EvoMD-LLM: Learning the Language of Species Evolution in Reactive Molecular Dynamics

Anonymous ACL submission

Abstract

While large language models (LLMs) excel at static scientific reasoning, they struggle to model the temporal structure of dynamic physical processes. We present **EvoMD-LLM** (Evolutionary Molecular Dynamics Large Language Model), a framework that reformulates species-level molecular dynamics as a symbolic temporal language modeling problem. Reactive MD trajectories are discretized into sequences of molecular events, where each token represents a chemical species augmented with its persistence duration, enabling standard autoregressive LLMs to learn compositional evolution over time through efficient fine-tuning. A key component of EvoMD-LLM is temporal scaffolding, which treats event duration as an explicit linguistic token and serves as a structured inductive bias, significantly reducing invalid or hallucinated molecular outputs compared to conventional sequence modeling approaches. We evaluate EvoMD-LLM on multiple temporal prediction tasks, achieving up to 68.66% accuracy and consistently outperforming sequential neural networks and language-based baselines. Beyond quantitative improvements, we qualitatively observe that the model is capable of generating interpretations for its own predictions by incorporating relevant chemical knowledge, even though it was not explicitly supervised with paired trajectory-explanation data. These results demonstrate that symbolic temporal language modeling provides an effective framework for grounding LLMs in dynamic physical simulations.

1 Introduction

The convergence of large language models (LLMs) and molecular representations has emerged as a promising direction in AI for Science. Recent paradigms have successfully aligned static molecular encodings, such as SMILES strings (Cavanagh et al., 2024), with natural language, enabling LLMs to support tasks ranging from molecular property prediction (Chithrananda et al., 2020) to retrieval-augmented chemical reasoning (Chen et al., 2025). However, most existing approaches operate on static molecular representations or rely on external tools for reasoning (Boiko et al., 2023). This limits their applicability to physical systems, which evolve over time through sequences of creation, persistence, and transformation events. As a result, enabling LLMs to model temporal physical processes remains a fundamental challenge in AI for Science (Wigh et al., 2022). Molecular dynamics (MD) simulations provide a natural description of temporal physical evolution by recording time-resolved atomic motions (Alder and Wainwright, 1957). Yet, raw MD trajectories consist of high-frequency continuous coordinates that are incompatible with the discrete, symbolic token space of language models. Emerging time-series foundation models (Ansari et al., 2024) remain inapplicable to this challenge, as their numerical quantization schemes destroy the compositional semantics and discrete identity intrinsic to chemical species. Directly aligning MD simulations with LLMs therefore presents a key abstraction challenge: how to represent continuous molecular evolution as symbolic sequences amenable to language modeling. Existing learning-based approaches to MD trajec-

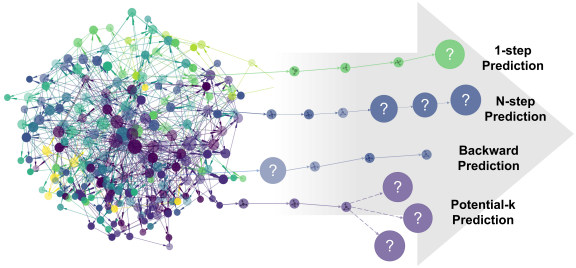

Figure 1: Conceptual overview of EvoMD-LLM. The framework interprets MD trajectories as structured sequences (Nodes: species; Edges: transformations) to reconstruct reaction pathways via four predictive tasks.

tories largely focus on structural dynamics in non-reactive or weakly reactive systems, such as protein folding (Tsai et al., 2020; Bera and Mondal, 2025; Murtada et al., 2024; Hussein Murtada et al., 2025), and are ill-suited for reactive processes characterized by discrete changes in chemical species.

To address this gap, we introduce **EvoMD-LLM** (**Evo**lutionary **M**olecular **D**ynamics **L**arge **L**anguage **M**odel), a framework that reformulates species-level molecular dynamics as a constrained generative language task. We propose a modality alignment scheme that translates continuous trajectories into discrete tokens, where duration serves as an explicit semantic modifier for each chemical species. This representation enables standard autoregressive LLMs to internalize the "grammar" of chemical evolution directly through fine-tuning, eliminating the need for external simulators or specialized architectures.

A key component of EvoMD-LLM is temporal scaffolding, which explicitly encodes event duration as a linguistic token. By treating persistence as a semantic attribute, temporal scaffolding provides a structured inductive bias that encourages the model to distinguish between long-lived stable states and short-lived transient intermediates. Empirical ablation studies show that this design significantly improves prediction accuracy and reduces invalid or hallucinated molecular outputs.

