph fallback for acyclic linkers. It differs from a Murcko-scaffold definition. Before assigning roles, skeleton memberships are combined across all references of each condition. Source roles exclude every selected held-out skeleton. Within held-out skeleton groups, a fixed hash ordering and round-robin allocation interleave adaptation and test conditions. This keeps the two roles disjoint while distributing skeleton families across both. The 512 family-test conditions cover 23 skeleton groups. The adaptation role is reserved for methods using target-family supervision. The reported family comparisons use no adaptation labels.

A.2 Pool composition and data interfaces

Figure 4 shows where within-pool discrimination can change recovery. Mixed-label pools contain both correct and incorrect attempts; all-match pools test retention of available successes, while no-match pools provide no recoverable candidate. All three types contribute to the reported denominators. Positive counts are computed for evaluation and are excluded from scorer inputs.

The keys are `query_id` for a condition and `candidate_id` for an attempt. Draws are zero-indexed, so a $K = 5$ request selects draws below five. Within a generation stream, condition, generator, and draw identify a pool member. The generator field routes records to the requested source; it is excluded from learned feature vectors. Missing physical or graph features retain explicit status. Recorded export success describes available data, not physical validity.

A.3 Confirmation sampling and exposure history

Confirmation excluded recorded MOF-Select fitting, development, calibration, testing, early probes, specified earlier relationship experiments, and the declared reference-neighbor screen. A history

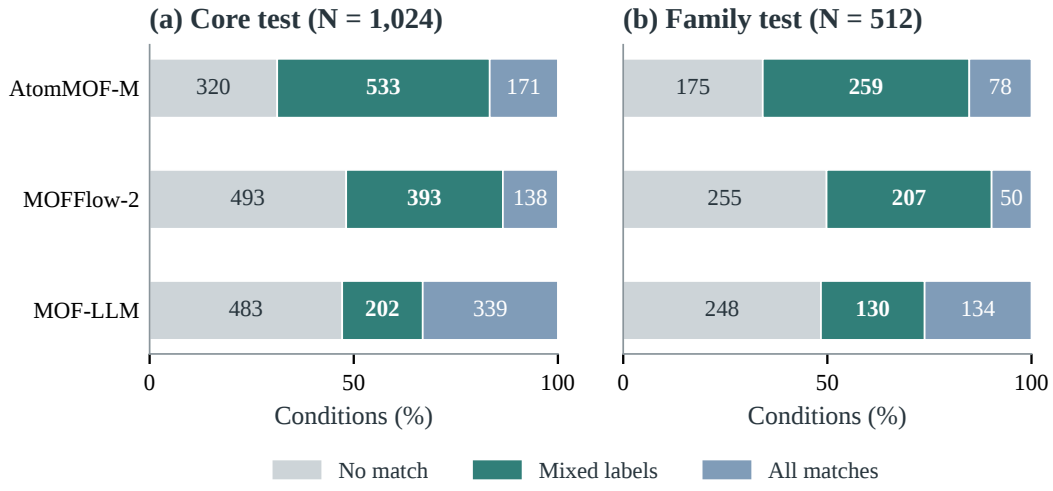

Figure 4: **Shared chemistry produces different selection problems.** Primary-label $K = 10$ pools, including failed attempts. No-match, mixed, and all-match pools contain zero, one–nine, and ten matches. Segment labels count conditions, each counted once per generator and role.

Table 7: **Data fields follow their learning role.** Identifiers join records; they are not predictive features. Reference-dependent quantities enter through separate supervision and evaluation interfaces.

Entity	Content	Use
Condition	Block elements, bonds, charges, chirality and multiplicities	Public chemical input
Attempt	Generator, draw, status, atom count, volume, CIF and periodic graph	Public candidate input
Pool / physical	Prefix-specific agreement, distances, error counts; UMA energy and forces	Optional public features
Labels / references	Primary and strict recovery; reference members, matches and exposure annotations	Role-specific supervision or evaluator metadata

check found that all 4,096 proposals appeared in an older official MOFFlow compatibility evaluation. Before final sampling and generation, the history rule was amended to retain task-specific exclusions and annotate that older exposure, leaving 4,060 eligible conditions before fixing 2,048. Earlier outcome values were not used in selection. Models, comparisons, labels, sample count, seeds, and stopping rules were fixed before the fresh candidate outcomes. There was one primary readout and no outcome-dependent sample extension. This is a task-new confirmation, not an externally registered or globally untouched holdout.

The artifact role `confirmation` names the original 1,024-condition core test; `prospective` identifies the later 2,048-condition confirmation. The paper uses "core test" and "confirmation" to distinguish them. Selector role separation and generator-training exposure are separate metadata.

B Generator and Baseline Implementations

B.1 Frozen generators and observable descriptors

Frozen generation uses AtomMOF-M with ten steps, MOFFlow-2 with 50 steps, and the archived MOF-LLM checkpoint. MOF-LLM uses stochastic sampling with temperature 0.1, top-$p = 0.6$, top-$k = 10$, and at most 2,048 new tokens. Here sampling top-k limits token choices and is distinct from the candidate budget K. Adapters requiring local building-block geometry use declared RDKit or training-source templates. Frozen outputs form the benchmark inputs, so selector training requires no generator weights. Upstream versions and generation manifests accompany the candidate records. Generation manifests retain run and draw identifiers, sampling seeds, and upstream revisions. MOF-LLM seeds each condition from its run identifier and condition ID, independently

of shard order. AtomMOF-M and MOFFlow-2 retain their recorded shard streams, including early completed shards. Repeated streams use new frozen run identifiers on the same conditions.

