SAME MOLECULE, DIFFERENT KNOWLEDGE: CROSS-IDENTIFIER BINDING FAILURES IN LLMS FOR CHEMISTRY

Anonymous authors
Paper under double-blind review

ABSTRACT

Chemical AI workflows routinely represent the same molecule through multiple co-referring identifiers, including trivial names, IUPAC names, and SMILES, implicitly assuming that these representations provide interchangeable access to an LLM's knowledge. To test this assumption, we first introduce a controlled benchmark and show that it does not hold. Across foundation models spanning 7B to 72B parameters, even chemistry-specialized models, co-referring molecular identifiers consistently expose different subsets of the model's knowledge. We then conduct a series of controlled behavioral, representational, and causal analyses to characterize this failure. We show that even when a model can identify a molecule from its SMILES and retrieve a fact through the corresponding name, the same fact can remain inaccessible directly from SMILES. We term this phenomenon *cross-identifier binding failure*. Further analysis of internal representations, conflicting identifiers, and parameter editing shows that neither improved chemical representation competence nor modification of individual facts is sufficient to establish reliable access across identifiers. Together, these results indicate that the core failure lies in the binding between an identifier and knowledge already present in the model. Motivated by this diagnosis, we introduce Binding Self-Distillation, a simple mitigation that uses lexical identity during training to transfer the model's existing knowledge access to the structural route. This enables factual retrieval directly from SMILES at inference time, without requiring a name lookup or training the underlying model from scratch.

1 INTRODUCTION

Large language models (LLMs) are increasingly used as general-purpose interfaces to knowledge and reasoning (Brown et al., 2020; Ouyang et al., 2022; Kojima et al., 2022). Beyond natural-language generation (Wei et al., 2022), they support applications in programming (Chen et al., 2021), medicine (McDuff et al., 2025; Johri et al., 2025), and scientific discovery (Singhal et al., 2023; Zhang et al., 2024b). In chemistry, LLMs are used for chemical question answering (Zhang et al., 2024a; Zhao et al., 2025), molecular property prediction (Jablonka et al., 2024), and molecule design (Bhattacharya et al., 2024). As these models become components of chemical workflows, an important question is whether the knowledge they contain can be accessed reliably under the different representations used to describe the same molecule.

A single molecule can be referred to through multiple co-referring identifiers (Kim et al., 2023). For example, as shown in Figure 1(A), aspirin may appear as the trivial name *aspirin*, the systematic IUPAC name *2-acetyloxybenzoic acid* (Favre & Powell, 2013), or the SMILES string `CC(=O)Oc1ccccc1C(=O)O` (Gómez-Bombarelli et al., 2018). These representations are routinely mixed across chemical workflows: biomedical resources commonly use lexical names (Hastings et al., 2016), whereas reaction datasets and molecular generative pipelines frequently use structural representations such as SMILES (Gómez-Bombarelli et al., 2018; Schwaller et al., 2019). Consequently, the same LLM may be queried about the same molecule through different identifiers. This practice relies on an implicit assumption: *co-referring identifiers should be interchangeable routes to the model's knowledge*. More generally, equivalence at the level of the underlying scientific entity is implicitly expected to translate into equivalence as an interface to the model's memory.

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

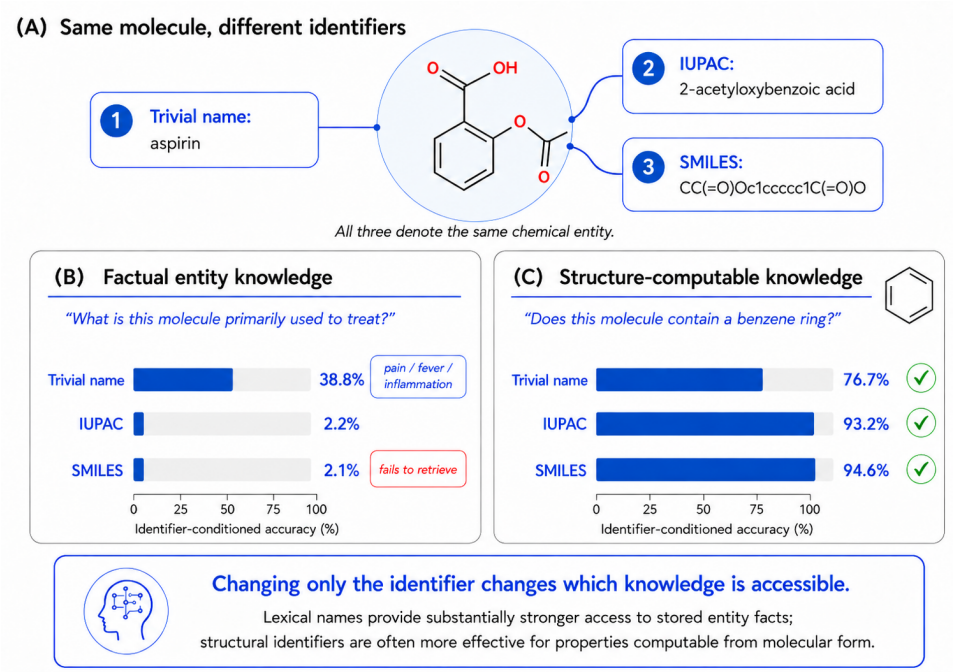


Figure 1: **Equivalent molecular identifiers expose different knowledge.** **(A)** A trivial name, IUPAC name, and SMILES string denote the same molecule (aspirin). **(B–C)** Identifier-conditioned accuracy over the 896 molecules in our benchmark, holding the molecule and queried property fixed while changing only the identifier. Lexical names provide substantially stronger access to entity-associated factual knowledge (**B**, therapeutic indication), whereas structural identifiers provide stronger access to structure-computable knowledge (**C**, benzene-ring presence). Representative results are shown for ChemDFM-v2.0-14B (Zhao et al., 2025).

In this work, we first show that this assumption does not hold. We construct **ChemIDBench**, a controlled benchmark for evaluating knowledge access across co-referring molecular identifiers, covering approved drugs with matched trivial names, IUPAC names, and SMILES, and evaluate seven chemistry-specialized and general-purpose LLMs. Holding the underlying molecule and queried property fixed while changing only its identifier reveals a systematic double dissociation: as shown in Figure 1(B,C), lexical names provide substantially stronger access to entity-associated factual knowledge, whereas structure-descriptive identifiers provide stronger access to properties computable from molecular form. Identifier choice therefore does more than change the difficulty of a query; it changes *which knowledge is reachable*. More surprisingly, this asymmetry persists even when the model recognizes the molecule. We isolate cases in which a model can generate the correct trivial name from SMILES and retrieve a fact through that name, yet still fails to retrieve the same fact directly from SMILES. We call this phenomenon *cross-identifier binding failure*. It separates three notions that are often implicitly conflated: recognizing the entity, possessing factual knowledge about it, and binding a particular identifier to that knowledge.

We next ask what prevents knowledge from transferring across identifiers and whether the missing access can be recovered. Supplying the lexical name restores much of the inaccessible factual knowledge, showing that the knowledge is present but not reached through the structural route. More importantly, activation patching provides causal evidence for this account: transplanting lexical-route states into a structural query restores inaccessible factual retrieval under SMILES alone. Layer-wise analysis further shows that chemistry-specific training can improve alignment between structural forms without establishing comparable name-structure binding. This weak binding is also a reliability problem: when conflicting identifiers are presented together, different access routes can control different answers, producing inconsistencies that models often fail to detect. Supplying the lexical identity is an effective inference-time remedy, but it requires an identifier lookup on every query. Standard knowledge editing does not reliably propagate factual access across co-referring identifiers, suggesting that modifying the factual association alone is insufficient. Motivated by this diagnosis, we introduce Binding Self-Distillation, which targets the missing access relation by transferring the

model's own name-conditioned access behavior to the SMILES route. It substantially improves factual retrieval directly from SMILES across seven models, without requiring the lexical name at inference time or training a model from scratch. A detailed discussion of related work is provided in Appendix A. In summary, our contributions are:

- We introduce **ChemIDBench**, a controlled benchmark for evaluating knowledge access across co-referring molecular identifiers, covering matched trivial names, IUPAC names, and SMILES and evaluating seven LLMs spanning 7B–72B parameters.
- We characterize how molecular identifier choice affects access to different types of knowledge, contrasting entity-associated factual queries with structure-computable queries and testing whether successful molecular recognition is sufficient for factual retrieval.
- Through lexical-bridge controls, conflicting-identifier experiments, representation analysis, and activation patching, we show that entity recognition, factual knowledge, and identifier-conditioned access are separable, providing behavioral and causal evidence for *cross-identifier binding failure*.
- We show that standard knowledge editing does not reliably propagate factual access across co-referring identifiers, revealing that modifying the factual association alone is insufficient. Motivated by this diagnosis, we introduce Binding Self-Distillation, which directly targets the missing access relation and substantially improves retrieval from SMILES without requiring the lexical name at inference time.

2 Benchmarking Cross-Identifier Knowledge Access

2.1 Benchmark Design

Our benchmark **ChemIDBench** isolates the effect of molecular identifier choice while holding the underlying molecule and queried property fixed. Each molecule is represented by three co-referring molecular identifiers: a trivial name, an IUPAC name, and a canonical SMILES string. We select these three because they are widely used across chemical and biomedical workflows (Hastings et al., 2016; Gómez-Bombarelli et al., 2018; Schwaller et al., 2019) while providing complementary interfaces to the same molecular entity: trivial names serve as familiar lexical references, IUPAC names provide systematic chemical nomenclature (Favre & Powell, 2013), and SMILES provides a compact structure-based representation (Gómez-Bombarelli et al., 2018). We begin with 3,025 molecules with approved drug–indication records in ChEMBL (Mendez et al., 2019; Hunter et al., 2025), using the ChEMBL preferred name as the trivial-name identifier. We then apply a strict entity-cleaning pipeline, removing salts, multi-fragment molecules, duplicate names and structures, invalid structures, and extreme molecule sizes, leaving 1,553 unambiguous molecular entities. We further retain only molecules with scorable therapeutic indications, yielding 897 candidates, and resolve their IUPAC names through PubChem (Kim et al., 2023). This produces 896 molecules with matched trivial-name, IUPAC-name, and SMILES representations. Full filtering criteria, indication normalization, and identifier-resolution checks are provided in Appendix B.1.

For each task, we query the same molecule independently through its trivial name, IUPAC name, and SMILES. The benchmark spans two complementary types of queries: (1) *Entity-associated queries* test access to information tied to the molecular entity, including molecule identification and therapeutic indication; (2) *Structure-computable queries* test properties that can be derived from molecular structure, including molecular formula, benzene-ring presence, nitrogen-atom count, ring count, and heavy-atom count, with ground truth computed using RDKit (Landrum, 2026). Models use greedy decoding; entity-associated outputs are scored against normalized accepted answers, while structure-computable properties are evaluated against RDKit-derived ground truth. Exact task definitions, prompts, ground-truth construction, decoding settings, and scoring procedures are provided in Appendix B.2.

We evaluate seven LLMs spanning 7B–72B parameters, including both chemistry-specialized and general-purpose models. To isolate the effect of chemistry-specific training on cross-identifier behavior, we compare ChemDFM-v2.0-14B and ChemDFM-v1.5-8B (Zhao et al., 2025) with Qwen2.5-14B-Instruct (Yang et al., 2024b) and Meta-Llama-3-8B-Instruct (Grattafiori et al., 2024), respectively. We further include Qwen2.5-7B-Instruct (Yang et al., 2024b), Llama-3.1-8B-Instruct (Grattafiori et al., 2024), and Qwen2-72B-Instruct (Yang et al., 2024a).

Table 1: Accuracy (%) under trivial name (N), IUPAC name (I), and SMILES (S). *Indication* denotes therapeutic-indication retrieval; *Benzene* denotes benzene-ring detection; and *Nitrogens* denotes nitrogen-atom counting. These three tasks provide the primary evidence for the cross-identifier dissociation; results for all seven benchmark tasks are reported in Appendix B.3. Best performance within each model and task is shown in bold.

	Indication			**Benzene**			**Nitrogens**		
Model	N	I	S	N	I	S	N	I	S
ChemDFM-v2.0-14B	**38.8**	2.2	2.1	76.7	93.2	**94.6**	41.6	**70.3**	59.6
ChemDFM-v1.5-8B	**22.5**	0.1	3.7	70.5	81.1	**84.4**	5.5	8.7	**21.2**
Qwen2.5-14B-Instruct	**37.8**	5.4	1.2	58.6	80.9	**86.9**	33.3	59.2	**65.6**
Qwen2.5-7B-Instruct	**34.2**	2.1	1.2	57.9	60.8	**80.1**	32.9	**53.0**	38.3
Qwen2-72B-Instruct	**43.5**	9.5	0.9	71.0	**88.6**	84.3	40.0	51.8	**61.6**
Meta-Llama-3-8B-Instruct	**38.6**	6.6	0.9	67.7	89.4	**89.7**	29.0	**57.4**	47.0
Llama-3.1-8B-Instruct	**39.1**	6.0	1.2	65.5	**90.5**	90.4	25.7	**49.9**	38.4

2.2 Equivalent Identifiers Expose Different Knowledge

If co-referring identifiers were interchangeable interfaces to model knowledge, changing only how a molecule is represented should have a negligible effect on the model's answer. Table 1 shows the opposite: changing the identifier can nearly erase access to entity-associated knowledge while improving access to structure-computable properties. For ChemDFM-v2.0-14B, therapeutic-indication accuracy collapses from 38.8% under the trivial name to 2.1% under SMILES. Yet on the same molecules, the direction reverses for benzene-ring detection: accuracy rises from 76.7% to 94.6%. This reversal holds across all seven models, spanning 7B to 72B parameters and including both chemistry-specialized and general-purpose LLMs. Across these models, trivial names are strongest for therapeutic indication, whereas SMILES outperforms the trivial name on benzene detection by 13.3–28.3 percentage points. Nitrogen counting shows the same broad reversal, though the better structural representation, IUPAC or SMILES, varies across models. Thus, structure-descriptive identifiers are not simply harder inputs: they can be better interfaces when the requested information is recoverable from molecular form.

Because each molecule is evaluated under every identifier, these differences can be tested molecule by molecule rather than only through aggregate accuracy. Paired McNemar tests (McNemar, 1947) confirm both directions: trivial names outperform SMILES for therapeutic-indication retrieval, whereas SMILES outperforms trivial names for benzene detection, with both effects significant across all seven models ($p < 10^{-7}$). Exact paired counts, p-values, and results for all seven benchmark tasks are reported in Appendix B.3. The key result is therefore not that one identifier is generally easier than another. Instead, the identifier determines *which knowledge is accessible*: lexical names favor entity-associated knowledge, while structure-descriptive identifiers favor information recoverable from molecular form. This raises a key question: is the SMILES failure simply a failure of molecular recognition?