We evaluate EvoMD-LLM on a comprehensive suite of temporal prediction tasks, as illustrated in Figure 1. Beyond quantitative metrics, we remarkably observe that the model exhibits emergent explanatory behaviors: despite lacking explicit supervision, it spontaneously produces plausible physical rationales for kinetic stability. These results demonstrate that symbolic temporal language modeling serves as an effective framework for learning species-level dynamics. Our main contributions are summarized as follows:

- **EvoMD-LLM Framework:** We propose a language modeling framework that reformulates species-level molecular dynamics as symbolic event sequences, enabling standard autoregressive large language models to model temporal evolution in reactive systems.
- **Temporal Scaffolding via Duration Tokens:** We introduce temporal scaffolding by explicitly encoding event persistence as linguistic tokens. This structured inductive bias significantly improves prediction accuracy and reduces invalid molecular outputs, as demonstrated by extensive ablation studies.
- **Unified Temporal Prediction Formulation:** We show that a single instruction-tuned language model can flexibly support diverse temporal prediction tasks, including forward forecasting and backward inference, without task-specific architectures.

2 Methods

We propose EvoMD-LLM to treat molecular evolution as a foreign language with its own grammar of causality and persistence. As illustrated in Figure 2, our framework operates through a four-stage pipeline: (1) Dynamic Modality Alignment; (2) Structured Instruction Formatting; (3) Heterogeneous Task Integration; and (4) Model Training and Inference. In this section, we detail the theoretical formulation and key algorithmic components.

2.1 Problem Formulation

We enable LLMs to learn the dynamics of chemical reactions by reformulating MD simulations as a structured symbolic text generation problem.

A standard MD simulation produces a raw trajectory $\mathcal{T}_{\text{raw}}$, recording atomic positions $\mathbf{R}$ and momenta $\mathbf{P}$ at each time step τ:

$$\mathcal{T}_{\text{raw}} = \{(\mathbf{R}(\tau), \mathbf{P}(\tau)) \mid 0 \leq \tau \leq T_{\text{total}}\}. \quad (1)$$

While physically complete, such trajectories are high-dimensional and dominated by thermal noise, which obscures long-term reaction patterns. To obtain a representation amenable to language modeling, we apply a transformation Φ that maps raw trajectories to a discrete sequence of molecular states:

$$\mathcal{X} = \Phi(\mathcal{T}_{\text{raw}}) = \{(m_i, \Delta t_i)\}_{i=1}^{n}, \quad (2)$$

where each state consists of a semantic unit m_i (molecular formula) and its temporal persistence Δt_i (duration). This abstraction suppresses high-frequency atomic fluctuations while preserving the causal sequence of chemical transformations. Unlike standard text generation where tokens are equidistant, chemical evolution is an irregularly sampled time series. We treat this sequence directly as natural language. This allows us to train the model using standard autoregressive cross-entropy loss, without requiring specialized regression architectures.

Generative Modeling Objective. We formulate reaction modeling as conditional sequence generation. Given a context sequence $\mathbf{x}$ and an instruction $\mathcal{I}$, the model generates a target sequence $\mathbf{y}$ according to the factorization:

$$P(\mathbf{y} \mid \mathbf{x}, \mathcal{I}) = \prod_{j=1}^{|\mathbf{y}|} P(y_j \mid y_{<j}, \mathbf{x}, \mathcal{I}), \quad (3)$$

where $\mathbf{y} = ((m'_1, \Delta t'_1), \ldots, (m'_L, \Delta t'_L))$. The instruction $\mathcal{I}$ specifies the task (e.g., forward or backward prediction), enabling a unified formulation across different reaction reasoning scenarios.

2.2 Dynamic Modality Alignment

To bridge the gap between continuous physical simulations and discrete symbolic reasoning, we construct a Dynamic Modality Interface. This process translates raw MD trajectories into a structured "grammar" of reaction events, characterized by semantic identity and temporal persistence.

From Continuous Trajectories to Discrete Events. Raw MD data consists of high-frequency atomic coordinates dominated by thermal noise. We adopt the ab initio bond-order determination method established by Dang et al. (2025) as the physical ground truth for identifying atomic connectivity. Building upon these snapshots, our framework projects the continuous evolution into a discrete event space by defining molecular formulas as atomic semantic units. Unlike standard NLP approaches that tokenize chemical strings into subword units (e.g., SMILES characters), we treat each distinct molecular formula as an atomic semantic unit. This preserves the integrity of chemical identity, allowing the LLM to reason over species-level transformations rather than character-level statistics.

We define a valid Molecular Event $\mathcal{E} = (m, \Delta t)$ as a tuple comprising a molecular species m and its persistence duration Δt. To distill chemically significant states from transient thermal fluctuations, we apply a temporal band-pass filter, retaining only events with durations $\Delta t \in [10, 500]$ ps. This operation effectively isolates stable reaction intermediates from high-frequency noise. Details about the original dataset scale and filtering statistics are provided in Appendix A.

Structured Context Construction. To enable autoregressive forecasting, the discrete event stream is segmented into structured input-output pairs using a sliding window approach. Each training example consists of a historical context window (3–5 events) and a target future event.