UMA-s-1p2 supplies single-point energy and force descriptors on the generated structures. These are scorer inputs, not relaxed structures or recovery labels. Energy is divided by atom count; relative energy subtracts the minimum within the requested pool. Force features use $\log(1+f)$ for maximum and RMS force. The observable size features are $\log n_{\mathrm{atoms}}$ and $\log(V/n_{\mathrm{atoms}})$. Agreement and pair-error counts are divided by $\max(1, K-1)$.

For Consensus, let h_i count matching neighbors and d_i be the mean normalized pair distance in the requested prefix. A pairwise nonmatch contributes distance one; a pair-computation exception is recorded separately and omitted from the distance sum, while the denominator remains $\max(1, K-1)$. The score is $h_i - 0.25 d_i/(1+d_i)$, so agreement count takes precedence over distance. Candidate ties then follow Appendix C.

B.2 Selector inputs and architectures

HGB uses the nine observable and physical scalars listed below. The two LightGBM families also receive public condition descriptors: building-block and atom counts, element fractions, charge and bond summaries. Both LightGBM families use identical features and development search grids. SchNet-pair uses the periodic structure encoder with classification and pairwise ranking losses. Full augments it with condition descriptors and pool context.

Full uses 128-dimensional element and block-role embeddings, four SchNet interaction blocks, 50 radial basis functions, a 6 Å cutoff, and at most 48 neighbors per atom. Node means and maxima, means for two block roles, and two global graph features form its structure representation. Separate projections encode public chemistry, observable scalars, and physical scalars. The five observable scalars are log atom count, log volume per atom, agreement, consensus distance, and errors. The four physical scalars are log maximum force, log RMS force, relative energy, and physical-feature missingness. Normalization uses training data only; each scalar retains a missing-value indicator. The candidate representation is scored with its pool mean, pool maximum, log budget, and valid-candidate fraction.

Attention-Full inserts four-head attention without positional embeddings, masked padding, residual normalization, and a 128–256–128 feed-forward block before pool aggregation. All-failed pools remain in evaluation but do not enter attention. Training combines binary cross-entropy with the mean within-pool positive–negative loss

$$\mathcal{L}_{\mathrm{pair}} = \operatorname*{mean}_{q:\, P_q, N_q \neq \emptyset} \frac{1}{|P_q||N_q|} \sum_{i \in P_q} \sum_{j \in N_q} \mathrm{softplus}(-(z_i - z_j)), \tag{3}$$

where z_i is a candidate logit, and P_q, N_q are the positive and negative candidate sets. All-positive and all-negative pools contribute to classification. The total loss is $\mathcal{L}_{\mathrm{BCE}} + \mathcal{L}_{\mathrm{pair}}$ with unit weights: binary cross-entropy averages scoreable candidates in the batch, and the pair term averages mixed-label pools and is zero when none occur. The shared scoring head has dropout 0.1; attention itself has zero dropout. AdamW uses learning rate 10^{-3}, weight decay 10^{-4}, cosine scheduling over 5,120 updates, gradient clipping at norm ten, and batches of 16 candidate pools. Development checkpoint selection occurs every 128 updates using Selected@1, with binary cross-entropy as tie criterion. Training seeds are 2026092011, 2026092012, and 2026092013.

HGB searches leaf counts $\{7, 15, 31\}$ and ℓ_2 penalties $\{0.1, 1\}$ with 200 iterations. The same-feature LightGBM classifier and LambdaMART ranker use 300 trees, leaf counts $\{15, 31, 63\}$, and learning rates $\{0.03, 0.1\}$, with six CPU threads. Ranking truncation level is three. Each tree family selects among six configurations on development data.

Eight source modes comprise three specialists, balanced-all, all-source, and three leave-one-source-out recipes. Allocation follows a label-independent hash. Leave-one-source-out excludes the named generator from training, development, and calibration. All-source retains three separate ten-candidate pools per condition, with condition normalization fitted once per unique condition. All modes use 2,048 distinct training conditions. The all-source mode has 6,144 training pools; the others have 2,048. Development and calibration each use 512 conditions with one pool per condition.

Balanced-all and all-source share their assignments in both roles. Full and both LightGBM families cover all eight source modes; Attention-Full and HGB cover the three specialists.

Table 8: **Coverage of the main learning comparisons.** Every listed learned recipe uses three fixed seeds. "All eight" comprises three specialists, balanced-all, all-source, and three leave-one-source-out recipes.