2.3 Recognition Does Not Imply Retrieval

To test whether failed molecular recognition explains the retrieval gap, we evaluate two complementary recognition criteria. First, we use a two-alternative forced-choice (2AFC) test, a standard forced-choice paradigm in which the model must select the correct answer from two alternatives. Given a SMILES string, the model chooses between the correct trivial name and one incorrect candidate; we retain only molecules answered correctly across three independent decoy conditions. Second, in the stricter free-form test, the model must generate the correct trivial name directly from SMILES without candidate answers. Full recognition protocols and subset-construction details are provided in Appendix B.4.

As shown in Table 2, under 2AFC, ChemDFM-v2.0-14B correctly recognizes 576 molecules, yet therapeutic-indication accuracy through SMILES is only 4.0%, close to 2.8% on the remaining molecules, while the same recognized molecules reach 42.7% through their trivial names. Free-form recognition gives the same result: among the 62 molecules whose trivial names are correctly generated from SMILES, indication accuracy is 40.3% through the name but only 3.2% through SMILES, compared with 3.6% outside the recognized subset. ChemDFM-v1.5-8B shows the same

Table 2: Therapeutic-indication accuracy (%) on molecules whose SMILES–name correspondence is recognized. "Rest" denotes SMILES accuracy on molecules outside the recognized subset. Higher accuracy indicates better performance; n denotes the size of the recognized subset.

	2AFC recognition				Free-form recognition			
Model	n	SMILES	Name	Rest	n	SMILES	Name	Rest
ChemDFM-v2.0-14B	576	4.0	42.7	2.8	62	3.2	40.3	3.6
ChemDFM-v1.5-8B	487	3.7	22.0	4.9	72	8.3	19.4	3.9
Qwen2.5-14B-Instruct	332	1.2	44.9	1.4	1	0.0	100.0	1.3
Qwen2.5-7B-Instruct	240	0.0	40.4	1.7	1	0.0	100.0	1.2
Qwen2-72B-Instruct	376	1.3	42.6	0.6	7	0.0	42.9	0.9
Llama-3.1-8B-Instruct	312	1.6	42.6	1.4	6	0.0	33.3	1.5

Table 3: Supplying lexical identity restores factual access. Therapeutic-indication accuracy (%). Higher values indicate better performance.

Model	SMILES	+ Name	Name	Recovery
ChemDFM-v2.0-14B	2.1	36.3	38.6	93.6
ChemDFM-v1.5-8B	3.9	21.4	23.0	91.8
Qwen2.5-14B-Instruct	1.3	37.6	37.8	99.4
Qwen2.5-7B-Instruct	1.2	32.8	33.9	96.6
Llama-3.1-8B-Instruct	1.2	38.6	39.0	99.1
Qwen2-72B-Instruct	0.8	42.5	43.5	97.7

broader separation: under 2AFC, the recognized molecules reach 22.0% through their trivial names but only 3.7% through SMILES. The broader 2AFC pattern also holds across the other models with recognized subsets. Recognition therefore does not explain the retrieval gap: a model can map a structural identifier to the correct lexical identity while access to entity-associated knowledge remains strongly identifier-dependent. We refer to this phenomenon as cross-identifier binding failure.

3 Knowledge Is Present but Unreachable

Recognition alone does not restore factual retrieval (Section 2.3). We therefore ask whether the missing knowledge is absent from the model or instead inaccessible through the structural identifier.

3.1 Lexical Identity Restores Factual Access

We supply the missing identifier correspondence by augmenting each SMILES with its correct trivial name, without providing any information about the queried indication. As shown in Table 3, this simple intervention nearly closes the gap to name-only retrieval across all models, recovering 91.8–99.4% of the lost accuracy. For ChemDFM-v2.0-14B, indication accuracy rises from 2.1% with SMILES alone to 36.3% with SMILES+Name, compared with 38.6% from the name alone. Qwen2.5-14B similarly rises from 1.3% to 37.6%, versus 37.8% from the name. Thus, supplying lexical identity is sufficient to make much of the otherwise inaccessible factual knowledge reachable. Exact prompts, the recovery definition, and reverse-direction controls are provided in Appendix C.1.

3.2 Conflicting Identifiers Expose Separate Access Routes

To test whether lexical and structural identifiers access a single coherent entity representation, we pair the SMILES of molecule A with the incorrect trivial name of molecule B. The resulting answers split by query type: identity and therapeutic indication largely follow the supplied name B, whereas benzene-ring remains close to those obtained from A's SMILES alone. As shown in Table 4, for ChemDFM-v2.0-14B, the conflicting input produces B's identity in 85.5% of cases and retains 32.3% indication accuracy for B, while benzene accuracy for A remains 95.3%, compared with 94.5% from its SMILES alone. Full results are provided in Appendix C.2.

The conflict is often undetected. We next ask whether models recognize that the two supplied identifiers are incompatible. When explicitly asked whether a SMILES and trivial name refer to the same molecule, models reject obvious mismatches much more reliably than structurally similar ones. Across the tested models, only 17.5–43.1% of similar incorrect names are rejected; even in a two-choice identification test, accuracy is only 56.4–73.9%. Thus, conflicting identifiers can drive different answers without the model reliably detecting the inconsistency. Full prompts, decoy construction, and discrimination results are reported in Appendix C.3.

Table 4: Conflicting identifiers control different outputs: Each conflicting prompt pairs the SMILES of molecule A with the incorrect trivial name of molecule B. For identity, we report how often the model returns the true molecule A or the supplied decoy B. For therapeutic indication, we report accuracy for B under the conflicting prompt and when B is queried by its name alone. For benzene-ring presence, we report accuracy for A under the conflicting prompt and when A is queried by its SMILES alone. All values are percentages. These quantities are diagnostic and do not have a single preferred direction.

	Identity under conflict		Indication of B		Benzene of A	
Model	True A	Decoy B	Conflict	Name only	Conflict	SMILES only
ChemDFM-v2.0-14B	1.0	85.5	32.3	38.6	95.3	94.5
ChemDFM-v1.5-8B	0.6	73.4	17.4	23.0	80.5	85.0
Qwen2.5-14B-Instruct	0.0	81.1	35.9	37.8	84.9	86.9
Qwen2.5-7B-Instruct	0.0	60.4	32.1	33.9	75.9	80.2
Llama-3.1-8B-Instruct	0.0	86.8	37.3	39.0	89.3	90.3
Qwen2-72B-Instruct	0.0	77.8	43.2	44.1	83.9	84.3

Table 5: Lexical activations causally restore retrieval from SMILES. Peak recovery (%) of the name–SMILES log-probability gap and the corresponding relative layer depth. Control patches use a different molecule with a different gold answer. Higher recovery indicates stronger restoration; peak depth is descriptive, while the control is expected not to yield positive recovery.

Model	Peak depth	Recovery	Control
ChemDFM-v2.0-14B	0.71	105.4	-88.3
ChemDFM-v1.5-8B	0.50	76.4	-79.6
Qwen2.5-14B-Instruct	0.71	106.0	-197.7
Qwen2.5-7B-Instruct	0.75	104.9	-111.4
Meta-Llama-3-8B-Instruct	0.52	135.0	-179.9
Llama-3.1-8B-Instruct	0.50	112.4	-208.2
Qwen2-72B-Instruct	0.83	138.4	-30.9

4 Diagnosing and Restoring Cross-Identifier Access

4.1 Causal Analysis via Activation Patching

The behavioral results suggest that factual knowledge is accessible through a molecule's trivial name but not through its SMILES. We test whether this difference can be causally traced to the internal state induced by the identifier. We use *activation patching* (Zhang & Nanda, 2024), a causal intervention that replaces an internal activation in a target computation with the corresponding activation from a source computation, to test whether the lexical state is sufficient to restore factual retrieval. For matched prompts querying the same fact, we treat the trivial-name run as the source and the SMILES run as the target. At each tested layer, we replace the residual-stream state at the final prompt token of the SMILES run with the corresponding source state. We measure recovery as the fraction of the name–SMILES teacher-forced log-probability gap closed by the intervention; a different-molecule activation with a different gold answer serves as the control. Full prompts, scoring, and implementation details are provided in Appendix D.

As shown in Table 5, the correct lexical state substantially restores factual retrieval in every model, with peak recovery ranging from 76.4% to 138.4%. In contrast, every different-molecule control is negative. The effect peaks at relative depths 0.50–0.83, rather than at a single architecture-specific layer. These interventions provide causal evidence that the lexical route produces a retrieval-supporting state that the SMILES route fails to construct on its own. We next ask whether this access asymmetry is reflected in the geometry of the two identifier representations.

4.2 Probing Cross-Identifier Binding in Model Representations

We examine whether co-referring molecular identifiers become aligned at the level of individual entities. If two identifier representations encode the same molecular identity in a shared geometry, the representation of a molecule under one identifier should retrieve its co-referring representation under another identifier from a large candidate set. We therefore measure cross-identifier top-1 retrieval. For each of the 896 molecules, we extract representations from identifier-specific prompts and use the representation under one identifier to retrieve the corresponding SMILES representation from all 896 candidates. We evaluate retrieval across hidden-state depths and report the max-

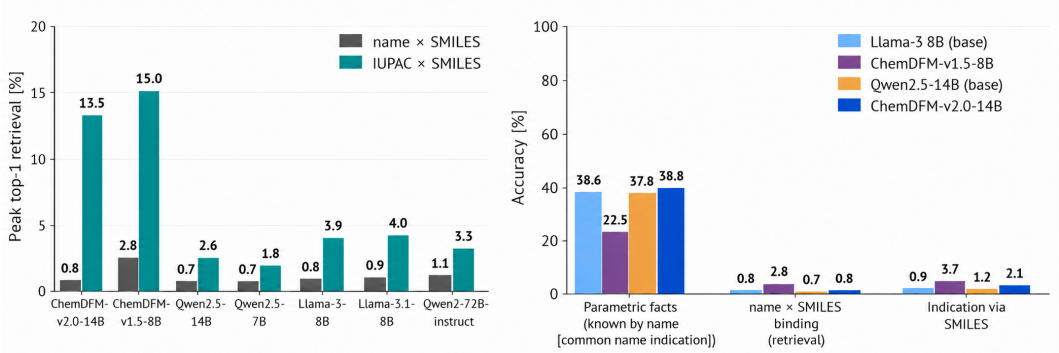


Figure 2: **Cross-identifier representation alignment and chemistry specialization. Left:** peak top-1 retrieval of the same molecule among 896 candidates across hidden-state depths; chance performance is 0.11%. Name–SMILES denotes trivial name → SMILES retrieval and IUPAC–SMILES denotes IUPAC name → SMILES retrieval. **Right:** comparison of the two general-purpose reference models with their chemistry-specialized counterparts. "Parametric facts known by name" denotes strict-scored trivial-name indication accuracy; "Name–SMILES binding" denotes the same cross-identifier retrieval metric; and "Indication via SMILES" denotes strict-scored SMILES-conditioned indication accuracy.

imum top-1 accuracy. Throughout this section, Name–SMILES denotes trivial name → SMILES retrieval, whereas IUPAC–SMILES denotes IUPAC name → SMILES retrieval. Full representation extraction, pooling, centering, normalization, and layer-wise evaluation details are provided in Appendix E. The result reveals a sharp asymmetry. Even at the best depth, Name–SMILES retrieval reaches only 0.7–2.8% across all models. Yet weak alignment is not inevitable across molecular identifiers: IUPAC–SMILES retrieval reaches 13.5% for ChemDFM-v2.0-14B and 15.0% for ChemDFM-v1.5-8B (Figure 2). The same models can therefore develop substantially stronger correspondence between two structure-descriptive forms while leaving lexical identity only weakly connected to molecular structure.

Chemistry specialization makes this separation especially clear. For the Meta-Llama-3-8B-Instruct → ChemDFM-v1.5-8B comparison, IUPAC–SMILES retrieval rises from 3.9% to 15.0%, whereas Name–SMILES retrieval rises only from 0.8% to 2.8%. For Qwen2.5-14B-Instruct → ChemDFM-v2.0-14B, IUPAC–SMILES retrieval rises from 2.6% to 13.5%, while Name–SMILES retrieval changes only from 0.7% to 0.8%. Thus, the chemistry-specialized models exhibit substantially stronger alignment between the two structure-descriptive identifiers than their general-purpose reference models without producing comparable binding between molecular structure and lexical identity. This representational asymmetry mirrors the behavioral access gap. Stronger chemical representations do not by themselves make entity-associated factual knowledge reliably accessible through SMILES: the models can substantially align IUPAC names with molecular structure while leaving the lexical route through which much of their factual knowledge is accessed only weakly connected to that same structure.

4.3 Lexical Routing Restores Factual Access from SMILES

The lexical bridge above suggests a simple inference mitigation: resolve a SMILES string to its lexical identity before asking for the fact. We test this over 891 molecules and 2,955 available molecule–relation pairs spanning therapeutic indication, molecular target, mechanism of action, and ATC class. For each query, we compare direct SMILES access, SMILES augmented with the molecule's ChEMBL preferred name obtained by database lookup, and the preferred name alone. The lookup supplies only the identifier correspondence, not the queried fact. Full construction, prompts, scoring, and controls are given in Appendix F.

As Table 6 shows, lexical routing closes 83.6–98.8% of the aggregate SMILES–name gap across all models. The model itself rarely supplies this bridge reliably: in a self-resolution control, SMILES-to-name identification succeeds for only 0.1–12.0% of molecules (Appendix F). Thus, most of the missing factual access can be recovered by supplying the identifier correspondence, but that correspondence is not reliably inferred from SMILES alone. Lookup-based routing therefore provides an effective inference workaround, but it requires lexical resolution on every query and remains

Table 6: Aggregate accuracy (%) over available molecule–relation pairs. Higher values indicate better performance.

Model	SMILES	+ Lookup	Name	Gap rec.
ChemDFM-v2.0-14B	2.4	20.8	24.4	83.6
ChemDFM-v1.5-8B	5.4	14.2	14.5	96.7
Qwen2.5-14B-Instruct	1.2	25.3	29.2	86.1
Qwen2.5-7B-Instruct	0.6	22.6	23.2	97.3
Meta-Llama-3-8B-Instruct	1.2	26.8	27.1	98.8
Llama-3.1-8B-Instruct	0.6	26.8	28.9	92.6

Table 7: All values are percentages. Specific transfer is computed as SMILES transfer minus leakage and is therefore reported in percentage points; brackets denote 95% bootstrap confidence intervals. Higher edit efficacy, specific transfer, and +Name recovery indicate better performance, while lower leakage is better. Raw SMILES transfer should be interpreted together with leakage.

Model	Method	Effic.	SMILES	Leak	Specific [95% CI]	+Name
ChemDFM-v2.0-14B	ROME	52.2	22.1	19.8	+2.3 [-0.9,5.5]	45.5
	FT	26.0	1.8	2.1	-0.3 [-1.6,1.2]	25.2
Qwen2.5-14B	ROME	17.9	4.4	4.9	-0.5 [-2.4,1.4]	12.3
	FT	29.0	11.5	8.0	**+3.5 [0.9,6.3]**	26.7
Qwen2.5-7B	ROME	47.9	4.0	7.4	-3.4 [-5.7,-0.9]	11.4
	FT	34.5	5.4	7.3	-1.9 [-4.4,0.8]	21.7
Llama-3.1-8B	ROME	44.2	21.1	8.1	**+13.0 [9.2,16.9]**	40.4
	FT	38.3	12.6	11.4	+1.2 [-1.6,4.1]	32.1

bounded by the knowledge accessible through the name. Can the same access instead be written into the model parameters, using standard knowledge editors as a diagnostic of what is missing?