A critical challenge in learning from physical simulations is the long-tail distribution of molecular events. Raw reaction data is often dominated by a few highly stable species (e.g., reactants or final products), while key intermediate transition states appear infrequently. Training directly on such imbalanced data would lead the LLM to degenerate into a trivial identity mapping (predicting the most frequent species). To mitigate this, we employ a stratified sampling strategy that balances the dataset across both molecular types and temporal regimes. This ensures that the model is exposed to a diverse range of reaction pathways and learns to differentiate between rapid intermediates and stable products based on structural context rather than mere occurrence frequency.

Detailed visualizations of the data evolution, species distribution, and the effects of balancing are presented in Appendix A (Figure 5).

2.3 Temporal Scaffolding

Standard Transformers, while adept at sequence ordering, remain agnostic to variable time intervals. To bridge this gap, EvoMD-LLM implements Explicit Temporal Tokenization by interleaving duration tokens Δt_i with species tokens. We reinterpret this strategy as a neural implementation of Run-Length Encoding (RLE). Since raw MD trajectories are dominated by high-frequency thermal fluctuations (e.g., non-reactive vibrations), explicitly tokenizing persistence performs semantic compression, effectively decoupling continuous physical time from the logical reaction sequence. This allows the LLM to skip redundant noise and reason directly across chemically significant timescales.

Theoretically, this design aligns with proven paradigms in variable-duration modalities, such as note sustain in Music Transformers (Huang et al., 2018) and phoneme alignment in FastSpeech (Ren et al., 2019). By treating duration as a semantic attribute, this scaffolding introduces a strong inductive bias: it forces the model to differentiate between thermodynamically stable states (long Δt) and transient intermediates, thereby suppressing "hallucinated" transitions that violate physical kinetic constraints.

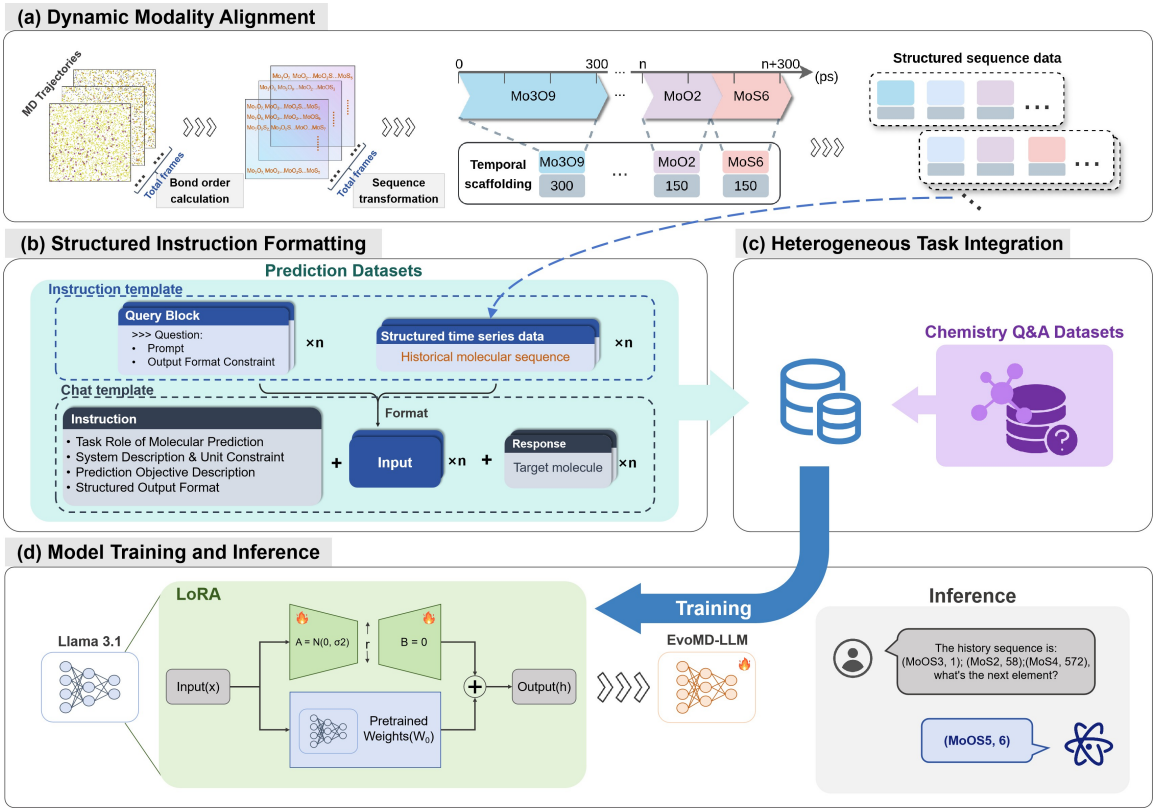

Figure 2: The overall framework of the model. Encompassing dynamic modality alignment, structured instruction formatting, heterogeneous task integration and model training along with inference.

2.4 Structured Instruction Formatting

To transform the interleaved event sequences into training samples, we employ a structured instruction-tuning paradigm. As illustrated in Figure 2(b), we design a domain-specific template that enforces strict syntactic constraints on the generative output.