Family	Candidate inputs	Source-recipe coverage
HGB	Nine observable / physical scalars	Three specialists
LightGBM / LambdaMART	Same scalars and chemistry	All eight, matched search
SchNet-pair	Periodic structure graph	Three specialists
Full	Structure, chemistry, scalars and pool context	All eight
Attention-Full	Full inputs and candidate attention	Three specialists
Input variants	Structure with chemistry / pool ablations	AtomMOF-M and MOFFlow-2

Reference-supervised contrastive control. A separate chemical encoder has three message-passing layers of width 128, with bond, charge, and chirality embeddings. Its 128-dimensional output is aligned with the periodic structure encoder using symmetric multi-positive InfoNCE. One known training reference is sampled per condition per update; same-condition references are never negatives. AdamW uses learning rate 3×10^{-4}, weight decay 10^{-4}, 5,120 updates, batches of 32 conditions, and initial temperature 0.07. Checkpoints maximize mean development Selected@1 on AtomMOF-M and MOFFlow-2. Inference scores cosine similarity between chemistry and each generated structure. Its extra reference supervision is identified in Table 9.

B.3 Confidence functions

For fixed Consensus recommendations, query-level HGB classifiers predict selected recovery from size, chemistry, pool statistics, or combined chemistry and pool statistics. Each uses 2,048 training conditions and selects among four configurations on 512 development conditions by AURC, then log loss. Controls transform the selected Consensus score using its development empirical CDF, logistic scaling, or isotonic regression. These models do not change the selected candidate. Neural selectors use sigmoid confidence in the probability-grid and source-tail analyses. The raw-score-quantile protocol uses the selected logit for Full, including the generator-transfer evaluation in Figure 3. LightGBM classification uses its selected probability, while LambdaMART retains its raw ranking score. Threshold grids are fixed or derived from source development data before independent calibration (Appendix C). The seven functions in the source-tail diagnostic are enumerated in Appendix E.

B.4 Implementation versions and corrections

The corrected physical-input-dependent matrix has 90 model instances: 48 LightGBM models, 24 Full models, nine Attention-Full models, and nine HGB models, including 18 reused instances. Unaffected SchNet and no-physical-input controls retain their original implementations. Historical small-budget, adaptation, and contrastive extensions have narrower coverage rather than complete corrected matrices.

A physical-feature reader repair affected 343 historical candidates: 307 core and 17/19 in repeated streams. Candidate CIFs and labels were unchanged. Affected models were refitted and recalibrated under the fixed main recipe. Fresh confirmation candidates already used complete-site features. The original frozen confirmation is preserved; corrected models on the same conditions provide a posthoc implementation sensitivity.

C Metrics, Calibration, and Statistical Procedures

C.1 Matching and failure handling

The primary call is `get_rms_dist(reference, candidate)` with `stol=0.5`, `ltol=0.3`, `angle_tol=10`, and `scale=True`. A non-null result for any same-condition

reference defines $y_i = 1$. Strict matching uses `stol=0.3`; the fine audit also uses `angle_tol=1`. Site tolerance is normalized rather than expressed in angstroms. RMS matching and maximum-displacement `fit` are audited separately.

Generation and parsing failures contribute zero recovery. A condition with no scoreable candidates receives a failed recommendation. Unresolved matching exceptions block complete readout. Ties use a fixed label-independent hash after method-specific criteria. Random@1 is the exact expectation over uniform draws. First evaluates draw zero, including its failure status. No rule receives an additional generation attempt when a structure or feature is unavailable.

C.2 Budget transitions and confidence ordering

Let $s_K(q), o_K(q)$ be selected recovery and oracle coverage. For nested budgets $K < L$, the transition identity is

$$\begin{aligned} S_L - S_K = \; & \Pr(o_K = 0, s_L = 1) + \Pr(o_K = 1, s_K = 0, s_L = 1) \\ & - \Pr(s_K = 1, s_L = 0). \end{aligned} \tag{4}$$

Probabilities denote empirical fractions of evaluated conditions. The terms are new hits, rescues, and losses. Opportunity is $O_L - O_K = \Pr(o_K = 0, o_L = 1)$; each opportunity need not become a new hit.

Equation 2 defines coverage $C(\tau)$, accepted risk $\rho(\tau)$, and correct yield $J(\tau)$ for threshold τ. At positive coverage, $J(\tau) = C(\tau)[1 - \rho(\tau)]$. AURC is the mean prefix error rate over all N confidence-ranked conditions, averaging over random orderings within tied confidence groups. Unscoreable conditions receive confidence $-\infty$ and remain in the evaluation denominator. For a confidence-tie group of size m with error fraction $\bar{e}$, suppose b conditions and E errors precede it. The expected error rate at its jth position is $(E + j\bar{e})/(b + j)$, for $j = 1, \ldots, m$. AURC averages these values over all positions. This removes dependence on arbitrary row order without changing candidate-selection ties.

C.3 Threshold calibration

The fixed probability grid is $0.50, 0.55, \ldots, 0.95$. Development quantiles span the zeroth through ninetieth percentiles in ten-point increments. The single-threshold alternative chooses the quantile with greatest development coverage subject to empirical risk at most α, before independent calibration. If no development threshold is eligible, it abstains. For n accepted calibration conditions and $e < n$ errors, the one-sided Clopper–Pearson upper bound at tail probability δ' is $\mathrm{Beta}^{-1}(1-\delta'; e+1, n-e)$; all-error or empty acceptance has upper bound one. Threshold searches correct within each scoring function, using δ' equal to δ divided by the number of tested thresholds. This does not control the entire inspected model suite simultaneously. Threshold grids are fixed or derived from development data before calibration. Among eligible thresholds, the chosen rule maximizes the calibration accepted count; an equal-count tie retains the first threshold in the recorded grid. The one-threshold rule has no threshold-search multiplicity penalty. Probability-grid and raw-score-quantile results are kept as separate protocols.