4.4 Standard Editors Do Not Reliably Transfer Across Identifiers

Can the lookup required above be avoided by modifying the model weights? We test this diagnostically with ROME (Gupta et al., 2024b;a) and fine-tuning (FT) (Ouyang et al., 2022): a counterfactual drug indication is edited through the molecule's trivial name and then queried through its co-referring SMILES. Because an edit may also increase production of the target for unrelated molecules, we subtract the corresponding locality leakage, $T_{\text{specific}} = T_{\text{SMILES}} - T_{\text{leak}}$. Full edit construction, IUPAC results, layer selection, and hyperparameters are given in Appendix G. Across the eight model–editor configurations, six show no reliable entity-specific transfer to SMILES beyond the leakage baseline, while only two exhibit statistically reliable transfer beyond this baseline, including substantial transfer for Llama-3.1-8B-Instruct with ROME (Table 7). This pattern remains when we restrict evaluation to edits that demonstrably succeed through the trivial name (Appendix G), showing that weak transfer is not simply caused by failed edits. Moreover, restoring the correct name alongside SMILES often makes the edited association accessible again. Thus, standard editing can sometimes propagate across identifiers, but this behavior is highly inconsistent across models and methods: modifying the factual association does not reliably repair the structural route used to access it.

5 Binding Self-Distillation Transfers Lexical Access to SMILES

The editing results suggest that changing a factual association is not enough: the missing object is the *access route* from SMILES to knowledge that is already reachable through the molecule's lexical identity. We therefore ask whether this access behavior itself can be written into the model. Our goal is not to teach a new fact, but to make the SMILES route reproduce knowledge access that the same model already exhibits when the correct name is available. We introduce **Binding Self-Distillation** (BSD). For a molecule with SMILES s, trivial name n, and a generic continuation stem z, we construct matched teacher and student contexts,

$$x_T = [\text{SMILES}(s);\ \text{“known as } n\text{”}; z], \qquad x_S = [\text{SMILES}(s); z].$$

Teacher and student use the same pretrained model. The teacher is frozen and receives the lexical identity, while the student receives only SMILES and learns a lightweight LoRA adapter by minimizing $D_{\text{KL}}\big(p_\theta(\cdot \mid x_T) \parallel p_{\theta,\phi}(\cdot \mid x_S)\big)$. Thus, the training signal is the model's own name-conditioned predictive behavior: no factual labels, generated target answers, or external teacher are

Table 8: Facts/entity is the average number of available factual relations retrieved per molecule before and after the per-entity binding intervention. Higher Facts/entity indicates better performance; n is the model-specific number of molecules retained by the original BSD entity-selection pipeline.

Model	n	Facts/entity
ChemDFM-v2.0-14B	389	$0.05 \rightarrow 0.87$
Qwen2.5-14B-Instruct	390	$0.06 \rightarrow 0.73$
Llama-3.1-8B-Instruct	389	$0.03 \rightarrow 0.68$
ChemDFM-v1.5-8B	47	$0.51 \rightarrow 0.85$
Qwen2.5-7B-Instruct	351	$0.04 \rightarrow 0.85$
Meta-Llama-3-8B-Instruct	378	$0.02 \rightarrow 0.78$
Qwen2-72B-Instruct	435	$0.01 \rightarrow 0.85$

used. We train one adapter per molecule using generic continuation stems, with the downstream factual evaluation prompts excluded from training. At inference time, the name is removed and only SMILES is provided. Full training configuration, prompts, and controls are in Appendix H. Binding Self-Distillation increases factual access through SMILES across all seven models (Table 8). For ChemDFM-v2.0-14B, the number of retrieved facts rises from 0.05 to 0.87 per entity. The gain is not restricted to the indication relation used to select eligible molecules: SMILES-conditioned accuracy rises from 2.1% to 51.4% for indication, 0.0% to 27.6% for molecular target, 0.0% to 5.0% for mechanism of action, and 3.3% to 11.4% for ATC class (Appendix H.5, Table 26). The effect also persists at substantially larger scale: Qwen2-72B-Instruct rises from 0.01 to 0.85 facts per entity over 435 molecules. Because the factual evaluation prompts are excluded from training, these improvements reflect transfer of the lexical access behavior across prompts rather than direct training on the evaluation answers.

How much of the lexical route can be internalized? On the same ChemDFM-v2.0-14B entities, explicitly supplying the correct name raises access from 0.05 to 1.01 facts per entity, while Binding Self-Distillation reaches 0.87, recovering 86% of the available SMILES–name gap without requiring the name at inference time. For Qwen2-72B-Instruct, the corresponding values are 0.01, 1.34, and 0.85, yielding 63% recovery (Appendix H.6, Table 27). Retention controls for these two highlighted settings also show that the intervention need not destroy behavior that already works: benzene verdicts are preserved for 94.1% of ChemDFM-v2.0-14B examples and 92.9% of Qwen2-72B-Instruct examples, with additional locality analyses reported in Appendix H.7. Together, these results provide a constructive complement to the editing diagnostic in Section 4.4. Standard editors modify factual associations but propagate them inconsistently across identifiers; Binding Self-Distillation instead targets the missing access relation. The result shows that knowledge already exposed by a lexical identifier can be made substantially more accessible from SMILES without requiring that lexical identifier at inference time.

6 Conclusion

In chemistry pipelines, molecular identity is often treated as a solved interface problem: once two identifiers are known to denote the same molecule, an LLM is implicitly expected to access the same underlying knowledge. We introduce **ChemIDBench**, a controlled benchmark for evaluating knowledge access across molecular representations, and find that across LLMs spanning 7B to 72B, changing only the identifier can change which knowledge is reachable. Surprisingly, a model can identify a molecule from SMILES and retrieve a fact through its trivial name, yet fail to retrieve the same fact directly from SMILES. When identifiers conflict, different routes can also drive different answers within the same prompt, often without the model detecting the inconsistency. Recognition controls, conflict experiments, representation analysis, and activation patching together show that entity recognition, knowledge possession, and identifier-conditioned access are separable capabilities. We call this **cross-identifier binding failure**. Standard knowledge editors do not reliably restore access across identifiers; we therefore introduce **Binding Self-Distillation**, which directly targets the missing access relation and substantially improves factual retrieval from SMILES without requiring the lexical name at inference time. Our current formulation performs this repair per entity, showing that cross-identifier access is repairable while leaving generalization to unseen entities as a future direction. Beyond chemistry, the broader identifier-dependent-access phenomenon also appears in our protein-domain replication with opaque UniProt accessions (Ahmad et al., 2025) in Appendix I. Overall, our results show that for LLMs to be used in science, knowledge is only useful if it remains accessible across the different representations of the same entity.

AI USE STATEMENT

Generative AI tools were used to assist with code implementation and debugging, as well as language editing and presentation of the manuscript. The core research idea, experimental design, and analyses were developed by the authors. AI-assisted code was reviewed, tested, and validated by the authors before use in the experiments. The authors take full responsibility for the final content of the paper.

REPRODUCIBILITY STATEMENT

We provide detailed information needed to reproduce our experiments in the appendix, including benchmark construction, prompts, scoring procedures, experimental protocols, and implementation details.

REFERENCES

Shadab Ahmad, Leonardo Jose da Costa Gonzales, Emily H Bowler-Barnett, Daniel L Rice, Minjoon Kim, Supun Wijerathne, Aurélien Luciani, Swaathi Kandasaamy, Jie Luo, Xavier Watkins, et al. The uniprot website api: facilitating programmatic access to protein knowledge. *Nucleic acids research*, 53(W1):W547–W553, 2025.

Dávid Bajusz, Anita Rácz, and Károly Héberger. Why is tanimoto index an appropriate choice for fingerprint-based similarity calculations? *Journal of cheminformatics*, 7(1):20, 2015.

George Arthur Baker, Mario Sanz-Guerrero, and Katharina von der Wense. Molecular string representation preferences in pretrained LLMs: A comparative study in zero- & few-shot molecular property prediction. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 1071–1085, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.56. URL https://aclanthology.org/2025.emnlp-main.56/.

Debjyoti Bhattacharya, Harrison J Cassady, Michael A Hickner, and Wesley F Reinhart. Large language models as molecular design engines. *Journal of Chemical Information and Modeling*, 64(18):7086–7096, 2024.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Jaron Cohen, Alexander G Hasson, and Sara Tanovic. Unveiling latent knowledge in chemistry language models through sparse autoencoders. *arXiv preprint arXiv:2512.08077*, 2025.

Henri A Favre and Warren H Powell. *Nomenclature of organic chemistry: IUPAC recommendations and preferred names 2013*. Royal Society of Chemistry, 2013.

Rafael Gómez-Bombarelli, Jennifer N Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, Ryan P Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268, 2018.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Akshat Gupta, Anurag Rao, and Gopala Anumanchipalli. Model editing at scale leads to gradual and catastrophic forgetting. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 15202–15232, Bangkok, Thailand,

August 2024a. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.902. URL https://aclanthology.org/2024.findings-acl.902/.

Akshat Gupta, Dev Sajnani, and Gopala Anumanchipalli. A unified framework for model editing. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 15403–15418, Miami, Florida, USA, November 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.903. URL https://aclanthology.org/2024.findings-emnlp.903/.

Wanqing Hao, Zetong Xu, Zhiyuan Han, Xiaoyan Bai, and Chenhao Tan. Mechanistic interpretability of chemical language models: Molecular geometry and representations in molmformer and smi-ted. In *Mechanistic Interpretability Workshop at ICML 2026*, 2026.

Janna Hastings, Gareth Owen, Adriano Dekker, Marcus Ennis, Namrata Kale, Venkatesh Muthukrishnan, Steve Turner, Neil Swainston, Pedro Mendes, and Christoph Steinbeck. Chebi in 2016: Improved services and an expanding collection of metabolites. *Nucleic acids research*, 44(D1): D1214–D1219, 2016.

Cheng-Hsun Hsueh, Paul Kuo-Ming Huang, Tzu-Han Lin, Che Wei Liao, Hung-Chieh Fang, Chao-Wei Huang, and Yun-Nung Chen. Editing the mind of giants: An in-depth exploration of pitfalls of knowledge editing in large language models. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 9417–9429, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.550. URL https://aclanthology.org/2024.findings-emnlp.550/.

Fiona MI Hunter, Harris Ioannidis, A Patrícia Bento, Nicolas Bosc, Sybilla Corbett, Eloy Felix, M Paula Magarinos, Emma Manners, Ines A Smit, Marleen de Veij, et al. Drug and clinical candidate drug data in chembl. *Journal of medicinal chemistry*, 68(19):19800, 2025.

Kevin Maik Jablonka, Philippe Schwaller, Andres Ortega-Guerrero, and Berend Smit. Leveraging large language models for predictive chemistry. *Nature Machine Intelligence*, 6(2):161–169, 2024.

Yuxin Jiang, Yufei Wang, Chuhan Wu, Wanjun Zhong, Xingshan Zeng, Jiahui Gao, Liangyou Li, Xin Jiang, Lifeng Shang, Ruiming Tang, Qun Liu, and Wei Wang. Learning to edit: Aligning LLMs with knowledge editing. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4689–4705, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.258. URL https://aclanthology.org/2024.acl-long.258/.

Shreya Johri, Jaehwan Jeong, Benjamin A Tran, Daniel I Schlessinger, Shannon Wongvibulsin, Leandra A Barnes, Hong-Yu Zhou, Zhuo Ran Cai, Eliezer M Van Allen, David Kim, et al. An evaluation framework for clinical use of large language models in patient interaction tasks. *Nature medicine*, 31(1):77–86, 2025.

Christian Kenneth, Etowah Adams, Liam Bai, and Gerard JP van Westen. What does a chemical language model know about molecules? *arXiv preprint arXiv:2606.23443*, 2026.

Yosuke Kikuchi, Yasuhiro Yoshikai, Shumpei Nemoto, Ayako Furuhama, Takashi Yamada, Hiroyuki Kusuhara, and Tadahaya Mizuno. Notation-level confounding: When inconsistent molecular notations mislead chemical language models. *arXiv preprint arXiv:2505.07139*, 2025.

Sunghwan Kim, Jie Chen, Tiejun Cheng, Asta Gindulyte, Jia He, Siqian He, Qingliang Li, Benjamin A Shoemaker, Paul A Thiessen, Bo Yu, et al. Pubchem 2023 update. *Nucleic acids research*, 51(D1):D1373–D1380, 2023.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022.

Greg Landrum. Rdkit: Open-source cheminformatics, 2026. URL https://www.rdkit.org. Version 2025.03.6.

Zhenyu Lei, Patrick Soga, Yaochen Zhu, Yinhan He, Yushun Dong, and Jundong Li. Moledit: Knowledge editing for multimodal molecule language models. In *Proceedings of the Nineteenth ACM International Conference on Web Search and Data Mining*, pp. 334–344, 2026.

Zhuoran Li, Xu Sun, Wanyu Lin, and Chang Wen Chen. Controllable molecule generation via sparse representation editing: An interpretability-driven perspective. 2026.

Daniel McDuff, Mike Schaekermann, Tao Tu, Anil Palepu, Amy Wang, Jake Garrison, Karan Singhal, Yash Sharma, Shekoofeh Azizi, Kavita Kulkarni, et al. Towards accurate differential diagnosis with large language models. *Nature*, 642(8067):451–457, 2025.

Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947.

David Mendez, Anna Gaulton, A Patrícia Bento, Jon Chambers, Marleen De Veij, Eloy Félix, María Paula Magariños, Juan F Mosquera, Prudence Mutowo, Michał Nowotka, et al. Chembl: towards direct deposition of bioassay data. *Nucleic acids research*, 47(D1):D930–D940, 2019.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.

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

Philippe Schwaller, Teodoro Laino, Théophile Gaudin, Peter Bolgar, Christopher A Hunter, Costas Bekas, and Alpha A Lee. Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction. *ACS central science*, 5(9):1572, 2019.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180, 2023.

Harold Stanislaw and Natasha Todorov. Calculation of signal detection theory measures. *Behavior research methods, instruments, & computers*, 31(1):137–149, 1999.

Kristof Varadi, Mark Marosi, and Peter Antal. Circuits, features, and heuristics in molecular transformers. *arXiv preprint arXiv:2512.09757*, 2025.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Bing Yan, Angelica Chen, and Kyunghyun Cho. Inconsistency of llms in molecular representations. *Digital Discovery*, 4(10):2876–2892, 2025.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024a.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024b.

Di Zhang, Wei Liu, Qian Tan, Jingdan Chen, Hang Yan, Yuliang Yan, Jiatong Li, Weiran Huang, Xiangyu Yue, Wanli Ouyang, et al. Chemllm: A chemical large language model. *arXiv preprint arXiv:2402.06852*, 2024a.