The construction consists of two components:

1. **System Context (Semantic Definition):** We utilize the system prompt to define the model's role as a "Scientific Simulator." Crucially, this prompt establishes the semantic mapping for our unified vocabulary, explicitly instructing the model that the output must alternate between molecular formulas (representing state identity) and time tokens (representing kinetic stability).

2. **Task Instruction (Historical Constraints):** The user prompt encapsulates the historical context window $x = \mathcal{H}_{<t}$. Unlike open-ended chat, we inject structural constraints into the instruction, limiting the generation search space to valid physical transitions.

By wrapping the raw sequences in this rigorous format, we align the stochastic nature of physical dynamics with the deterministic syntax required for language modeling.

2.5 Heterogeneous Task Integration

To synergize domain-specific dynamic modeling with general scientific reasoning, we construct a heterogeneous instruction dataset comprising two distinct streams as shown in Figure 2(c).

Structured Forecasting Stream. We curate prediction tasks covering 1-step, 2-step, and backward trajectory forecasting. Crucially, we restrict the training horizon to short-term contexts (maximum 2 steps). By mastering local transition rules, the model is forced to acquire temporal inductive reasoning capabilities, enabling it to generalize to long-horizon (N-step) planning during inference without explicit supervision on long sequences.

Linguistic Regularization Stream. While structured forecasting teaches the model the 'syntax' of reaction rules, it risks reducing chemical formulas to arbitrary symbols. To prevent catastrophic forgetting of general capabilities and provide semantic anchoring for chemical tokens, we inter-

leave synthetic Chemistry Q&A pairs. This stream serves as a semantic anchor, forcing the model to ground the symbolic molecular tokens in its pretrained scientific knowledge base. This ensures that EvoMD-LLM evolves into a dual-capable agent: structurally grounded in specific physical dynamics while linguistically aligned with general chemical principles.

2.6 Model Training and Inference

As illustrated in Figure 2(d), we employ a supervised fine-tuning (SFT) framework (Ouyang et al., 2022; Taori et al., 2023; Daniel Han and team, 2023; Touvron et al., 2023), aligning the model to predict target molecular events directly from structured input sequences. SFT enables EvoMD-LLM to internalize domain-specific transition rules into its parameters. The training process is explicitly designed to balance structural precision with linguistic generalization.

Input Representation and Architecture We utilize the Llama 3.1 8B (Meta AI Team, 2024) backbone without architectural modifications. Formulas are tokenized using the standard Byte-Pair Encoding (BPE) (Sennrich et al., 2016) vocabulary. This formatting encourages the model to process chemical formulas as semantic units, leveraging pretrained linguistic priors to model statistical regularities in molecular evolution.

Parameter-Efficient Optimization To align the model with reaction dynamics while preserving general scientific reasoning, we employ Low-Rank Adaptation (LoRA) (Hu et al., 2022). By injecting trainable low-rank matrices into the attention and feed-forward layers while freezing pretrained weights, we achieve two strategic objectives: (1) prevention of catastrophic forgetting, ensuring the retention of the base model's linguistic priors essential for the QA component; and (2) training efficiency, which allows for rapid convergence on consumer-grade hardware by focusing capacity exclusively on modeling temporal chemical patterns.

Optimization Strategy To enable dual competence in structured forecasting and general reasoning, we employ a multi-task sampling strategy: structured prediction datasets and chemistry Q&A instructions (Section 2.5) are interleaved during training. The final loss is computed as the standard autoregressive cross-entropy over the target tokens of both tasks, ensuring the model simultaneously optimizes for domain-specific dynamics and linguistic fluency.

3 Experiments

We evaluate EvoMD-LLM on a range of temporal prediction tasks derived from molecular dynamics trajectories to assess its ability to model symbolic chemical evolution.

3.1 Tasks and Evaluation Protocol

To rigorously assess symbolic dynamic modeling, we utilize the Mo-S reactive system as a testbed. This system serves as a challenging benchmark due to its intrinsic stochasticity and the coexistence of competing reaction pathways (e.g., simultaneous growth and etching), demanding reasoning capabilities beyond simple pattern matching. This section evaluates whether EvoMD-LLM effectively addresses the proposed symbolic–temporal abstraction gap in modeling dynamic chemical systems. Specifically, we evaluate EvoMD-LLM on four temporal prediction tasks: 1-step prediction for short-range consistency, N-step prediction for iterative long-horizon forecasting, backward prediction for bidirectional reasoning of precursor states, and Potential-k prediction to capture the stochastic nature of chemical evolution (Coley et al., 2019).

Performance is quantified using complementary metrics. Prediction accuracy and potential-k accuracy track the presence of ground-truth states within the 1 and k predictions, respectively. These serve as proxies for logical consistency, assessing whether the model captures the valid causal logic of chemical evolution. Conversely, missing rate calculates the proportion of generated outputs that fail to parse as valid molecular formulas, thereby measuring the model's Syntactic Validity and adherence to the chemical grammar. These metrics jointly assess the model's ability to navigate branching chemical reaction pathways while maintaining both structural integrity and instructional adherence.