C.4 Paired intervals and aggregation

Bootstrap comparisons average fixed seeds within condition and resample paired conditions 10,000 times. Repeated streams are averaged within condition. Family resampling draws from 23 groups and divides summed outcomes by the sampled condition count. Main intervals are exploratory pointwise 95% intervals. The two original locked confirmation contrasts use nominal 97.5% intervals each; corrected-input intervals retain the same level with a posthoc interpretation. For a method contrast, the resampled value is the within-condition difference of seed-averaged outcomes. Source-macro comparisons first average the three generator-specific differences for each shared condition, then resample conditions. They do not treat three generators or repeated seeds as extra independent samples. Percentile endpoints are taken from the resulting bootstrap distribution. Best table entries mark point estimates; interval colors identify pointwise contrasts and do not imply a suite-wide significance guarantee.

D Additional Selection and Transfer Results

D.1 Simple rules and alternative supervision

Table 9 completes the control comparisons with the first attempt, physical ranking rules, and chemical–structure contrastive learning. Physical descriptors and pool agreement rank different aspects of the same candidates. Their relative point estimates vary by source and role. The contrastive control uses training reference geometry and is therefore reported in a separate supervision group. It offers a reference-supervised alternative with similar MOF-LLM family point recovery (40.30%) to Full (40.17%).

Table 9: **Simple rules and the reference-supervised control.** Selected@1 (%), primary label, $K = 10$; corrected UMA descriptors. Bold marks the best of six simple rules. *Contrastive learning additionally uses training reference structures (three-seed mean).

	AtomMOF-M		MOFFlow-2		MOF-LLM	
Rule / control	Core	Family	Core	Family	Core	Family
First attempt	40.04	35.74	30.27	26.17	41.99	36.52
Random	39.07	35.37	30.08	27.05	41.97	36.60
Lowest energy/atom	41.89	38.09	31.25	31.25	**43.55**	**39.06**
Lowest RMS force	41.50	37.30	31.45	31.45	43.07	**39.06**
Lowest maximum force	41.02	36.91	31.25	32.81	43.07	38.28
Consensus	**46.88**	**39.06**	**34.38**	**33.59**	43.16	36.33
Chemical–structure contrastive*	42.12	37.57	32.03	28.32	44.40	40.30

D.2 Label sensitivity and learning controls

Table 10: **Recovery under stricter matching.** Selected@1 (%), site tolerance 0.3, $K = 10$; core $n = 1{,}024$, family $n = 512$. Learned specialists: three-seed means. Oracle gives coverage; bold marks the largest selected-recovery point estimate per column.

	AtomMOF-M		MOFFlow-2		MOF-LLM	
Selector	Core	Family	Core	Family	Core	Family
Oracle@10	35.06	30.47	21.19	16.80	24.41	15.82
Random	14.45	11.17	8.97	5.55	17.41	9.57
Consensus	17.48	13.48	9.96	**6.64**	17.19	9.38
Lowest energy/atom	16.21	12.70	9.86	6.05	**20.31**	**12.50**
HGB	17.09	12.50	10.06	6.25	19.14	10.94
LightGBM	17.29	13.48	9.67	6.05	19.14	10.94
LambdaMART	17.09	12.30	**10.45**	6.25	17.77	10.16
SchNet-pair	16.05	10.74	9.99	5.66	19.69	11.26
Full	17.09	12.57	10.09	6.32	20.15	10.68
Attention-Full	**17.51**	**13.54**	10.09	6.58	19.60	10.68

Table 10 evaluates the main selectors at stricter tolerance using the same weights as primary-label evaluation. This tests the geometric precision of already selected structures. MOF-LLM Full remains above Consensus in point estimates; the family-group interval for its relative gain crosses zero. The lowest-energy rule has the highest strict-label point estimates on both MOF-LLM tests, whereas learned selectors lead under the primary label (Table 2). Thus matching tolerance can change the leading selection rule.

The current machine-readable matrix retains model family, source recipe, seed, target generator, role, budget, and label. Its 744 cells include 720 learned-model cells and the single-source, balanced, all-source, and leave-one-source-out settings. They share conditions and are not independent confirmation experiments.

Table 11 separates chemistry, pool information, and physical inputs on the two generators with complete input variants. Relative to the chemistry–pool variant, Full's physical features change

recovery by -0.20 pp $[-1.20, 0.81]$ and -0.39 pp $[-1.82, 1.04]$ on AtomMOF-M core and family tests, and $+0.23$ pp $[-0.98, 1.46]$ and $+2.21$ pp $[0.39, 4.10]$ on MOFFlow-2. These are pointwise 95% paired intervals.

Table 11: **Input ablations with matched encoders and training budgets.** Selected@1 (%), primary label, $K = 10$; three-seed means. Pool adds observable scalars and mean/max aggregation; Full adds physical features to chemistry–pool. Bold marks column maxima.