Fred Zhang and Neel Nanda. Towards best practices of activation patching in language models: Metrics and methods. In *International Conference on Learning Representations*, volume 2024, pp. 1651–1678, 2024.

Mingxu Zhang, Yuhan Li, Lujundong Li, Dazhong Shen, Hui Xiong, and Ying Sun. Slim: Sparse latent steering for interpretable and property-directed llm-based molecular editing. *arXiv preprint arXiv:2605.10831*, 2026.

Yu Zhang, Xiusi Chen, Bowen Jin, Sheng Wang, Shuiwang Ji, Wei Wang, and Jiawei Han. A comprehensive survey of scientific large language models and their applications in scientific discovery. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pp. 8783–8817, 2024b.

Zihan Zhao, Da Ma, Lu Chen, Liangtai Sun, Zihao Li, Yi Xia, Bo Chen, Hongshen Xu, Zichen Zhu, Su Zhu, et al. Developing chemdfm as a large language foundation model for chemistry. *Cell Reports Physical Science*, 6(4), 2025.

A Related Work

LLMs and molecular representations in Chemistry. LLMs have increasingly been adapted to chemistry through domain-specific pretraining and instruction tuning. ChemLLM (Zhang et al., 2024a) and ChemDFM (Zhao et al., 2025), for example, train language models on chemical knowledge and tasks that include textual molecular representations such as SMILES (Gómez-Bombarelli et al., 2018; Schwaller et al., 2019). In practice, the same molecule may be presented to these models through a trivial (Gómez-Bombarelli et al., 2018; Schwaller et al., 2019), a systematic IUPAC name (Gómez-Bombarelli et al., 2018; Schwaller et al., 2019), or a SMILES string (Gómez-Bombarelli et al., 2018; Schwaller et al., 2019). Our work asks whether these co-referring representations are interchangeable not simply as inputs for downstream task performance, but as access routes to knowledge already stored in the model.

Representation dependence and consistency in chemistry. Recent work has shown that molecular representation can substantially affect LLM behavior. Baker et al. (2025) compare multiple molecular representations for property prediction and find strong representation preferences across pretrained LLMs. Yan et al. (2025) study cross-representation consistency between chemically equivalent SMILES and IUPAC inputs, showing that equivalent representations can produce inconsistent predictions and that consistency regularization only partially reduces this gap. Even within SMILES, alternative valid encodings of the same molecule can change learned representations and downstream performance (Kikuchi et al., 2025). These studies establish that chemically equivalent representations are not behaviorally interchangeable. We ask a different question: whether co-referring identifiers provide access to the *same knowledge already present in the model*. We distinguish failed recognition from failed retrieval, separate entity-associated facts from structure-computable properties, and examine what happens when two individually valid but mutually inconsistent identifiers are presented together.

Mechanistic analysis of molecular and factual representations. Recent mechanistic work has examined what chemical language models encode in their internal representations (Cohen et al., 2025; Varadi et al., 2025; Kenneth et al., 2026; Li et al., 2026; Zhang et al., 2026). Sparse-autoencoder analyses have identified latent features associated with molecular syntax, structural motifs, physicochemical properties, and pharmacological concepts (Cohen et al., 2025; Kenneth et al., 2026), while layerwise probing and causal interventions have been used to distinguish information that is merely decodable from information that is functionally used by molecular models (Hao et al., 2026; Varadi et al., 2025). These studies primarily ask what molecular structure and properties are represented inside models trained on molecular strings. Our question is different: why can knowledge about the same entity be accessible through one co-referring identifier but not another? We therefore study not only whether information is represented, but whether identifier-specific internal states provide effective access to the same factual knowledge.

Knowledge editing. Knowledge editing methods aim to modify selected factual associations while preserving unrelated model behavior. ROME and fine-tuning (Gupta et al., 2024b;a) provide representative parameter-based approaches, while subsequent work has broadened evaluation beyond edit success and paraphrase generalization. Recent evaluations (Hsueh et al., 2024; Jiang et al., 2024) further test whether an edit propagates to logically related facts. In chemistry, MolEdit (Lei et al., 2026) considers knowledge editing in multimodal molecule–language models and evaluates reliability, locality, and generality for molecule–text generation tasks. We introduce a different generalization requirement: *cross-identifier transfer*. If two identifiers denote the same entity, an edit made through one identifier should remain accessible when that entity is queried through another. We find that standard editors do not reliably satisfy this requirement for trivial-name–to–SMILES transfer, motivating cross-identifier binding as a distinct intervention target from conventional edit efficacy, paraphrase generalization, or cross-modal editing.

B Benchmarking Cross-Identifier Knowledge Access: Additional Details

B.1 Benchmark Design

Initial ChEMBL retrieval. We select ChEMBL (Mendez et al., 2019) drug-indication records corresponding to approved indications (phase 4) and group them by molecule. For each molecule,

we retain its ChEMBL preferred name, canonical SMILES, and all qualifying MeSH indication headings. This produces 3,025 unique molecules before subsequent entity and structure filtering.

Entity and structure filtering. Because our benchmark compares different identifiers for the *same* molecular entity, we first restrict the collection to molecules with well-defined lexical and structural representations. We retain only candidates with both a ChEMBL preferred name and canonical SMILES, and restrict preferred names to at most two whitespace-separated words to exclude long, formulation-like labels. Preferred names are stripped and lowercased before filtering. To focus on single molecular entities rather than salts, counterions, or mixtures, we remove salt and counterion forms by exact whitespace-token matching against a fixed manually specified lexicon,[1] and exclude multi-fragment structures whose ChEMBL canonical SMILES contains ".".

We further remove duplicate preferred names and structures so that an entity is not represented multiple times in the benchmark. Duplicate preferred names are identified by exact equality after lowercasing, retaining the first molecule in sorted ChEMBL-ID order. Remaining SMILES are parsed with RDKit (Landrum, 2026), ensuring that structure-derived ground truth can be computed reliably, and we restrict the parsed molecular graphs to 6–70 atoms to bound the benchmark to a consistent molecular-size range. For structure deduplication, each parsed molecule is canonicalized as an isomeric SMILES and duplicate structures are removed by exact equality of the resulting strings. These filters leave 1,553 candidate molecules.

Therapeutic-indication target construction. For each retained molecule, we collect all MeSH headings associated with phase-4 drug indications in ChEMBL. To make answer matching robust to superficial formatting differences, we normalize each heading by lowercasing, replacing non-alphanumeric characters with spaces, and collapsing repeated whitespace. For headings containing commas, we additionally include a comma-inverted variant to accommodate alternative word orderings of the same MeSH heading. Duplicate variants are removed, and strings of length at most three characters are discarded to avoid extremely short, non-discriminative matches.

We then compute document frequency over the accepted aliases at the molecule level, counting each alias at most once per molecule, and remove aliases occurring in more than 2% of the 1,553 candidates. This excludes overly generic indication terms that provide little molecule-specific signal. Because a molecule may have multiple qualifying MeSH headings, we define a single deterministic reference indication as the lexicographically first unique original heading; this is only a reproducible tie-breaking rule and does not imply clinical priority. A molecule is retained only if at least one accepted indication alias remains and its reference indication is still scorable under the resulting alias set, ensuring that every retained molecule has a valid target for the indication task. This filtering leaves 897 molecules.

IUPAC-name resolution. For each of the 897 remaining molecules, we retrieve an IUPAC name from PubChem (Kim et al., 2023) using the PubChem PUG REST interface. We first query PubChem by the ChEMBL preferred name and take the first returned IUPAC name. If this lookup does not yield an IUPAC name, we query PubChem for compound identifiers associated with the preferred name, select the first returned compound, and retrieve its IUPAC name. Only if both name-based procedures fail do we query PubChem using the molecule's canonical SMILES. PubChem mappings are additionally validated against the corresponding ChEMBL structures using the InChIKey connectivity block. An IUPAC name is resolved for 896 of the 897 candidates; the single unresolved molecule is discarded, yielding a final benchmark of 896 molecules.

B.2 Evaluation Protocol

Prompt construction. Each molecule is evaluated independently under three identifier conditions: trivial name, IUPAC name, and canonical SMILES. For a given molecule and task, the queried property and target answer are held fixed; only the identifier used to refer to the molecule changes.

[1] The lexicon contains `sulfate`, `sulphate`, `hydrochloride`, `chloride`, `sodium`, `potassium`, `calcium`, `magnesium`, `acetate`, `citrate`, `tartrate`, `maleate`, `mesylate`, `besylate`, `fumarate`, `succinate`, `phosphate`, `nitrate`, `bromide`, `iodide`, `carbonate`, `gluconate`, `lactate`, `oxalate`, `salicylate`, `stearate`, `tosylate`, `dihydrate`, `monohydrate`, `hydrate`, `hydrobromide`, `bitartrate`, `pamoate`, and `valerate`.

Table 9: **Benchmark construction.** Successive filtering reduces the initial ChEMBL collection to 896 molecules with trivial-name, IUPAC-name, and SMILES identifier fields and scorable therapeutic-indication annotations.

Construction stage	# Molecules
ChEMBL molecules with qualifying phase-4 indications	3,025
Entity and structure filtering	1,553
Therapeutic-indication normalization and filtering	897
Successful PubChem IUPAC-name resolution	896

Table 10: **Identifier-specific prompt prefixes.** Only the molecular identifier changes across the three conditions.

Identifier	Prompt prefix
Trivial name	Consider the molecule commonly known as "{NAME}".
IUPAC name	Consider the molecule named {IUPAC}.
SMILES	Consider the molecule with SMILES {SMILES}.

Table 10 shows the identifier-specific prefixes. The same task instruction is appended under each identifier condition, yielding seven tasks per identifier and 21 prompts per molecule. Across 896 molecules, this results in 18,816 prompts per model.

Benchmark tasks. The benchmark contains two entity-associated tasks and five structure-computable tasks. IDENTIFY asks the model to recover the molecule's trivial name, while INDICATION asks for a therapeutic indication associated with the molecule. The remaining tasks ask for molecular formula, benzene-ring presence, nitrogen-atom count, ring count, and heavy-atom count. Exact task instructions are provided in Table 11.

Ground-truth construction. For IDENTIFY, the ChEMBL preferred name is used as the gold answer. For INDICATION, predictions are evaluated against the strict set of accepted MeSH-derived aliases constructed as described in Appendix B.1. Ground truth for the five structure-computable tasks is derived from the canonical SMILES using RDKit (Landrum, 2026). Molecular formula is calculated from the parsed molecular structure. Benzene-ring presence is determined by testing for an aromatic six-membered carbon ring. Nitrogen count is the number of nitrogen atoms in the molecule, ring count is the number of rings returned by RDKit's ring-information representation, and heavy-atom count is the number of non-hydrogen atoms.

Decoding. All models are evaluated using greedy decoding with a maximum of 40 generated tokens per query. Sampling is disabled, and input sequences are truncated to at most 1,536 tokens. The same decoding procedure is used across identifier forms and benchmark tasks.

Entity-associated scoring. For IDENTIFY, generated responses and ChEMBL preferred names are normalized before matching. Normalization lowercases text, removes possessive suffixes and punctuation, collapses repeated whitespace, and applies a light plural normalization that removes a terminal "s" from tokens longer than four characters unless they end in "ss". A response is counted as correct when the normalized preferred name occurs as a complete word-boundary-matched phrase in the normalized response. For INDICATION, empty responses and refusals are rejected, as are responses containing more than 12 normalized tokens. Remaining responses are normalized using the same text-processing procedure and counted as correct when any alias in the strict accepted-answer set occurs as a complete word-boundary-matched phrase in the normalized response.

Structure-computable scoring. For FORMULA, generated responses are parsed into element-count compositions after normalizing formatting variations including subscripts, letter case, and punctuation. The scorer tests up to the first four formula-like candidates in the response and, if none matches, parses the complete response. A prediction is counted as correct when the resulting element counts exactly match the RDKit-derived gold formula.

Table 11: **Benchmark task prompts.** Each task instruction is appended to one of the three identifier-specific prefixes in Table 10.

Task	Prompt
IDENTIFY	What is the common generic name of this molecule? Answer in English. Reply with the name only.
INDICATION	What medical condition is this substance primarily used to treat? Answer in English. Reply with the condition only. Do not explain.
FORMULA	What is the molecular formula of this molecule? Answer in English. Reply with the formula only, e.g. C9H8O4. Do not explain.
BENZENE	Does this molecule contain a benzene ring? Answer in English. Reply with exactly one word: yes or no. Do not explain.
NITROGENS	How many nitrogen atoms does this molecule contain? Answer in English. Reply with a single integer only. Do not explain.
RING COUNT	How many rings does this molecule contain? Answer in English. Reply with a single integer only. Do not explain.
HEAVY ATOMS	How many non-hydrogen atoms does this molecule contain? Answer in English. Reply with a single integer only. Do not explain.

For the integer-valued tasks, NITROGENS, RING COUNT, and HEAVY ATOMS, the first integer appearing in the generated response is extracted and compared with the corresponding RDKit-derived gold value. A response containing no integer is counted as incorrect.

For BENZENE, responses are interpreted using a negation-first yes/no rule: explicit negative cues are evaluated before positive cues, ensuring that statements such as "does not contain" are classified as negative. Empty, refusal, or otherwise unparseable responses are counted as incorrect.

B.3 CROSS-IDENTIFIER ADDITIONAL RESULTS AND STATISTICAL TESTS

This section reports the complete cross-identifier results and paired statistical tests supporting Section 2.2. Benchmark construction and evaluation procedures are described in Appendices B.1 and B.2, respectively.

Primary cross-identifier results. Table 12 reports the four tasks most directly used in our analysis. Therapeutic-indication retrieval consistently favors the trivial name, whereas benzene-ring detection and nitrogen counting favor structure-descriptive identifiers across models. Identification measures whether the model can recover the trivial name from IUPAC or SMILES; the trivial-name condition serves as a ceiling control because the requested name is already present in the prompt.

Additional structure-computable tasks. The benchmark additionally includes molecular-formula prediction, total ring count, and heavy-atom count. These tasks do not exhibit the stable identifier ordering observed for benzene detection and nitrogen counting (Table 13). Formula prediction strongly favors IUPAC and SMILES for ChemDFM-v2.0-14B, but the trivial name performs best for most other models. Ring-count performance similarly varies in both magnitude and preferred identifier, while heavy-atom accuracy is generally low. We therefore report these tasks for completeness.

Paired significance testing. Because every molecule is evaluated under each identifier condition, the observations are paired. We use McNemar's test (McNemar, 1947) to determine whether the directional differences between identifier conditions occur systematically across molecules. For two identifier conditions A and B, let n_{10} denote the number of molecules answered correctly under A but incorrectly under B, and n_{01} the number answered incorrectly under A but correctly under B. Under the null hypothesis of no directional difference, the two types of discordant pair are expected to occur equally often. We use the continuity-corrected statistic:

$$\chi^2 = \frac{(|n_{10} - n_{01}| - 1)^2}{n_{10} + n_{01}},$$

Table 12: **Primary cross-identifier results.** Accuracy (%) when the underlying molecule and queried property are held fixed and only the supplied identifier changes. *Identify* denotes trivial-name recovery, *Therapeutic indication* denotes indication retrieval, *Benzene* denotes benzene-ring detection, and *Nitrogens* denotes nitrogen-atom counting. Best performance within each model and task is shown in bold.