3.2 Experimental Setup

Baseline Methods To evaluate the effectiveness of EvoMD-LLM, we compare it with four representative categories of baselines for temporal knowledge integration:

- **In-context learning (ICL)** (Dong et al., 2024; Luo et al., 2025): We assess standard prompting capabilities including **zero-shot (ZS)** ($k = 0$) and **few-shot (FS)** ($k = 3$) (Brown et al.,

2020). To probe scalability, we also implement **many-shot** (1,000 examples) (Agarwal et al., 2024) and **full-Context** (∼7,000 examples) to test the upper bound of long-context reasoning.

- **Retrieval-Augmented generation (RAG)** (Lewis et al., 2020; Zhong et al., 2025): A dynamic memory baseline where a retriever selects the k most similar historical subsequences from the training set to provide input-dependent context.

- **Sequential baselines (Numerical Modality)**: To assess the necessity of symbolic abstraction over raw numerical fitting, we consider two representative neural baselines that operate on numerical composition vectors (encoding atomic counts) rather than semantic tokens. Baselines include an LSTM (Hochreiter and Schmidhuber, 1997) and a custom encoder-only Transformer (Vaswani et al., 2017), which map trajectories to latent vectors for direct numerical regression.

Implementation Details Our final instruction-tuning dataset comprises over 20,000 samples, combining 7,015 stratified trajectory sequences with auxiliary scientific Q&A data. Among these samples, the generated RMD symbolic dataset is split into training and test sets at a 9:1 ratio. EvoMD-LLM is initialized with Llama 3.1 8B. We employ LoRA for parameter-efficient fine-tuning, optimizing approximately 42 M parameters ($r = 16, \alpha = 16$). The model is trained for 2 epochs using the AdamW optimizer (Loshchilov and Hutter, 2019) with a global batch size of 8 and a peak learning rate of 2e-4. We utilize a linear learning rate scheduler and mixed-precision (bfloat16) with a maximum sequence length of 2048. All experiments are conducted on a single consumer-grade GPU (NVIDIA RTX 4090D) to demonstrate accessibility.

3.3 Overall Performance Comparison

As shown in Table 1, EvoMD-LLM significantly outperforms all baselines, achieving 68.66% accuracy and a 0% missing rate. In comparison, the RAG baseline achieves only 36.61% accuracy, while zero-shot prompting suffers from a high missing rate (29.49%). Notably, simply extending the context window (Content-1000/All) yields marginal gains over few-shot prompting, with accuracy plateauing around 18%. This saturation suggests that naive long-context prompting fails to effectively capture complex temporal dependencies. Conversely, our SFT approach enables the model to learn structured temporal correlations directly, leading to superior predictive accuracy and stability.

Figure 3(a) presents the confusion matrix of molecular species predicted by the EvoMD-LLM. The strong diagonal dominance indicates robust discriminative capability, with only minor confusion among chemically similar or temporally adjacent species. This highlights the model's ability to capture fine-grained symbolic and temporal distinctions within the molecular event space.

3.4 Multi-step, Backward and Potential-k Prediction Analysis

We further evaluate EvoMD-LLM on N-step, backward, and potential-k prediction tasks to assess its ability to model long-range temporal dependencies and stochastic reaction dynamics.

As shown in Figure 3(b), accuracy decreases monotonically as the prediction horizon increases.While the average accuracy over the 3-step horizon remains high at 50.09% (Table 2), the point-wise accuracy for the final step drops to 37.74%. This trend reflects error accumulation in autoregressive forecasting, while the remaining performance indicates that fine-tuning improves temporal consistency over multiple steps.

Figure 3(c) shows that EvoMD-LLM substantially outperforms the LLaMA 3.1 base model in both potential-k and backward prediction. Potential-1 accuracy increases from 13.96% to 68.66%, and potential-3 accuracy reaches 77.1%, with a smaller gap between the two metrics, indicating more stable candidate ranking. In addition, EvoMD-LLM achieves 57.12% accuracy in backward prediction, demonstrating its ability to infer plausible precursors from downstream states.

3.5 Comparison with Sequential Baseline

Table 2 compares EvoMD-LLM with an LSTM and an encoder-only model. For the 1-step prediction task, EvoMD-LLM achieves the highest accuracy at 68.66%, outperforming both LSTM (40.74%) and the encoder-only baseline (65.95%).