	AtomMOF-M		MOFFlow-2	
Inputs	Core	Family	Core	Family
Structure	42.51	38.61	30.99	28.26
Structure + chemistry	43.07	37.76	31.67	28.97
Structure + pool	46.61	38.87	**33.82**	30.40
Structure + chemistry + pool	**46.84**	**40.04**	32.98	30.40
Full (+ physical features)	46.65	39.65	33.20	**32.62**

Learning objectives and source supervision. Across 48 matched-feature LambdaMART-minus-classification comparisons, 12 pointwise 95% intervals lie below zero and 36 include zero; none lies above zero. At $K = 10$, none of the 18 source/test comparisons supports balanced multi-source training over its specialist at equal label count. None of the 18 comparisons of all-source against balanced training has an interval above zero either. The positive $K = 5$ Full macro contrasts in Table 5 use the same $K = 10$-trained models. The intervals assess each contrast separately; they do not test a difference in supervision effects between budgets.

D.3 Transfer to a held-out generator

The 36 primary-label $K = 10$ leave-one-source-out contrasts compare two-source training with each constituent source specialist on the held-out generator. Figure 5 pairs each estimated recovery difference with its uncertainty interval. Twenty-five of the 36 estimates have magnitude below 1 pp. Five pointwise 95% intervals lie above zero and 31 include zero. The interval-supported gains span all three learned families. The largest is 2.99 pp [0.88, 5.11], from LambdaMART on AtomMOF-M core conditions relative to MOF-LLM-only training. Full also gains on held-out MOF-LLM families relative to MOFFlow-2-only training. These results identify useful source mixtures for specific targets; the remaining intervals leave the direction of benefit unresolved. Held-out-generator labels enter neither fitting, development selection, nor calibration.

D.4 Resampling, redundancy, and label sensitivity

Table 12 reports the MOF-LLM robustness checks with condition or family resampling as indicated.

Table 12: **MOF-LLM Full minus Consensus under robustness checks.** Gains (pp) with pointwise 95% bootstrap intervals; teal marks intervals above zero. N: distinct conditions. New streams are averaged within condition; family resampling uses 23 groups; strict site tolerance is 0.3.

Setting	Role	N	Resampling	Gain	95% interval
Original pools	Core	1024	Condition	+1.37	[-0.23, 2.93]
Original pools	Family	512	Family	**+3.84**	[1.43, 5.98]
Two new streams	Core	256	Condition	**+4.30**	[1.43, 7.23]
Two new streams	Family	256	Condition	**+5.01**	[1.82, 8.27]
Strict label	Family	512	Family	+1.30	[-0.88, 3.29]

After excluding three core and three family conditions incident to reference neighbors, the corrected MOF-LLM Full-minus-Consensus core gain remains 1.37 pp at two-decimal precision, and the family gain changes from 3.84 to 3.86. Across the two additional streams, the family gain changes from 5.01 to 5.07. Complete per-condition outcomes and contrast tables are included in the artifact; this appendix presents the principal label and redundancy sensitivities.