Model	Identifier	Identify	Therapeutic indication	Benzene	Nitrogens
ChemDFM-v2.0-14B	Trivial name	**93.3**	**38.8**	76.7	41.6
	IUPAC name	3.8	2.2	93.2	**70.3**
	SMILES	6.9	2.1	**94.6**	59.6
ChemDFM-v1.5-8B	Trivial name	**72.7**	**22.5**	70.5	5.5
	IUPAC name	2.7	0.1	81.1	8.7
	SMILES	8.0	3.7	**84.4**	**21.2**
Qwen2.5-14B-Instruct	Trivial name	**85.4**	**37.8**	58.6	33.3
	IUPAC name	2.9	5.4	80.9	59.2
	SMILES	0.1	1.2	**86.9**	**65.6**
Qwen2.5-7B-Instruct	Trivial name	**41.3**	**34.2**	57.9	32.9
	IUPAC name	1.8	2.1	60.8	**53.0**
	SMILES	0.1	1.2	**80.1**	38.3
Qwen2-72B-Instruct	Trivial name	**62.1**	**43.5**	71.0	40.0
	IUPAC name	5.2	9.5	**88.6**	51.8
	SMILES	0.8	0.9	84.3	**61.6**
Meta-Llama-3-8B-Instruct	Trivial name	**31.4**	**38.6**	67.7	29.0
	IUPAC name	3.1	6.6	89.4	**57.4**
	SMILES	0.6	0.9	**89.7**	47.0
Llama-3.1-8B-Instruct	Trivial name	**60.9**	**39.1**	65.5	25.7
	IUPAC name	2.8	6.0	**90.5**	**49.9**
	SMILES	0.7	1.2	90.4	38.4

which is compared with a χ^2 distribution with one degree of freedom.

Table 14 reports the paired results for identification, therapeutic indication, and benzene-ring detection. The two comparisons emphasized in the main text, trivial name versus SMILES for therapeutic indication and SMILES versus trivial name for benzene detection, are significant for all seven models ($p < 10^{-7}$). Thus, the aggregate differences reflect directional asymmetries on the same paired molecules rather than differences between evaluation sets.

B.4 Recognition and Cross-Identifier Binding

This section provides the implementation details for the recognition controls used in Section 2.3. We evaluate whether a model can recover the trivial name associated with a SMILES string using two complementary criteria: two-alternative forced choice (2AFC) and free-form generation.

2AFC recognition. For each molecule, the model is shown its canonical SMILES together with the correct trivial name and one incorrect candidate:

```
A compound is given as SMILES {SMILES}.
Which name identifies it?
A) {option_a}
B) {option_b}
Reply with a single letter only.
```

The order of the correct and incorrect candidates is randomized. Responses are scored by extracting a standalone `A` or `B`.

We construct three decoy conditions. In the *near-analogue* condition, the incorrect candidate is sampled from molecules with Morgan-fingerprint similarity of at least 0.4. We use radius-2, 2,048-bit Morgan fingerprints (Rogers & Hahn, 2010) and compute molecular similarity using the Tanimoto coefficient (Bajusz et al., 2015). If no molecule satisfies the similarity threshold, the decoy is sam-

Table 13: **Additional structure-computable tasks.** Accuracy (%) for molecular-formula prediction, total ring count, and heavy-atom count under each molecular identifier. Majority-class baselines for ring count and heavy-atom count over the 896 molecules are 23.9% and 5.5%, respectively. Best performance within each model and task is shown in bold.

Model	Identifier	Formula	Ring count	Heavy atoms
ChemDFM-v2.0-14B	Trivial name	7.7	**24.4**	6.8
	IUPAC name	**56.8**	16.9	**10.6**
	SMILES	43.5	7.7	9.8
ChemDFM-v1.5-8B	Trivial name	**0.0**	23.0	**3.1**
	IUPAC name	**0.0**	26.0	2.3
	SMILES	**0.0**	**48.1**	1.5
Qwen2.5-14B-Instruct	Trivial name	**12.7**	24.3	2.7
	IUPAC name	5.2	19.9	**4.8**
	SMILES	1.1	**39.3**	3.3
Qwen2.5-7B-Instruct	Trivial name	**7.0**	26.8	0.9
	IUPAC name	3.9	**39.4**	**3.9**
	SMILES	0.2	14.3	1.2
Qwen2-72B-Instruct	Trivial name	**16.7**	32.5	4.0
	IUPAC name	7.1	**55.1**	8.5
	SMILES	2.9	48.4	**10.9**
Meta-Llama-3-8B-Instruct	Trivial name	**8.7**	**24.9**	2.3
	IUPAC name	4.4	20.9	4.9
	SMILES	1.1	18.3	**7.5**
Llama-3.1-8B-Instruct	Trivial name	**10.2**	27.0	0.4
	IUPAC name	5.7	**30.2**	6.1
	SMILES	2.5	22.8	**7.8**

Table 14: **Paired McNemar tests for cross-identifier differences.** Each entry reports n_{10} versus n_{01}, where n_{10} favors the direction stated in the column heading, followed by the continuity-corrected McNemar p-value.

Model	Identify: Name > SMILES	Indication: Name > SMILES	Benzene: SMILES > Name
ChemDFM-v2.0-14B	775 vs. 1 ($p = 2 \times 10^{-169}$)	333 vs. 4 ($p = 2 \times 10^{-71}$)	198 vs. 37 ($p = 2 \times 10^{-25}$)
ChemDFM-v1.5-8B	598 vs. 19 ($p = 9 \times 10^{-120}$)	183 vs. 14 ($p = 5 \times 10^{-33}$)	212 vs. 88 ($p = 1 \times 10^{-12}$)
Qwen2.5-14B-Instruct	765 vs. 1 ($p = 3 \times 10^{-167}$)	331 vs. 3 ($p = 1 \times 10^{-71}$)	322 vs. 68 ($p = 1 \times 10^{-37}$)
Qwen2.5-7B-Instruct	370 vs. 1 ($p = 2 \times 10^{-81}$)	299 vs. 4 ($p = 5 \times 10^{-64}$)	292 vs. 93 ($p = 6 \times 10^{-24}$)
Qwen2-72B-Instruct	553 vs. 4 ($p = 2.9 \times 10^{-119}$)	384 vs. 2 ($p = 8.9 \times 10^{-84}$)	215 vs. 96 ($p = 2.2 \times 10^{-11}$)
Meta-Llama-3-8B-Instruct	281 vs. 5 ($p = 2 \times 10^{-59}$)	338 vs. 0 ($p = 5 \times 10^{-75}$)	248 vs. 51 ($p = 9 \times 10^{-30}$)
Llama-3.1-8B-Instruct	542 vs. 2 ($p = 4 \times 10^{-118}$)	343 vs. 4 ($p = 1 \times 10^{-73}$)	278 vs. 55 ($p = 5 \times 10^{-34}$)

pled from the five nearest molecules. In the *random* condition, the decoy is sampled uniformly from all other eligible molecules. In the *gross-mismatch* condition, the decoy is sampled from molecules whose heavy-atom count differs from the target by more than a factor of two, falling back to the full eligible pool when necessary. Heavy-atom counts follow the RDKit-based definition in Appendix B.2. Sampling uses a fixed random seed of 0. Decoys are sampled independently across conditions and therefore need not be distinct. A molecule is considered recognized under 2AFC only if the model answers correctly under all three decoy conditions.

Free-form recognition. The stricter free-form control uses the standard SMILES-to-name identification task defined in Appendix B.2: given only the canonical SMILES, the model must generate the molecule's trivial name without candidate answers. Predictions and gold names are normalized by lowercasing, removing possessive suffixes and non-alphanumeric punctuation, normalizing

whitespace, and lightly singularizing words longer than four characters that end in `s` but not `ss`. Recognition succeeds when the normalized gold name occurs as a word-boundary-delimited substring of the normalized response.

Recognized-subset evaluation. For either recognition criterion, let R denote the set of recognized molecules that also appear in the therapeutic-indication evaluation. Table 2 reports three quantities. *Name* is therapeutic-indication accuracy when molecules in R are queried through their trivial names; *SMILES* is accuracy on the same molecules when queried through canonical SMILES; and *Rest* is canonical-SMILES accuracy on molecules outside R. Thus, the Name and SMILES columns always compare exactly the same recognized molecules. All therapeutic-indication results reported in Table 2 use the strict indication-scoring criterion defined in Appendix B.2.

C Identifier-Bridge and Conflict Experiments

This section provides the full protocols and additional results supporting Section 3.

C.1 Identifier-Bridge Protocol

For each molecule with canonical SMILES s and trivial name n, we evaluate therapeutic-indication retrieval under three matched conditions. The exact prompts are:

SMILES only.

```
Consider the molecule with SMILES {SMILES}.
What medical condition is this substance primarily used to
treat?
Answer in English.  Reply with the condition only.  Do not
explain.
```

SMILES + name.

```
Consider the molecule with SMILES {SMILES}, which is known
as "{NAME}".
What medical condition is this substance primarily used to
treat?
Answer in English.  Reply with the condition only.  Do not
explain.
```

Name only.

```
Consider the molecule commonly known as "{NAME}".
What medical condition is this substance primarily used to
treat?
Answer in English.  Reply with the condition only.  Do not
explain.
```

The SMILES+name condition therefore supplies only the correspondence between the structural and lexical identifiers; it does not reveal the therapeutic indication. Each fact and identifier condition is queried independently. Decoding and indication scoring follow Appendix B.2.

We quantify how much of the SMILES–name access gap is closed as

$$\text{Recovery} = \frac{A_{\text{SMILES+Name}} - A_{\text{SMILES}}}{A_{\text{Name}} - A_{\text{SMILES}}} \times 100,$$

where A denotes therapeutic-indication accuracy. A value of 100% means that supplying the name closes the entire gap between SMILES-only and name-only retrieval.

Reverse bridge. We also test the converse intervention for two structure-computable properties: benzene-ring presence and nitrogen-atom count. Here we compare the trivial name alone, the

Table 15: **Reverse identifier bridge.** Accuracy (%) for Qwen2-72B-Instruct. Reverse recovery is the fraction of the name-to-SMILES accuracy gap closed by adding SMILES to the name. Values above 100% indicate that the combined prompt slightly exceeds SMILES alone.

Property	Name	Name + SMILES	SMILES	Recovery
Benzene	71.1	85.7	84.3	111.0
Nitrogens	39.6	62.3	61.5	103.6

Table 16: **Conflicting identifiers control different outputs:** Each input pairs the SMILES of molecule A with the incorrect trivial name of molecule B. Identity reports retrieval of A versus B; therapeutic indication compares retrieval of B's indication under conflict with retrieval from B's name alone; benzene compares accuracy under conflict with A's SMILES-only condition. All values are percentages.

Model	Identity: A/B	Indication of B: conflict / name	Benzene of A: conflict / SMILES
ChemDFM-v2.0-14B	1.0 / 85.5	32.3 / 38.6	95.3 / 94.5
ChemDFM-v1.5-8B	0.6 / 73.4	17.4 / 23.0	80.5 / 85.0
Qwen2.5-14B-Instruct	0.0 / 81.1	35.9 / 37.8	84.9 / 86.9
Qwen2.5-7B-Instruct	0.0 / 60.4	32.1 / 33.9	75.9 / 80.2
Llama-3.1-8B-Instruct	0.0 / 86.8	37.3 / 39.0	89.3 / 90.3
Qwen2-72B-Instruct	0.0 / 77.8	43.2 / 44.1	83.9 / 84.3

trivial name augmented with the canonical SMILES, and the SMILES alone. For example, the benzene prompts use `Consider the molecule commonly known as "{NAME}".`, `Consider the molecule commonly known as "{NAME}", whose SMILES is {SMILES}.`, and `Consider the molecule with SMILES {SMILES}.`, followed by the same benzene-ring question used in the benchmark. Nitrogen counting uses the corresponding three identifier prefixes followed by the standard nitrogen-count question.

Table 15 reports the complete saved reverse-bridge results for Qwen2-72B-Instruct. Adding SMILES to the lexical prompt closes, and slightly exceeds, the name–SMILES gap for both properties.

C.2 Conflicting-Identifier Protocol

To test whether lexical and structural identifiers are reconciled into a single entity representation, we pair the canonical SMILES of molecule A with the trivial name of a different molecule B. For each molecule, the decoy is sampled from molecules whose normalized reference therapeutic indication differs from that of the true molecule. One candidate is drawn using a fixed random seed of 0.

The conflicting identifier is presented as

```
the molecule with SMILES {SMILES}, which is known as
"{DECOY_NAME}"
```

and each queried property is evaluated in a separate prompt. For identity, therapeutic indication, benzene-ring presence, and nitrogen count, we append the same task-specific question used in the corresponding benchmark evaluation. Thus, the experiment does not ask multiple properties jointly.

For identity, we measure how often the response follows the true molecule A specified by the SMILES versus the conflicting lexical identity B. For therapeutic indication, we measure retrieval of B's indication and compare it with retrieval when B is queried through its name alone. For structure-computable properties, we compare accuracy under conflict with the corresponding SMILES-only condition for A.

Table 16 reports the cross-model results used in Section 3.2. Entity-associated outputs largely follow the incorrect lexical identity, whereas benzene judgments remain close to the SMILES-only condition. For Qwen2-72B-Instruct, the same conflicting prompts additionally retain 59.9% nitrogen-count accuracy, while the supplied decoy name is recovered in 77.8% of identity queries and the

true name in 0.0%. This provides a second structure-computable example of the same split between lexical and structural access.

C.3 Direct Detection of Identifier Conflicts

We separately test whether models can explicitly recognize that a supplied SMILES and trivial name are inconsistent. Each canonical SMILES is evaluated with its correct name and with three types of incorrect names.

Mismatch construction. For the near-analogue condition, we represent each molecule using a radius-2, 2,048-bit Morgan fingerprint (Rogers & Hahn, 2010), a circular molecular fingerprint that encodes local structural subpatterns, and measure pairwise similarity using the Tanimoto coefficient (Bajusz et al., 2015). The incorrect name is sampled uniformly from molecules with Tanimoto similarity at least 0.4 to the target. If no such candidate exists, the decoy is sampled from the five nearest molecules, with all other eligible molecules used as a final fallback.

For the random condition, the decoy is sampled uniformly from all other molecules with valid fingerprints. For the gross-mismatch condition, the decoy is sampled from molecules whose heavy-atom count differs from the target by more than a factor of two, falling back to the full eligible pool when necessary. Sampling uses a fixed random seed of 0; decoys are sampled independently across conditions and therefore need not be distinct.