As the prediction horizon increases, all methods exhibit performance degradation. EvoMD-LLM consistently maintains higher accuracy in the 2-step

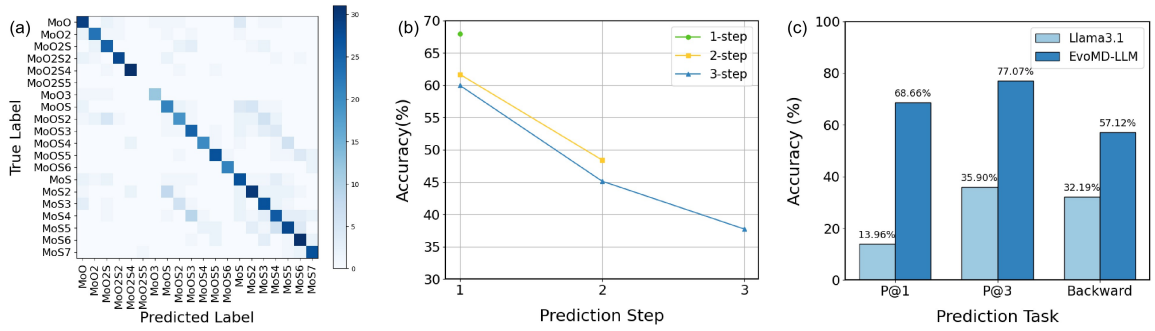

Figure 3: Experimental results. (a) Confusion matrix showing discriminative capability. (b) Accuracy decay over N-step forecasting horizons. (c) Performance comparison against the LLaMA 3.1 base model across three tasks.

Methods	Accuracy ↑	Missing Rate ↓
Zero-shot	13.96	29.49
Few-shot	17.02	5.63
Content-1000	17.28	1.05
Content-All	18.39	0.07
RAG ($k = 5$)	36.61	1.28
EvoMD-LLM	**68.66**	**0.00**

Table 1: Comparison of 1-step accuracy and missing rate between EvoMD-LLM and various LLM-based methods.

Model	**1-step**	**2-step**	**3-step**	**Backward**
LSTM	40.74	32.77	30.23	30.20
Encoder-only	65.95	51.71	36.84	52.71
EvoMD-LLM	**68.66**	**55.06**	**50.09**	**57.12**

Table 2: Performance comparison on molecular forecasting tasks. Reported values for N-step tasks represent the **average accuracy** over the entire prediction horizon (1 to N). Best results are **bolded**.

and 3-step settings, with particularly clear gains in the 3-step task. In the backward prediction task, EvoMD-LLM again outperforms both baselines. Overall, these results indicate that EvoMD-LLM provides more robust temporal modeling across different prediction settings.

3.6 Ablation Studies and Analysis

3.6.1 Impact of Temporal Scaffolding

As detailed in Table 3, excising the duration targets leads to a consistent performance degradation. The 1-step prediction accuracy declines from 68.66% to 55.70%, with a comparable drop in backward reasoning. These results empirically validate that temporal supervision is not merely an auxiliary task but a necessary constraint for correct chemical reasoning.

Model Variant	**1-Step**	**Backward**
w/o Temporal	55.70	49.15
w/o Q&A	65.53	55.98
EvoMD-LLM (Full)	**68.66**	**57.12**

Table 3: Ablation study. The significant drop in *w/o Temporal* validates the necessity of kinetic scaffolding, while excluding Q&A (*w/o Q&A*) has a relatively minor impact on domain-specific forecasting.

3.6.2 Impact of Multi-Task Instruction Tuning

A key design goal of EvoMD-LLM is to improve structured molecular forecasting without degrading general scientific language capabilities. To assess this, we evaluate an ablated variant trained without the chemistry Q&A dataset (w/o Q&A).

We adopt both qualitative and quantitative evaluations. For qualitative assessment, we use Qwen3 (Yang et al., 2025) as an automated evaluator to score model responses across multiple scientific capability and quality dimensions. In addition, we report performance on the AI2 Reasoning Challenge (ARC) benchmark (Clark et al., 2018) to measure standardized scientific reasoning, containing a total of 3548 test samples.

As shown in Figure 4(a–b), removing Q&A supervision leads to consistent degradation across all evaluated dimensions, with the largest drops observed in mechanism reasoning and question answering. The ablated model also exhibits reduced coherence and fluency, suggesting that natural language supervision contributes to stable scientific expression.

Figure 4(c) further shows that EvoMD-LLM maintains strong performance on both ARC-e and ARC-c, comparable to the base LLaMA 3.1 model, while the w/o Q&A variant performs noticeably worse. These results indicate that the Q&A dataset

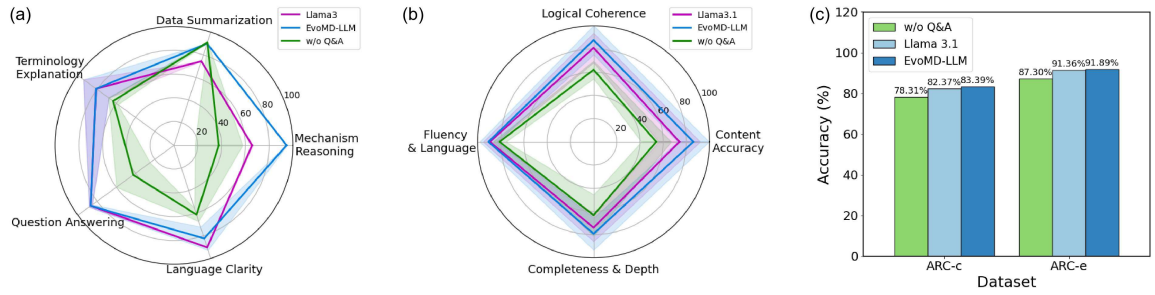

Figure 4: Evaluation of general language understanding. (a–b) present scores from an automated evaluation using Qwen3 as a judge. (a) displays performance across five distinct capability dimensions, while (b) assesses four key criteria for output quality. The shaded areas in the radar charts represent the standard deviation across the evaluation questions. (c) shows the accuracy of the models on the ARC-c (Challenge) and ARC-e (Easy) benchmark datasets.