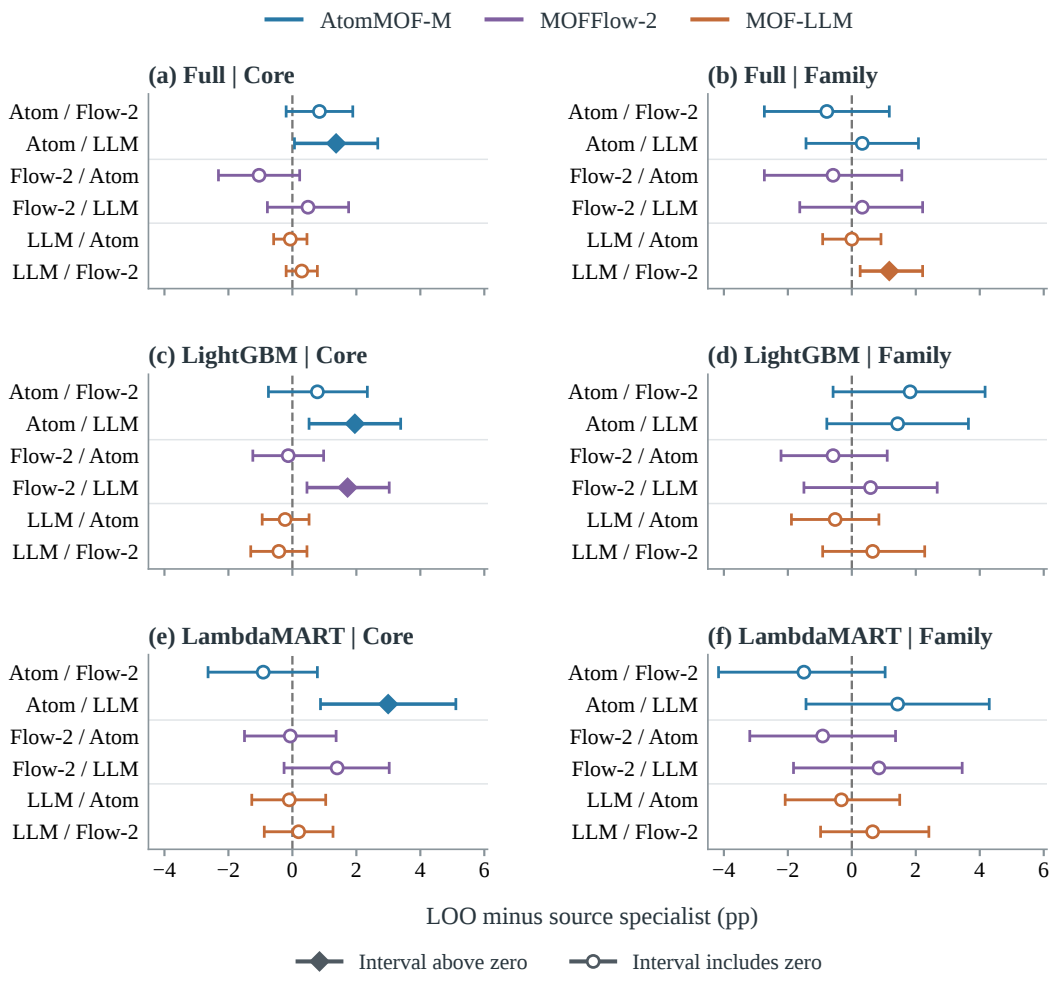

Figure 5: **Complete leave-one-source-out selection comparisons.** Three-seed recovery differences (pp) with pointwise 95% paired condition-bootstrap intervals; primary label, $K = 10$. Rows compare two-source training with the named constituent specialist on the held-out target, at equal label budget. Labels abbreviate *target / specialist*; colors identify targets. Filled diamonds mark intervals above zero; open circles mark intervals including zero.

Excluding exact-repeat pools leaves MOF-LLM gains of 1.56 pp (core, $n = 639$) and 3.38 pp (family, $n = 414$). On core pools with one to four matches, Full gains 26.88–40.91 pp over Consensus; with six to nine, it loses 18.52–42.59 pp. These descriptive, label-defined strata each contain 17–31 conditions.

E Confidence Learning and Selective Acceptance

E.1 Confidence inputs at fixed candidate choices

Table 13 compares all seven confidence families on the same Consensus recommendations. Size and chemistry alone provide different signals from the realized candidate pool. On MOF-LLM, pool features improve ordering, and adding chemistry gives the lowest core AURC; pool-only confidence has the lowest family point estimate. On AtomMOF-M and MOFFlow-2, transformed Consensus scores remain competitive. This comparison isolates confidence from candidate choice. Score transformations can change AURC when they introduce confidence ties.

The paired comparison of selected-score ECDF and condition–pool confidence appears in Table 6 in the main text. It retains the same recommendations and outcomes for all six source/role settings.

Table 13: **Confidence inputs and score transformations.** AURC on fixed Consensus recommendations, primary label, $K = 10$ (lower is better). Learned functions: three-seed means; bold: column minima. Paired intervals for ECDF versus condition–pool appear in Table 6.

	AtomMOF-M		MOFFlow-2		MOF-LLM	
Confidence	Core	Family	Core	Family	Core	Family
Selected-score ECDF	**0.2257**	**0.3047**	0.3389	0.3630	0.3409	0.4137
Logistic scaling	**0.2257**	0.3049	**0.3383**	**0.3628**	0.3408	0.4135
Isotonic regression	0.2265	0.3067	0.3408	0.3659	0.3491	0.4126
Size	0.3893	0.3924	0.4628	0.4352	0.4550	0.4468
Condition	0.3315	0.3770	0.4182	0.4386	0.3786	0.4442
Pool	0.2297	0.3116	0.3414	0.3706	0.3013	**0.3774**
Condition + pool	0.2331	0.3197	0.3401	0.3676	**0.2894**	0.3792

E.2 Empirical tail error and calibration uncertainty

The source-tail diagnostic uses a different set of seven functions per generator: three Full-selector seeds, three condition–pool confidence seeds for fixed Consensus recommendations, and one empirical-CDF transform of the selected Consensus score. They represent seven fixed functions, not seven independent model families. Table 14 counts their calibration outcomes for the single-threshold setting. The diagnosis is specific to the scoring functions, threshold rule, and risk target shown.

Table 14: **Source certification separates tail error from uncertainty.** Counts among seven fixed scoring functions per generator, $K = 10$, 512 calibration conditions, $\alpha = 0.10$, one development-selected threshold. Bound-limited: empirical risk $\leq \alpha$ but upper bound $> \alpha$; tail error: empirical risk $> \alpha$ at that threshold.

Generator	Certified	Bound-limited	Tail error
AtomMOF-M	**4**	**3**	0
MOFFlow-2	0	**7**	0
MOF-LLM	0	0	**7**

Table 15 reports test outcomes for all four certified functions in this diagnostic: three AtomMOF-M Full seeds and Consensus with ECDF confidence. The other 17 functions comprise three AtomMOF-M condition–pool functions and all seven functions for each remaining generator. They abstain on both test roles, with zero coverage and correct yield and undefined risk. The same source-calibrated threshold is used on core and family tests, with family outcomes treated as empirical transfer results.

Table 15: **Test outcomes of all four certified functions.** AtomMOF-M, primary label, $K = 10$; 512 calibration conditions, one development-selected threshold, $\alpha = 0.10$, $\delta = 0.05$. Core/family $N = 1{,}024/512$. C, ρ, and J are coverage, accepted error, and correct yield (%).