Direct consistency check. The exact prompt is:

```
A compound is given as SMILES {SMILES} and is labelled
"{LABEL}".
Does this label correctly identify the structure
shown?
Reply with exactly one word:  yes or no.
```

The response parser evaluates negative cues before positive cues. Negative cues include `no`, `not`, `incorrect`, `mismatch`, `wrong`, `different`, and `false`; positive cues include `yes`, `correct`, `match`, and `true`.

Near-analogue 2AFC. We additionally ask the model to choose between the true trivial name and the selected near-analogue:

```
A compound is given as SMILES {SMILES}.
Which name identifies it?
A) {OPTION A}
B) {OPTION B}
Reply with a single letter only.
```

The order of the true and incorrect candidates is randomized.

Discrimination metric. Mismatch-rejection rate alone can be misleading because a model could obtain a high rejection rate simply by rejecting most name–SMILES pairs. We therefore also report the signal-detection sensitivity index d' (Stanislaw & Todorov, 1999), which accounts for false alarms on correctly matched pairs. Let H denote the probability of rejecting a mismatched pair and F the probability of incorrectly rejecting a correctly matched pair. We compute

$$d' = Z(H) - Z(F),$$

where Z is the inverse standard-normal cumulative distribution function. Larger d' indicates better discrimination between matched and mismatched identifier pairs. Before applying Z, both probabilities are clipped to $[0.001, 0.999]$. No sample-size-dependent correction is applied.

Gross mismatches are substantially easier to detect than structurally similar ones. For ChemDFM-v2.0-14B, rejection falls from 84.3% for gross mismatches to 36.9% for near analogues, with d' decreasing from 2.53 to 1.19. Across the remaining models, near-analogue rejection ranges from

Table 17: **Direct detection of identifier conflicts.** "Correct accepted" is the percentage of valid name–SMILES pairs that the model accepts as matching. "Near reject" and "Gross reject" are the percentages of near-analogue and gross mismatches correctly rejected, respectively. The corresponding d' values measure discrimination while accounting for false alarms on correctly matched pairs. "Near 2AFC" is accuracy when choosing between the true trivial name and a structurally similar incorrect name. All rates are percentages except d'.

Model	Correct accepted	Near reject	Near d'	Gross reject	Gross d'	Near 2AFC
ChemDFM-v2.0-14B	93.6	36.9	1.19	84.3	2.53	73.9
ChemDFM-v1.5-8B	65.5	43.1	0.22	62.0	0.70	71.9
Qwen2.5-14B-Instruct	76.5	38.6	0.43	59.7	0.97	62.8
Qwen2.5-7B-Instruct	87.3	17.5	0.21	28.5	0.57	56.4
Qwen2-72B-Instruct	91.7	22.2	0.62	47.0	1.31	65.5

17.5% to 43.1%, with d' between 0.21 and 0.62. Near-analogue 2AFC accuracy is only 56.4–73.9%, despite a 50% chance baseline.

For Qwen2-72B-Instruct, for which complete saved random-mismatch discrimination results are also available, random mismatches fall between the two extremes: rejection is 32.6% with $d' = 0.94$, compared with 22.2% and 0.62 for near analogues and 47.0% and 1.31 for gross mismatches. Random-mismatch 2AFC accuracy is 70.6%.

D Activation-Patching Details

This section provides the implementation details for the causal intervention in Section 4.1.

D.1 Source and Target Construction

For each molecule, we construct matched prompts querying the same therapeutic indication through either the trivial name (*source*) or canonical SMILES (*target*):

```
The molecule commonly known as "{NAME}" is primarily used to
treat

The molecule with SMILES {SMILES} is primarily used to treat
```

The gold continuation is the reference indication preceded by a single space. For chat models, the source or target string is placed in the user turn with the system message `You are a helpful assistant.` and the model's chat template is applied with a generation prompt. The gold continuation is then scored autoregressively under teacher forcing.

D.2 Item Selection

We first compute the unpatched mean gold-token log-probability under the trivial-name and SMILES prompts. We retain only molecules for which lexical access is measurably stronger,

$$\mathrm{lp}_{\mathrm{name}} - \mathrm{lp}_{\mathrm{SMILES}} \geq 0.20$$

nats per gold token. This ensures that recovery is evaluated on examples that exhibit a pre-existing name–SMILES access gap. For Qwen2-72B-Instruct, 491 of the 896 benchmark molecules satisfy this criterion, spanning 176 distinct indication strings.

D.3 Activation Intervention

At a tested transformer block ℓ, we capture the block-output residual state at the final prompt token in the trivial-name source run. We then run the matched SMILES prompt and replace the complete block-output vector at its final prompt position with the captured source vector:

$$h^{(\ell)}_{\text{SMILES}} \leftarrow h^{(\ell)}_{\text{name}}.$$

All other token positions remain unchanged, and computation proceeds normally through the remaining transformer blocks. Each layer intervention is evaluated independently.

Layers are searched using a coarse-to-fine procedure: an initial regularly spaced sweep identifies a coarse peak, after which the surrounding layers are evaluated more densely. The reported peak is the tested layer with the largest mean recovery. Relative depths in the main text normalize the selected layer index by model depth.

D.4 Teacher-Forced Scoring and Recovery

For a gold continuation $g_1, \ldots, g_T$, we compute mean token log-probability

$$\text{lp} = \frac{1}{T} \sum_{t=1}^{T} \log p(g_t \mid x, g_{<t}),$$

where x is the rendered prompt. Gold tokens are scored autoregressively under teacher forcing. Logits are converted to float32 before applying `log_softmax`.

Recovery is computed separately for every molecule i and tested layer ℓ:

$$R_{i,\ell} = \frac{\text{lp}^{\text{patched}}_{i,\ell} - \text{lp}^{\text{SMILES}}_{i}}{\max(\text{lp}^{\text{name}}_{i} - \text{lp}^{\text{SMILES}}_{i}, 10^{-6})}.$$

The reported layer-level recovery is the arithmetic mean of $R_{i,\ell}$ across selected molecules, multiplied by 100. Thus, 100% indicates that patching closes the original name–SMILES gap on average, while values above 100% indicate that the patched computation exceeds the corresponding name-conditioned score. Standard errors are computed across the per-molecule recovery values.

D.5 Different-Entity Control

To test whether recovery arises from injecting arbitrary lexical activity, each focal molecule is assigned a control molecule with a different reference indication. The control is sampled uniformly from the selected items using a fixed random seed of 0. The assignment is made once and reused at every tested layer.

At each layer, we therefore repeat the intervention using the control molecule's trivial-name activation while retaining the focal molecule's SMILES target and gold indication. Control recovery is computed using the same denominator as the matched intervention:

$$R^{\text{ctrl}}_{i,\ell} = \frac{\text{lp}^{\text{ctrl}}_{i,\ell} - \text{lp}^{\text{SMILES}}_{i}}{\max(\text{lp}^{\text{name}}_{i} - \text{lp}^{\text{SMILES}}_{i}, 10^{-6})}.$$

The consistently negative controls in Table 5 show that recovery is not produced merely by inserting an arbitrary lexical-state vector.

Qwen2-72B-Instruct layer sweep. For the 80-layer Qwen2-72B-Instruct model, the initial sweep evaluates every fourth layer. After a coarse peak at layer 68, the surrounding region is searched densely, yielding 32 tested layers in total. The final peak occurs at layer 66, with mean recovery $138.4\% \pm 8.0$ percentage points (SEM); the corresponding different-entity control is -30.9%.

E Cross-Identifier Representation Analysis

This section provides the implementation details and complete results for the representation analysis in Section 4.2.

E.1 Representation Extraction

For each of the 896 benchmark molecules, we construct three identifier-specific prompts:

```
the molecule commonly known as "{NAME}"
the molecule named {IUPAC_NAME}
the molecule with SMILES {SMILES}
```

Each complete prompt is tokenized using the checkpoint's tokenizer with its default special-token behavior. Batches are right-padded and truncated to at most 512 tokens. Hidden states are extracted for every state returned by the model.

For molecule i, identifier form a, and hidden-state index ℓ, let $\mathbf{H}_{i,a,t}^{(\ell)}$ denote the hidden state at token position t, and let $m_{i,a,t} \in \{0, 1\}$ be the corresponding attention-mask value. We represent the prompt by the arithmetic mean over all non-padding tokens:

$$\mathbf{h}_{i,a}^{(\ell)} = \frac{\sum_t m_{i,a,t} \mathbf{H}_{i,a,t}^{(\ell)}}{\sum_t m_{i,a,t}}.$$

Thus, the representation is a mean over the complete non-padding prompt rather than over the identifier tokens alone. It includes the fixed prompt prefix, the molecular identifier, punctuation such as quotation marks when present, and any special tokens added by the tokenizer. Padding tokens are excluded.

We use the hidden-state sequence returned directly by the model. Hidden-state index 0 corresponds to the embedding output, followed by the states returned after successive transformer computations. We therefore report the exact hidden-state index used by the implementation rather than converting it to a separately defined block number.

The extraction pipeline also supports pooling at the final non-padding prompt token, but all results reported in Section 4.2, Figure 2, and this appendix use the mean-pooled representations defined above. We do not select the pooling strategy separately for different identifier pairs.

E.2 Cross-Identifier Retrieval

We evaluate whether the representation of a molecule under one identifier can retrieve the representation of the same molecule under another identifier. The candidate set contains $N = 896$ molecule rows.

Before retrieval, representations are centered separately for each identifier form and hidden-state index. For identifier form a,

$$\boldsymbol{\mu}_a^{(\ell)} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{h}_{i,a}^{(\ell)},$$

and each centered representation is L2-normalized:

$$\tilde{\mathbf{h}}_{i,a}^{(\ell)} = \frac{\mathbf{h}_{i,a}^{(\ell)} - \boldsymbol{\mu}_a^{(\ell)}}{\left\| \mathbf{h}_{i,a}^{(\ell)} - \boldsymbol{\mu}_a^{(\ell)} \right\|_2 + 10^{-8}}.$$

For a query molecule i represented under identifier a, retrieval under identifier b is

$$\hat{j} = \arg \max_{j \in \{1,\ldots,N\}} \left(\tilde{\mathbf{h}}_{i,a}^{(\ell)} \right)^{\top} \tilde{\mathbf{h}}_{j,b}^{(\ell)}.$$

Because the vectors are L2-normalized, this dot product is cosine similarity after mean-centering. No PCA, whitening, or other representation transformation is applied.

Top-1 retrieval is correct when $\hat{j} = i$. With 896 candidate rows, uniform chance performance is

$$\frac{1}{896} \approx 0.11\%.$$

Table 18: **Peak cross-identifier representation retrieval.** Top-1 retrieval accuracy (%) among 896 molecule rows using mean-pooled whole-prompt hidden representations. Name–SMILES denotes trivial name → SMILES retrieval, and IUPAC–SMILES denotes IUPAC name → SMILES retrieval. "Peak idx." is the hidden-state index attaining the maximum accuracy; index 0 is the embedding output. "States" is the total number of returned hidden states. Chance performance is 0.11%.

	Name → SMILES		IUPAC → SMILES		
Model	Top-1	Peak idx.	Top-1	Peak idx.	States
ChemDFM-v2.0-14B	0.8	28	13.5	31	49
ChemDFM-v1.5-8B	2.8	24	15.0	32	33
Qwen2.5-14B-Instruct	0.7	36	2.6	30	49
Qwen2.5-7B-Instruct	0.7	3	1.8	25	29
Meta-Llama-3-8B-Instruct	0.8	13	3.9	18	33
Llama-3.1-8B-Instruct	0.9	6	4.0	18	33

Throughout the paper, **Name–SMILES** denotes the directional retrieval task

$$\text{trivial name} \rightarrow \text{SMILES},$$

and **IUPAC–SMILES** denotes

$$\text{IUPAC name} \rightarrow \text{SMILES}.$$

The reported metric is not an average over the two retrieval directions.

The 896-row candidate set contains distinct molecule identifiers, trivial names, and canonical SMILES. Two IUPAC strings each occur for two distinct molecule rows, yielding 894 distinct IUPAC strings overall. Evaluation remains row-index based: the representation corresponding to the same molecule row is treated as the gold target.

E.3 Layer-Wise Peak Results

Top-1 retrieval is computed independently at every saved hidden-state index. For each model and identifier pair, we report the maximum retrieval accuracy over these indices. Different identifier pairs may therefore attain their maximum at different depths.

Table 18 reports the complete mean-pooled, forward-retrieval results together with the hidden-state index at which each maximum occurs.

F Lookup-Based Lexical Routing: Evaluation and Controls

This section provides the evaluation setup, scoring procedure, and self-resolution control for the lexical-routing experiment in Section 4.3.

F.1 Multi-Relation Evaluation Set

We evaluate four entity-associated relations: therapeutic indication, molecular target, mechanism of action (MoA), and ATC therapeutic class. Starting from the cleaned benchmark collection described in Appendix B.1, we augment each molecule with available ChEMBL relation annotations and retain molecules for which at least two of the four relations are available. This yields 891 molecules and 2,955 molecule–relation pairs.

Therapeutic indications are obtained from ChEMBL `drug_indication` records with `max_phase_for_ind >= 4`. Molecular targets are obtained through `mechanism.target_chembl_id` and mapped to the corresponding target preferred name. Mechanism-of-action labels use `mechanism_of_action`. For ATC, we consider up to the first three available classifications and use the level-4 description, falling back to the level-3 description when necessary. When multiple targets, mechanisms, or ATC labels are available, all are retained as acceptable answers.

Table 19: **Coverage of the multi-relation evaluation set.** Number of molecules with at least one accepted answer for each relation.

Relation	Molecules
Therapeutic indication	891
Molecular target	572
Mechanism of action	651
ATC therapeutic class	841
Total molecule–relation pairs	2,955

F.2 Lexical-Routing Conditions

For each available molecule–relation pair, we evaluate four conditions:

1. **SMILES**: query the relation directly from the molecule's canonical SMILES.
2. **Self-resolution**: first ask the model to generate the molecule's common name from SMILES, then insert the generated name into the factual query.
3. **Lookup**: supply the ChEMBL preferred name associated with the molecule together with its SMILES before asking the factual question.
4. **Name**: query the relation directly from the ChEMBL preferred name.

The lookup condition implements the routing path

$$\text{SMILES} \xrightarrow{\text{name lookup}} \text{Lexical identity} \xrightarrow{\text{LM}} \text{Fact}.$$

The lookup supplies only the correspondence between the SMILES and the lexical identifier; it contains no information about the queried relation. In the implementation, the name is the ChEMBL preferred name already associated with the molecule in the benchmark record. No live resolver API is called during inference.

All factual queries use

```
Answer the question about the compound below.
Compound: {REFERENCE}
Question: {QUESTION}
Answer in English. Reply with the answer only. Do not
explain.
```

The relation-specific questions are

```
what medical condition is it primarily used to treat?
what is its primary molecular target?
what is its mechanism of action?
what therapeutic class does it belong to?
```

with the following reference strings:

SMILES: `SMILES {SMILES}`
Self-resolution: `SMILES {SMILES} (identified as {GENERATED_NAME})`
Lookup: `SMILES {SMILES} (resolved by name lookup to {NAME})`
Name: `{NAME}`

For self-resolution, the model first receives

```
A database lookup returned this compound as SMILES: {SMILES}
What is its common generic name?
Answer in English. Reply with the name only.
```

Table 20: **Self-resolution results.** SMILES-to-name identification accuracy and downstream factual accuracy (%) when the generated name is inserted into the query. Relation columns are evaluated over molecules for which the corresponding relation is available.