Input History Sequence	Prediction	Generated Reasoning
Case 1: Trend Following (Oxygen → Sulfur Exchange)		
(MoO,98); ($MoOS_2$,3); (MoS,182)	**Mol:** MoS_3 **Time:** 106 ps	The transition from MoS to MoS_3 is driven by **surface diffusion and adsorption** of additional MoS units. The predicted duration suggests the formation of a **stable three-dimensional structure** favored by thermodynamics.
Case 2: Stability Detection (Distinguishing Stable vs. Transient)		
(MoO_2,2); (MoO,29); ($MoOS_2$,26)	**Mol:** MoS **Time:** 277 ps	The transition is driven by the **decomposition of $MoOS_2$** due to high temperatures. The predicted duration (277 ps) suggests that MoS is **relatively stable**, forming a **metastable trap** at these conditions.
Case 3: Structure Building (Increasing Complexity)		
($MoOS_2$,43); ($MoOS_4$,6); (MoS_3,7)	**Mol:** MoS_5 **Time:** 365 ps	The transition involves the **addition of two sulfur atoms** through a series of elementary reactions. This process reflects the increasing preference for **higher-order coordination structures** in Mo-S systems.

Table 4: Qualitative examples of reasoning generated by EvoMD-LLM. The selected cases demonstrate the model's ability to track linear evolutionary pathways without oscillation.Highlights indicate the textual description of **reaction mechanisms (Red)** and **stability/metastability (Blue)** aligned with temporal cues.

plays an important role in preserving general scientific reasoning during domain-specific fine-tuning.

3.7 Qualitative Analysis

Quantitative metrics summarize predictive accuracy but provide limited insight into model behavior. We therefore present a qualitative analysis to examine the how EvoMD-LLM explains its predictions without explicit supervision on reaction mechanisms. This analysis focuses on the alignment of learned sequence patterns with semantic explanations, rather than on validating ab initio physical correctness.

Table 4 demonstrates the model's context awareness. For instance, it correctly links early-stage sulfidation to surface diffusion (Case 1) while identifying high-temperature decomposition in intermediate phases (Case 2). Furthermore, it utilizes the duration token as a semantic pivot to differentiate between kinetically stable products and metastable traps, as evidenced by the distinct duration predictions. Additional qualitative examples, including typical failure modes, are provided in Appendix G.

4 Conclusion

In this work, we introduce EvoMD-LLM, a framework that re-frames molecular dynamics as a symbolic language modeling problem, thereby internalizing the "grammar" of chemical evolution into LLMs. By aligning continuous physical trajectories with discrete semantic tokens through an Temporal Scaffolding strategy, we enable the model to treat temporal persistence as a semantic component. We demonstrate that this design introduces a robust inductive bias toward temporally consistent generation, leading to improved forecasting accuracy and a substantial reduction in invalid molecular states. More broadly, EvoMD-LLM highlights the potential of language-based models as general-purpose sequence learners for scientific simulations, suggesting a promising direction for bridging linguistic abstraction with time-resolved molecular dynamics in AI-driven materials discovery.

Limitations

While EvoMD-LLM demonstrates promising capabilities in modeling symbolic chemical evolution, several limitations remain to be addressed in future work:

Generalization to Unseen Chemical Spaces. Our evaluation focuses on the Mo-S CVD system. We selected this system not merely for data availability, but as a representative "complex prototype" of inorganic synthesis: it features high-degree stochastic branching, reversibility, and multi-phase transitions (nucleation, etching, growth), which are often absent in linear organic reaction datasets. However, extending this framework to diverse chemical spaces, including heterogeneous biological systems, remains an open challenge for future scaling.

Autoregressive Error Accumulation. As observed in N-step prediction tasks, the model suffers from error accumulation typical of autoregressive generation, leading to performance degradation over long horizons. Unlike numerical solvers that strictly enforce conservation laws, the current probabilistic generation may occasionally drift into physically invalid states. Integrating physical constraints (e.g., mass conservation or energy consistency) directly into the loss function could mitigate this issue in future iterations.

Loss of Fine-Grained Geometry. While our coarse-grained symbolic representation efficiently filters thermal noise and captures high-level reaction logic, it inevitably discards fine-grained conformational information (e.g., precise bond lengths and angles). Consequently, EvoMD-LLM is currently less suitable for tasks requiring exact geometric verification. Future work could explore multimodal architectures that jointly model symbolic evolution and geometric deformation to achieve fully comprehensive dynamic reasoning.