Rule	Test	Accepted	Errors	C	ρ	J
Full (seed 1)	Core	204	9	19.92	4.41	19.04
	Family	93	7	18.16	7.53	16.80
Full (seed 2)	Core	196	5	19.14	2.55	18.65
	Family	75	3	14.65	4.00	14.06
Full (seed 3)	Core	202	4	19.73	1.98	19.34
	Family	100	5	19.53	5.00	18.55
Consensus + ECDF	Core	193	4	18.85	2.07	18.46
	Family	95	6	18.55	6.32	17.38

The 7,992 operating-point records identify the scoring function, source, seed, target generator, role, label, risk target, threshold rule, calibration counts and bound, target counts, coverage, and correct yield. Zero acceptance is retained explicitly and never represented as a measured zero error rate.

E.3 Transfer of a fixed acceptance rule

Figure 3 includes all three fixed seeds of leave-MOF-LLM-out Full. Confidence is the selected raw logit, with thresholds drawn from source-development quantiles. The accepting seed, 2026092011, uses threshold 2.88258, with 95 source-calibration acceptances and an upper bound of 0.075617688. This fixed threshold is applied to all three generators in both test roles; each point pairs the resulting coverage and accepted error. On held-out MOF-LLM, it accepts 533 core and 239 family queries, with 174 and 88 errors: risks of 0.326454 and 0.368201. Seeds 2026092012 and 2026092013 have no certified source threshold and reject all queries on all three generators. The figure's lower annotation represents these 12 empty operating points across seeds, generators, and roles; their risks are undefined.

The source-tail analysis separates empirical confidence quality from calibration uncertainty. The transfer example observes a fixed rule under a changed candidate distribution. These are distinct from jointly learning candidate choice and confidence, which remains an available use of the benchmark interfaces.

F Structural Quality, Redundancy, and Exposure

F.1 Geometric precision and local structure

The fixed audit covers 1,920 candidates from 64 conditions and 66 references, reproducing primary and strict labels with no unresolved geometric computations. The sample contains 32 core and 32 family conditions, all ten attempts per source, and no outcome-based filtering. These are descriptive checks on fixed conditions. Figure 6 separates sensitivity to matching tolerance from local contacts. Primary matches include contacts below 0.5 Å in 9/262 AtomMOF-M, 7/215 MOFFlow-2, and 6/261 MOF-LLM candidates. Under the strict label, the corresponding counts are 1/119, 0/66, and 0/121. Geometric matching and local-contact screening therefore provide complementary information.

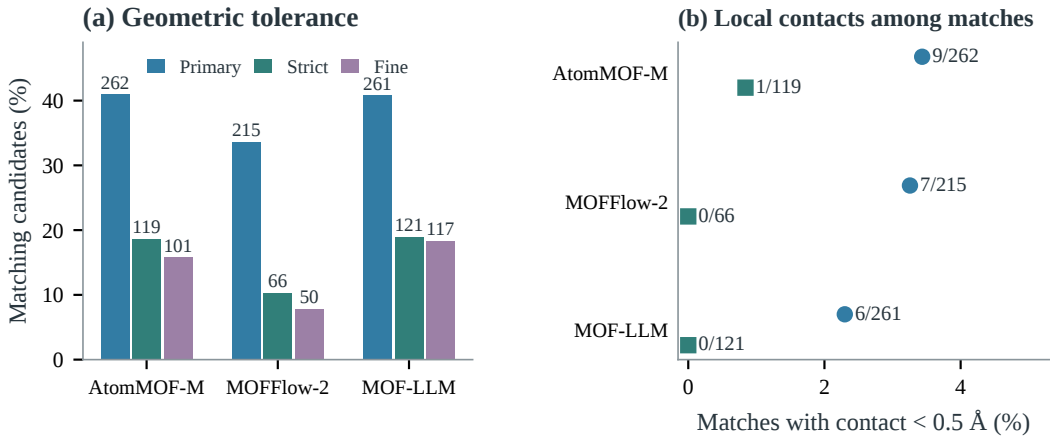

Figure 6: **Recovery tolerance and local structural quality.** (a) Matches among 640 candidates per generator: primary/strict site tolerances 0.5/0.3; fine also tightens angle tolerance to 1°. (b) Periodic contacts below 0.5 Å among primary (blue circles) and strict (teal squares) matches. Labels give contact-positive/matching counts. These are descriptive geometric checks, not synthesizability estimates.

Table 16 adds continuous distances among primary matches. The reported angstrom values use the matcher-aligned average lattice, including its volume normalization. They are not unaligned coordinate errors. Turning off rescaling or requiring a same-reference volume agreement within 10% retains all primary labels in this audit. The tolerance sensitivity is therefore not explained by that volume check on these conditions. An unmatched structure may be a plausible arrangement absent from the reference set.

Table 16: **Continuous displacement among primary matches.** Distances (Å) use the matcher-aligned average lattice after volume scaling. Unmatched candidates have no imputed distance. P90 denotes the 90th percentile.