Model	Self-ID	Indication	Target	MoA	ATC
ChemDFM-v2.0-14B	7.1	4.9	11.4	4.5	1.4
ChemDFM-v1.5-8B	12.0	6.6	11.4	2.0	2.5
Qwen2.5-14B-Instruct	0.3	1.2	4.0	3.1	1.1
Qwen2.5-7B-Instruct	0.1	1.6	0.7	0.8	0.0
Meta-Llama-3-8B-Instruct	0.7	2.6	3.3	1.1	1.8
Llama-3.1-8B-Instruct	0.6	2.5	5.8	0.3	1.2

Prompts are wrapped in the model's chat template when available, using the system message `You are a helpful assistant.` Generation is greedy (`do_sample=False`) with at most 32 new tokens. Across 2,955 available molecule–relation pairs, the four factual-query conditions produce 11,820 factual queries.

F.3 Answer Normalization and Scoring

Predictions and accepted answers are normalized by lowercasing, removing possessive suffixes, replacing non-alphanumeric punctuation with spaces, collapsing whitespace, and lightly singularizing words longer than four characters that end in `s` but not `ss`. Empty responses, refusals, and responses longer than 14 normalized words are counted as incorrect.

For all four relations, a response is correct when an accepted normalized label occurs as a word-boundary-delimited phrase in the normalized prediction. For molecular target and mechanism of action, we additionally accept a prediction when it covers at least 60% of the content words of an accepted label after removing stopwords and tokens of length at most two. Therapeutic indication and ATC class use only the phrase-matching criterion.

F.4 Gap-Recovery Metric

The main-text table reports aggregate factual accuracy under the SMILES, lookup, and name conditions. To quantify how much of the SMILES–name access gap is recovered by lexical routing, we compute

$$\mathrm{GapRecovery} = 100 \times \frac{A_{\mathrm{lookup}} - A_{\mathrm{SMILES}}}{\max(A_{\mathrm{name}} - A_{\mathrm{SMILES}}, 10^{-9})},$$

where A_{SMILES}, A_{lookup}, and A_{name} denote aggregate accuracy under the corresponding conditions. The complete aggregate results are reported in Table 6 in the main text.

F.5 Self-Resolution Control

Self-resolution asks whether the model can construct the missing lexical bridge without an external identifier lookup. The model first predicts a name from SMILES, and that generated name is then used in the downstream factual query regardless of whether it matches the benchmark name.

SMILES-to-name identification ranges from 0.1% to 12.0% across the six models. The low identification rates explain why self-resolution does not provide the same access gains as supplying the correct lexical correspondence directly.

G Knowledge Editing: Setup and Cross-Identifier Transfer

This section provides the experimental details for Section 4.4. We evaluate ROME and fine-tuning (FT) on ChemDFM-v2.0-14B, Qwen2.5-14B-Instruct, Qwen2.5-7B-Instruct, and Llama-3.1-8B-Instruct. Each counterfactual edit is introduced through the molecule's trivial name and evaluated through the name, IUPAC name, and SMILES.

Table 21: **Knowledge-editing evaluation prompts.**

Condition	Prompt
Efficacy	`{Name.capitalize()} is primarily used to treat`
IUPAC	`The molecule named {iupac} is primarily used to treat`
SMILES	`The molecule with SMILES {smiles} is primarily used to treat`
SMILES+Name	`The molecule with SMILES {smiles}, which is known as "{name}", is primarily used to treat`
Locality	`{Unrelated_name.capitalize()} is primarily used to treat`

G.1 Edit Construction and Evaluation

For each model, we retain molecules whose therapeutic indication is already retrievable through the trivial name, yielding 389, 390, 351, and 389 molecules for ChemDFM-v2.0-14B, Qwen2.5-14B-Instruct, Qwen2.5-7B-Instruct, and Llama-3.1-8B-Instruct, respectively.

For each molecule x, we replace its original indication with a counterfactual indication y^* sampled from another eligible molecule. Candidates sharing normalized content words with any accepted true indication of x are excluded before uniform sampling with seed 0. The original raw indication string is used as the editing target. The edit request is

`{common name} is primarily used to treat`

with y^* as the desired continuation. Each edit is applied and evaluated independently, after which the modified weights are restored.

We evaluate the edited target under the following prompts:

Generation is deterministic with `do_sample=False` and `max_new_tokens=24`. We score the generated continuation using the same normalized phrase-matching procedure as the main factual-retrieval evaluation.

Editing implementation details. For each model and editing method, we first evaluated candidate editing layers on a 30-example pilot sweep and used the pilot results to select the layer for the full experiment. Because the pilot exhibited different efficacy–locality trade-offs across models and methods, we did not impose a single scalar selection metric across all settings. The resulting ROME/FT layers were 24/24 for ChemDFM-v2.0-14B, 30/36 for Qwen2.5-14B-Instruct, 17/14 for Qwen2.5-7B-Instruct, and 20/16 for Llama-3.1-8B-Instruct, respectively. ROME edits the MLP down-projection at the selected layer and optimizes the target vector for at most 25 steps using Adam with learning rate 0.5, weight decay 10^{-3}. FT optimizes the same down-projection for 25 steps using Adam with learning rate 5×10^{-4}, batch size 1, zero weight decay, and element-wise updates constrained to 5×10^{-4} from the original weights. Both methods use FP32 models, and evaluation uses greedy decoding with at most 24 new tokens.

G.2 Leakage-Corrected Cross-Identifier Transfer

Retrieval of y^* through SMILES does not necessarily imply that the edit transferred to the corresponding molecule: an edit may increase production of y^* for unrelated molecules as well. We therefore estimate this background effect using locality probes spanning molecular similarities based on radius-2, 2,048-bit Morgan fingerprints and Tanimoto similarity.

We define

$$T_{\text{leak}} = \text{mean}\big(\texttt{post_new} \mid \texttt{axis=locality}\big),$$

and report entity-specific SMILES transfer as

Table 22: **Full cross-identifier knowledge-editing results.** Leak is the pooled locality rate and Specific is SMILES minus Leak. Brackets give 95% bootstrap confidence intervals. Recovery is SMILES+Name performance relative to name-conditioned efficacy.

Model	Method	n	Efficacy	IUPAC	SMILES	Leak	Specific [95% CI]	SMILES+Name	Recovery
ChemDFM-v2.0-14B	ROME	389	52.2	24.2	22.1	19.8	+2.3 [−0.9,5.5]	45.5	87.2
	FT	389	26.0	2.8	1.8	2.1	-0.3 [−1.6,1.2]	25.2	97.0
Qwen2.5-14B-Instruct	ROME	390	17.9	3.8	4.4	4.9	-0.5 [−2.4,1.4]	12.3	68.6
	FT	390	29.0	14.4	11.5	8.0	+3.5 [0.9,6.3]	26.7	92.0
Qwen2.5-7B-Instruct	ROME	351	47.9	4.3	4.0	7.4	-3.4 [−5.7,−0.9]	11.4	23.8
	FT	351	34.5	8.5	5.4	7.3	-1.9 [−4.4,0.8]	21.7	62.8
Llama-3.1-8B-Instruct	ROME	389	44.2	21.9	21.1	8.1	+13.0 [9.2,16.9]	40.4	91.3
	FT	389	38.3	18.8	12.6	11.4	+1.2 [−1.6,4.1]	32.1	83.9

Table 23: **Transfer conditioned on successful name edits.** Specific is conditional SMILES transfer minus conditional leakage. Brackets give 95% bootstrap confidence intervals.

Model	Method	n_{succ}	SMILES	Leak	Specific [95% CI]	SMILES+Name
ChemDFM-v2.0-14B	ROME	203	33.50	32.04	+1.46 [−3.63,6.51]	67.98
	FT	101	4.95	5.94	-0.99 [−5.12,3.95]	64.36
Qwen2.5-14B-Instruct	ROME	70	18.57	20.99	-2.42 [−10.98,6.83]	57.14
	FT	113	30.97	20.14	+10.83 [3.39,18.63]	67.26
Qwen2.5-7B-Instruct	ROME	168	7.14	13.23	-6.09 [−10.73,−1.36]	22.62
	FT	121	12.40	16.99	-4.59 [−10.96,2.12]	46.28
Llama-3.1-8B-Instruct	ROME	172	38.37	11.95	+26.43 [19.63,33.19]	74.42
	FT	149	26.85	22.74	+4.10 [−1.77,10.31]	64.43

$$T_{\text{specific}} = T_{\text{SMILES}} - T_{\text{leak}}.$$

Confidence intervals are obtained from 10,000 percentile-bootstrap resamples with seed `20260921`, resampling at the edited-molecule level and carrying the associated locality probes with each sampled molecule.

G.3 Full Results

Across the eight configurations, six show no reliable positive entity-specific transfer to SMILES beyond the leakage baseline. The two exceptions are Qwen2.5-14B-Instruct with FT and Llama-3.1-8B-Instruct with ROME, showing that standard editing can transfer across identifiers, but does so inconsistently across models and methods.

G.4 Conditioning on Successful Name Edits

To rule out the possibility that weak transfer is caused simply by unsuccessful edits, we repeat the analysis using only molecules for which the counterfactual target is successfully retrieved through the trivial-name efficacy prompt. Leakage is recomputed using locality probes associated only with these successful edits.

The conclusion is unchanged after conditioning on successful name edits: the same two configurations show clear positive specific transfer beyond leakage, while the remaining six do not. Weak cross-identifier transfer therefore cannot be explained simply by failure to install the edit through the trivial name.

H Binding Self-Distillation: Experimental Details and Additional Results

This section provides the experimental details and supplementary results for Binding Self-Distillation in Section 5.

Table 24: **Model-specific Binding Self-Distillation sets.** The reported sets are the fixed entity sets selected by the original BSD entity-selection pipeline using therapeutic-indication retrieval through the trivial name.

Model	Entities
ChemDFM-v2.0-14B	389
Qwen2.5-14B-Instruct	390
Llama-3.1-8B-Instruct	389
ChemDFM-v1.5-8B	47
Qwen2.5-7B-Instruct	351
Meta-Llama-3-8B-Instruct	378
Qwen2-72B-Instruct	435

H.1 Experimental Setup and Entity Selection

Binding Self-Distillation is evaluated in a per-entity setting: a separate lightweight adapter is trained and evaluated for each molecule. Parameters are therefore not shared across molecules.

To ensure that the lexical route already exposes factual knowledge before binding, we construct model-specific sets using therapeutic indication. Starting from the multi-relation molecule collection, we retain a molecule only when the model correctly retrieves its therapeutic indication through the trivial name. Eligibility is determined only by indication; molecular target, mechanism of action (MoA), and ATC class are not additionally filtered for name-conditioned correctness.

The reported Binding Self-Distillation experiments use the fixed entity sets selected by the original experimental pipeline. In that pipeline, responses were matched against accepted ChEMBL indication headings after lowercasing, punctuation and possessive removal, whitespace normalization.

H.2 Teacher–Student Construction

For a molecule with SMILES s, trivial name n, and generic continuation stem u, the student receives only the structural identifier,

$$x_S = \texttt{The molecule with SMILES } s\, u,$$

whereas the teacher receives the same context augmented with the lexical identity,

$$x_T = \texttt{The molecule with SMILES } s\texttt{, which is known as ``}n\texttt{''}\texttt{, } u.$$

Teacher and student are two forward passes through the same pretrained model. The teacher remains frozen, while a LoRA adapter ϕ on the student is optimized using

$$\mathcal{L}_{\text{bind}} = D_{\text{KL}}\left(p_\theta(\cdot \mid x_T) \,\|\, p_{\theta,\phi}(\cdot \mid x_S)\right).$$

The KL divergence is computed over the full vocabulary at the final prompt position. No teacher continuation is generated and no factual answer is provided as a training target. Binding Self-Distillation therefore transfers the model's own predictive behavior induced by lexical identity rather than training on external factual labels.

At every optimization step, six generic continuation stems are sampled uniformly with replacement from a pool of 24. These include stems such as `is a compound that`, `is known for`, `is a drug that`, `acts on`, and `is associated with`. The exact downstream evaluation prompts are excluded from the training pool. Some stems nevertheless cue broad biomedical relation families, so we interpret the resulting gains as cross-prompt transfer across multiple relations rather than strictly relation-held-out generalization.

Anchor regularization. Each binding intervention additionally uses eight unrelated anchor molecules sampled without replacement from the remaining eligible entities using

Table 25: **Binding Self-Distillation configuration for the six verified local runs.**

Hyperparameter	Value
Editing mode	One adapter per entity
Target layer	24
LoRA layers	20–24
LoRA modules	gate/up/down projection
Rank r	8
Scaling α	16
Dropout	0
Optimizer	AdamW
Learning rate	1×10^{-3}
Weight decay	0
Optimization steps	120
Binding contexts / step	6
Anchor molecules	8
Anchor weight	1.0
Objective	Full-vocabulary next-token KL

`random.Random(0)`. At each optimization step, an anchor is paired with a newly sampled generic stem. The frozen model provides the original next-token distribution for that prompt, and an additional KL term with weight 1.0 regularizes the adapted model toward the original behavior.

H.3 Optimization Configuration

For ChemDFM-v2.0-14B, ChemDFM-v1.5-8B, Qwen2.5-7B-Instruct, Qwen2.5-14B-Instruct, Llama-3.1-8B-Instruct, and Meta-Llama-3-8B-Instruct, LoRA is applied to the `gate_proj`, `up_proj`, and `down_proj` modules in MLP layers 20–24, with target layer 24. We use rank $r = 8$, scaling $\alpha = 16$, zero dropout, AdamW with zero weight decay, learning rate 10^{-3}, and 120 optimization steps per entity. Each step uses six binding contexts and eight anchor contexts.

Qwen2-72B-Instruct uses target layer 48 (relative depth 0.60).

H.4 Evaluation

We evaluate four factual relations:

Relation	Continuation ending
Indication	`is primarily used to treat`
Target	`has as its primary molecular target`
MoA	`acts by the following mechanism:`
ATC	`belongs to the therapeutic class`

For each available relation, we evaluate:

SMILES: `The molecule with SMILES {s} {ending}`,
Bridge: `The molecule with SMILES {s}, which is known as "{name}", {ending}`,
Name: `{Name.capitalize()} {ending}`.

Generation is greedy with at most 24 new tokens. Generated answers and accepted targets are normalized by lowercasing, punctuation and possessive removal, whitespace normalization, and light depluralization. A whole-phrase match to any accepted target is counted as correct. For molecular target and MoA, we additionally accept an answer covering at least 60% of the non-stopword content tokens of an accepted target.

Let R_e denote the available factual relations for entity e. We report

Table 26: Relation-wise Binding Self-Distillation results on the fixed entity sets selected by the original BSD entity-selection pipeline. Values are retrieval accuracy (%) through SMILES before and after the per-entity binding intervention.