		RMS displacement		Maximum displacement	
Generator	Matches	Median	P90	Median	P90
AtomMOF-M	262	0.69	1.22	2.21	3.60
MOFFlow-2	215	0.95	1.36	2.64	3.85
MOF-LLM	261	0.72	1.37	2.20	3.79

Additional diagnostics use the supplied chemical graph. A nonmetal bond is flagged when its periodic distance divided by the sum of RDKit covalent radii falls outside $[0.65, 1.5]$. Metal coordination counts N, O, F, P, S, Cl, Br, and I neighbors within 1.3 times the summed covalent radii and compares donor-count multisets with same-condition references. Atom indexing is checked against the generator-native graph before using these relations. These heuristics describe local geometry without imposing a universal coordination number.

F.2 Reference redundancy and training exposure

The reference audit evaluates all 152 declared same-composition cross-role pairs without the former volume prefilter. Eight loose RMS matches in seven condition pairs all fail the corresponding `fit` criteria. Six test conditions are incident to these pairs; Appendix D gives exclusion sensitivity. Exact repeated-CIF checks concern byte identity rather than all geometric equivalences. Positive-candidate count is evaluator metadata, never a selector input.

Available MOFFlow-2 training/validation inventories identify 305 overlapping core conditions. Complete AtomMOF and MOF-LLM training-condition exposure is unknown. These inventories annotate generator exposure separately from the disjoint selector roles. Family-group sensitivity tests the stated chemical partition; primary and strict outcomes are reported separately.

G Artifact Usage and Computational Costs

G.1 Packages and a minimal use path

The accompanying 93 MB review supplement contains the complete tabular benchmark, including the core, repeated streams, and confirmation, with labels and role manifests. It supplies the physical-feature corrections, training, scoring, and evaluation runtimes, frozen analysis records, and one saved LightGBM example. Candidate CIF/graph shards, the full selector checkpoints, and confirmation candidate scores are indexed as separate companion assets.

The use path applies the patch, trains on training/development roles, scores public inputs, and evaluates through a separate label join. Training requires no test or calibration labels; inference opens no supervision. Distributed labels remain accessible to package owners, so this is a protocol boundary rather than a hidden label service. Engineering smoke runs are marked and rejected by formal evaluation.

The following CPU path starts in the extracted review supplement. The unpacker verifies the archives and applies the core and repeated-stream corrections. The training command runs the formal six-configuration LambdaMART recipe; every output directory must be new.

```
python unpack.py --output workspace
R=workspace/mof-select-physical-correction-v3
T=$R/runtimes/training
python -m pip install -r $T/requirements-tested.txt
B=workspace/corrected-core-lite-v3
python $T/train.py --bundle "$B" --family lambdamart \
  --mode balanced_all --seed 2026092011 --output runs/ranker
python $T/score.py --bundle "$B" --model runs/ranker \
  --generator atom_m --split confirmation --k 10 \
  --output runs/ranker_scores.csv
python $T/evaluate.py --bundle "$B" --model runs/ranker \
  --output runs/ranker_evaluation
```

Here `confirmation` is the original core-test role (Appendix A). The same core patch must precede current-model inference. Repeated streams use their corresponding patches; fresh confirmation already contains complete-site features and retains its locked outcomes.

The tested tabular environment uses Python 3.10, NumPy 1.25.2, pandas 2.3.3, SciPy 1.15.2, scikit-learn 1.7.2, and LightGBM 4.6.0. Graph methods additionally use PyTorch 2.4.1 and PyG 2.6.1 with compatible CUDA dependencies. The structure audit uses pymatgen 2024.10.22. Package requirements preserve these tested versions.

G.2 Reproducibility checks and computational cost

Local isolated-consumption checks reproduce selected candidates from 90 models on 540 cells with four fixed conditions each: 2,160 repeated query checks and 16,200 candidate scores. Maximum score difference is approximately 1.34×10^{-5}. Portable training reproduces two complete tree fits exactly and exercises three Full modes for 32 updates. These local checks validate package consumption and the tested training interfaces. Reconstruction checks the stored selection comparisons and 7,992 calibration operating points.

The assembled review supplement also passes extraction and data checks covering all 245,760 attempts and 7,168 unique conditions. Its saved LightGBM example scores all 10,240 core-test attempts, with labels joined after scoring. A 20-iteration CPU demo trains on 128 conditions without reading test or calibration labels. These checks validate package use and add no new scientific comparison.

Neural batches use four H100 GPUs as independent workers, with 88 CPU cores and 960 GB host memory. The fixed-budget implementation-unification batch took about 65 minutes of node wall time, including CPU work and reused instances. This is not the cost of retraining every reported baseline. Generation ledgers cover their recorded cohorts; equal candidate counts do not equalize GPU time, and partial costs are not summed into an unmeasured total. The initial generation and decision-learning cohort used 6.30 node-hours (25.20 allocated GPU-hours), and the subsequent pool-baseline cohort used 2.33 node-hours (9.30 allocated GPU-hours). These ledgers exclude earlier costs of reused outputs. Allocated GPU-hours include process startup and CPU stages, rather than measuring active GPU execution or per-generator inference speed.

G.3 Availability and upstream terms

The supplement includes review-use permission for the authors' original contributions and preserves upstream attribution and terms. Raw reference CIFs, upstream databases, and pretrained generator checkpoints are excluded. Tabular learning and evaluation require none of these assets. The companion manifest and downloader identify the exact structural payloads, selector weights, and confirmation candidate scores needed for the remaining reproduction paths. Anonymous hosting of these larger companion assets remains pending.