	Indication		Target		MoA		ATC		
Model	Before	After	Before	After	Before	After	Before	After	Facts/entity
ChemDFM-v2.0-14B	2.1	51.4	0.0	27.6	0.0	5.0	3.3	11.4	0.05 → 0.87
Qwen2.5-14B-Instruct	4.1	13.6	1.4	43.6	0.6	20.8	0.0	10.3	0.06 → 0.73
Llama-3.1-8B-Instruct	0.3	38.8	0.7	19.9	1.2	8.4	1.1	7.9	0.03 → 0.68
ChemDFM-v1.5-8B	25.5	42.6	5.1	23.1	17.1	12.2	6.8	13.6	0.51 → 0.85
Qwen2.5-7B-Instruct	1.1	36.2	0.4	32.3	0.0	19.9	2.7	9.1	0.04 → 0.85
Meta-Llama-3-8B-Instruct	0.3	42.3	0.7	21.8	0.6	13.0	0.3	8.8	0.02 → 0.78

Table 27: Binding Self-Distillation versus explicit lexical access on the fixed entity sets selected by the original BSD entity-selection pipeline. Bridge gives factual access when the correct name is explicitly supplied. Recovery measures the fraction of the SMILES-to-bridge gain recovered without the name at inference time.

Model	n	Facts/entity	Bridge	Recovery
ChemDFM-v2.0-14B	389	0.05 → 0.87	1.01	86%
Qwen2-72B-Instruct	435	0.01 → 0.85	1.34	63%

$$F_c = \frac{1}{N} \sum_{e=1}^{N} \sum_{r \in R_e} \mathbb{1}\left[\mathrm{hit}(e, r, c)\right],$$

where F_c is the mean number of correctly retrieved available relations per entity under condition c. Relations without an accepted answer are not queried.

H.5 Cross-Model Results

Table 26 reports relation-level results.

Across all six models, aggregate factual access increases after Binding Self-Distillation. The gains extend beyond therapeutic indication to molecular target, mechanism of action, and ATC class, supporting transfer beyond the relation used to construct the eligible entity sets.

H.6 Recovery Relative to Explicit Lexical Access

We compare the learned binding intervention with the inference-time lexical bridge, which explicitly supplies the correct trivial name. We define bridge recovery as

$$100 \, \frac{F_{\mathrm{post,SMILES}} - F_{\mathrm{pre,SMILES}}}{\max(F_{\mathrm{pre,bridge}} - F_{\mathrm{pre,SMILES}}, 10^{-9})}.$$

For ChemDFM-v2.0-14B, Binding Self-Distillation recovers 86% of the factual-access gain provided by explicitly supplying the lexical identity. For Qwen2-72B-Instruct, it recovers 63%. In both cases, inference after training uses SMILES alone.

H.7 Retention and Locality

We additionally verify that the increased factual access need not come from destroying behavior that already works. For ChemDFM-v2.0-14B, benzene outputs are preserved on 94.1% of the retention probes. Name-conditioned indication accuracy remains 74.8%; target accuracy changes from 56.2% to 55.2%, ATC accuracy from 16.0% to 16.3%, and neighboring-molecule factual correctness from 75.9% to 76.2%.

For Qwen2-72B-Instruct, exact-string comparison substantially understates retention because the model frequently rephrases its answers. The underlying benzene yes/no verdict is retained in 92.9% of cases, benzene correctness against RDKit changes from 83.9% to 86.9%, and neighboring-molecule factual correctness remains 80.5%.

I Generality Beyond Chemistry: Protein-Domain Replication

Our main experiments concern molecules, raising the possibility that identifier-dependent knowledge access is specific to SMILES or structural notation. We therefore repeat the key behavioral and causal tests in a qualitatively different domain: proteins. A protein can be addressed through a familiar lexical identifier such as its gene symbol or through an exact UniProt accession (Ahmad et al., 2025) such as `P04637`. Unlike SMILES, an accession is an opaque database key and encodes no biological structure. A similar asymmetry therefore cannot be attributed to difficulty interpreting structural notation.

I.1 Dataset Construction and Evaluation

We query the UniProt REST API (Ahmad et al., 2025) using `(reviewed:true) AND (organism_id:9606)`, requesting accession, primary gene, recommended protein name, subcellular-location annotation, and disease annotation. The data-generation script does not preserve an authoritative UniProt release identifier or download timestamp.

For each returned entry, we retain the primary accession, the primary gene symbol from the first gene record, and the recommended full protein name; entries missing any of these fields are discarded. We additionally extract the first recognized subcellular location and the first disease identifier. An entry is retained if either annotation is available, so a recognized location is not required for inclusion. No explicit accession, gene-symbol, or protein-name deduplication is subsequently applied.

This produces 16,388 entries with 16,388 unique accessions and 16,336 distinct gene symbols. Of these, 15,925 have a recognized subcellular location. ChemDFM-v2.0-14B is evaluated on all 16,388 entries. For the other four models, entries are sorted by accession and the first 1,000 are used; this is deterministic truncation rather than random sampling. The experiment scripts do not preserve a reason for this 1,000-entry limit.

Subcellular location is mapped to the ordered label set

{nucleus, cytoplasm, cell membrane, mitochondrion, endoplasmic reticulum, golgi apparatus, secreted, lysosome, peroxisome, c

After normalization, the mapper returns the first label in this order whose text occurs in a UniProt location annotation; the first recognized annotation is retained. There is no additional alias table or multi-label evaluation. The `cytoskeleton` class does not occur in the resulting data, leaving nine observed labels. In the full set, the largest class is cytoplasm ($4{,}349/15{,}925 = 27.31\%$); among the first 1,000 entries, 979 have locations and the largest class is cell membrane ($281/979 = 28.70\%$).

Prompts and scoring. All behavioral prompts have the form

```
Consider {phrase}.
{question}
Answer in English.  {format} Do not explain.
```

The identifier-dependent phrase is one of

```
the human protein encoded by the gene {symbol}
the human protein {protein_name}
the human protein with UniProt accession {accession}
the human protein with UniProt accession {accession},
encoded by the gene {symbol}
the human protein with UniProt accession {accession},
encoded by the gene {decoy_symbol}.
```

For identification, the question is `What is the official gene symbol for this protein?` with `Reply with the gene symbol only.` Thus, "Symbol ID," "Protein-

Table 28: **Identifier-dependent access also appears for proteins.** Identification columns report recovery of the official gene symbol. "Loc. via Acc." is subcellular-location accuracy through the accession. ID and Loc. Bridge report the fraction of the respective accession–gene-symbol gap recovered when the correct gene symbol is supplied alongside the accession. Decoy ID measures whether identification follows an incorrect gene symbol in a conflicting-identifier prompt. All values except Entries are percentages.

Model	Entries	Symbol ID	Name ID	Accession ID	Loc. via Acc.	ID Bridge	Loc. Bridge	Decoy ID
ChemDFM-v2.0-14B	16,388	99.5	66.7	0.0	23.9	99.9	100.1	99.4
ChemDFM-v1.5-8B	1,000	5.5	18.8	0.0	0.0	20.0	5.6	1.0
Qwen2.5-14B-Instruct	1,000	98.0	60.8	0.0	22.0	100.7	101.0	98.6
Qwen2.5-7B-Instruct	1,000	86.4	51.8	0.0	21.9	98.3	101.4	85.3
Llama-3.1-8B-Instruct	1,000	90.8	60.3	0.0	9.3	98.9	97.6	90.9

name ID," and "Accession ID" all measure recovery of the official gene symbol when conditioning on the corresponding identifier.

For factual retrieval, the question is `What is its primary subcellular location?`, and the model is asked to reply with one of the ten location labels above. Bridge and conflicting-label conditions use the gene symbol as the lexical identifier.

Generation is greedy with at most 32 new tokens and prompt truncation at 1,024 tokens. Models supporting chat templates additionally receive the system message `You are a helpful assistant.`

The scorer rejects empty or refusal-like outputs and normalized responses longer than 12 words. Identification succeeds when the gold gene symbol appears as a case-insensitive whole phrase; no gene-symbol aliases are used. For location, generated text and the stored gold label are lowercased, possessives and punctuation are removed, whitespace is normalized, and light depluralization is applied. A whole-phrase occurrence of the gold label is then counted as correct. If several locations are mentioned, the response therefore succeeds whenever the stored gold label appears.

I.2 Identifier-Dependent Protein Knowledge

Table 28 reports the complete behavioral results. We distinguish bridge recovery for *identification* from bridge recovery for *subcellular location*; these are separate evaluations.

Accession-to-identity accuracy is effectively zero for every model. In contrast, four models identify proteins reliably from gene symbols (86.4–99.5%); ChemDFM-v1.5-8B has substantially weaker lexical protein knowledge. Location retrieval through the accession is likewise weak, ranging from 0.0% to 23.9%.

Supplying the correct gene symbol nearly eliminates this asymmetry for the four models with strong lexical knowledge. For identification, bridge recovery is 98.3–100.7%; for subcellular location it is 97.6–101.4%. Values slightly above 100% occur when the accession-plus-symbol prompt performs marginally better than the pure gene-symbol prompt.

For a fact f, bridge recovery is computed as

$$\mathrm{BridgeRecovery}_f = 100\,\frac{A_{\mathrm{acc+symbol},f} - A_{\mathrm{acc},f}}{\max(A_{\mathrm{symbol},f} - A_{\mathrm{acc},f},\, 10^{-9})}.$$

For example, Qwen2.5-14B-Instruct identifies $0/1000$, $987/1000$, and $980/1000$ proteins from accession, accession-plus-symbol, and symbol prompts, respectively, producing 100.7% identification recovery.

We additionally pair each accession with an incorrect gene symbol. For every entry, the decoy is sampled with seed 0 from the same evaluated population, subject only to having a different gene-symbol string. For the four models with strong lexical knowledge, identification follows this incorrect symbol in 85.3–99.4% of cases. Specifically, the decoy is produced in $16{,}283/16{,}388$, $986/1000$, $853/1000$, and $909/1000$ cases for ChemDFM-v2.0-14B, Qwen2.5-14B-Instruct, Qwen2.5-7B-Instruct, and Llama-3.1-8B-Instruct, respectively. ChemDFM-v1.5-8B follows it in only $10/1000$ cases, consistent with its weak gene-symbol route.

Table 29: **Activation patching in the protein domain.** Peak is the layer maximizing mean normalized recovery after the complete coarse-to-fine sweep; relative depth is peak layer divided by model depth. Rank is evaluated at the coarse-sweep peak (see text). Rank top-1 reports whether the correct location has the highest teacher-forced score among the nine observed location classes.

Model	Items	Peak	Rel. depth	Recovery	Control	Rank layer	Rank top-1
ChemDFM-v2.0-14B	689	47	0.98	89.0	-104.2	40	$11.6 \rightarrow 50.7$
ChemDFM-v1.5-8B	607	25	0.78	109.9	-113.7	28	$12.7 \rightarrow 62.6$
Qwen2.5-14B-Instruct	677	39	0.81	94.6	-125.7	40	$11.8 \rightarrow 55.5$
Qwen2.5-7B-Instruct	674	24	0.86	105.3	-137.3	24	$11.9 \rightarrow 37.8$
Llama-3.1-8B-Instruct	680	31	0.97	88.0	-130.1	24	$11.8 \rightarrow 38.1$

Thus, the access asymmetry and conflicting-identifier behavior do not require a structural representation such as SMILES: they also arise between a familiar lexical identifier and an opaque exact database key.

I.3 Activation Patching in the Protein Domain

We next repeat the causal intervention from Section 4.1. Unlike the behavioral experiment, patching does not select examples according to generated-answer correctness. It begins with accession-sorted entries having a mapped location and uses the raw-completion prompts

$x_{\mathrm{src}} =$ `The human protein encoded by the gene {symbol} is primarily located in the,`

$x_{\mathrm{tgt}} =$ `The human protein with UniProt accession {accession} is primarily located in the`

The gold completion is the canonical location label with a leading space.

For each item, we compute the teacher-forced mean log-probability per gold token under the lexical source and accession target. We retain examples for which the lexical route exceeds the accession route by at least 0.20 nats per token. Examples are sorted by this gap, capped at 80 per gold label, and then capped at 1,000 total examples. Because only nine location labels occur and eligibility varies by model, the resulting sets contain 607–689 examples.

At layer L, we capture the residual-stream output of decoder block `layers[L]` at the *last prompt token* of the lexical source and overwrite the corresponding last-token state in the accession computation. Thus, the intervention patches one residual-stream position after decoder block L; it does not patch the identifier token span itself. Layer indexing is zero-based.

Let s_i and t_i denote source and untouched-target mean gold-token log-probabilities for item i, and $p_{i,L}$ the corresponding score after patching. Normalized recovery is

$$R_{i,L} = \frac{p_{i,L} - t_i}{\max(s_i - t_i, 10^{-6})}.$$

Reported recovery is 100 times the arithmetic mean of $R_{i,L}$ over items, without clipping. Recovery can therefore exceed 100% when the patched target assigns greater probability to the gold continuation than the lexical source itself.

For the control, seed 0 selects another item from the same patching set with a different gold location. Its source state is patched identically, giving

$$C_{i,L} = \frac{c_{i,L} - t_i}{\max(s_i - t_i, 10^{-6})}.$$

The same control assignment is reused across layers. Negative control values therefore indicate that an incompatible entity state reduces the gold probability below that of the untouched accession route.

Patching the matched lexical state yields 88.0–109.9% normalized recovery across all five models, whereas different-location controls strongly decrease the gold score. The intervention therefore provides causal evidence that information produced along the lexical route is sufficient to improve prediction from the otherwise weak accession route.

Ranking analysis. As a complementary threshold-free measure, we score the complete gold strings for every location represented in the patching set using mean teacher-forced log-probability per token. Although the benchmark defines ten possible labels, `cytoskeleton` is absent from the data, so the executed ranking analysis contains nine candidates and uniform chance is 11.1%. The correct location initially ranks first only 11.6–12.7% of the time, but rises to 37.8–62.6% after patching.

The implementation computes this ranking at the *coarse-sweep peak*, rather than at the final dense-sweep maximum. We therefore report its actual evaluation layer separately in Table 29 and do not treat the ranking values as measurements at the final peak layer.

Layer selection and localization. The patching sweep first evaluates layers $0, 4, 8, \ldots$ below each model's depth. It selects the coarse layer with highest mean normalized recovery, then evaluates all previously untested layers within ± 8 layers of this coarse peak. The final reported peak is the layer with maximum mean recovery over the complete saved sweep; rank and control metrics do not influence selection.

The effective layers are substantially later than in chemistry. Chemistry peaks at relative depths 0.50–0.75, whereas protein patching peaks at 0.78–0.98. At these depths—especially 0.97–0.98 for Llama-3.1-8B-Instruct and ChemDFM-v2.0-14B—the intervention occurs close to the output of the network. We therefore interpret the protein experiment as evidence of *causal sufficiency*, not as a replication of the chemistry-specific localization pattern.