Interpretability and Hallucination Risks. The explanatory outputs provided by EvoMD-LLM are derived from aligning trajectory patterns with scientific knowledge learned during training, rather than ab initio derivation. Consequently, explanations often rely on plausible but geometrically ungrounded terminology, suggesting retrieval-based association driven by pre-trained priors. Furthermore, the model occasionally over-interprets stochastic cues as deterministic stability guarantees and defaults to generic linear narratives for rare intermediates, reflecting reduced sensitivity in low-frequency regimes.

References

Rishabh Agarwal, Avi Singh, Lei M. Zhang, Bernd Bohnet, Luis Rosias, Stephanie C. Y. Chan, Biao Zhang, Ankesh Anand, Zaheer Abbas, Azade Nova, John D. Co-Reyes, Eric Chu, Feryal M. P. Behbahani, Aleksandra Faust, and Hugo Larochelle. 2024. Many-shot in-context learning. *Advances in Neural Information Processing Systems*, 37:76930–76966.

Berni Julian Alder and Thomas Everett Wainwright. 1957. Phase transition for a hard sphere system. *The Journal of chemical physics*, 27(5):1208.

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Yuyang Wang. 2024. Chronos: Learning the language of time series. *Transactions on Machine Learning Research*.

Palash Bera and Jagannath Mondal. 2025. Accurate prediction of the kinetic sequence of physicochemical states using generative artificial intelligence. *Chemical Science*, 16(20):8735–8751.

Daniil A Boiko, Robert MacKnight, and Gabe Gomes. 2023. Emergent autonomous scientific research capabilities of large language models. *arXiv preprint arXiv:2304.05332*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. volume 33, pages 1877–1901.

J. Cavanagh, K. Sun, A. Gritsevskiy, D. Bagni, T. Head-Gordon, and T. D. Bannister. 2024. Smileyllama: Modifying large language models for directed chemical space exploration. In *NeurIPS 2024 Workshop on AI for New Drug Modalities*.

Xiuying Chen, Tairan Wang, Taicheng Guo, Kehan Guo, Juexiao Zhou, Haoyang Li, Zirui Song, Xin Gao, and Xiangliang Zhang. 2025. Unveiling the power of language models in chemical research question answering. *Communications Chemistry*, 8(1):4.

Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. 2020. Chemberta: Large-scale self-supervised pretraining for molecular property prediction. *CoRR*, abs/2010.09885.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *CoRR*, abs/1803.05457.

Connor W Coley, Wengong Jin, Luke Rogers, Timothy F Jamison, Tommi S Jaakkola, William H Green, Regina Barzilay, and Klavs F Jensen. 2019. A graph-convolutional neural network model for the prediction of chemical reactivity. *Chemical science*, 10(2):370–377.

OpenCompass Contributors. 2023. Opencompass: A universal evaluation platform for foundation models. https://github.com/open-compass/opencompass.

Zhengzheng Dang, Zhichen Tang, Jixin Wu, Yide Chang, and Yanming Wang. 2025. Unraveling the reaction networks and key pathways during the gas phase stage in cvd synthesis of mos2. *Chemical Engineering Journal*, 503:157957.

Michael Han Daniel Han and Unsloth team. 2023. Unsloth.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, Xu Sun, and Zhifang Sui. 2024. A survey on in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1107–1128.

S. Hochreiter and J. Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Cheng-Zhi Anna Huang, Ashish Vaswani, Jakob Uszkoreit, Ian Simon, Curtis Hawthorne, Noam Shazeer, Andrew M Dai, Matthew D Hoffman, Monica Dinculescu, and Douglas Eck. 2018. Music transformer: Generating music with long-term structure. In *International Conference on Learning Representations*.

Mhd Hussein Murtada, Z Faidon Brotzakis, and Michele Vendruscolo. 2025. Md-llm-1: A large language model for molecular dynamics. *arXiv e-prints*, pages arXiv–2508.

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

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Feifei Luo, Jinglang Zhang, Qilong Wang, and Chunpeng Yang. 2025. Leveraging prompt engineering in large language models for accelerating chemical research. *ACS Central Science*, 11(4):511–519.

Meta AI Team. 2024. The llama 3 herd of models: the llama 3.1 family (405b parameters, 128k context window). Technical report. Technical Report.

Mhd Hussein Murtada, Z Faidon Brotzakis, and Michele Vendruscolo. 2024. Language models for molecular dynamics. *bioRxiv*, pages 2024–11.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Yi Ren, Yangjun Ruan, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. 2019. Fastspeech: Fast, robust and controllable text to speech. *Advances in neural information processing systems*, 32.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Stanford alpaca: An instruction-following llama model.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, and 49 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Sun-Ting Tsai, En-Jui Kuo, and Pratyush Tiwary. 2020. Learning molecular dynamics with simple language model built upon long short-term memory neural network. *Nature communications*, 11(1):5115.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Daniel S Wigh, Jonathan M Goodman, and Alexei A Lapkin. 2022. A review of molecular representation in the age of machine learning. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 12(5):e1603.

A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, and Z. Qiu. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Xianrui Zhong, Bowen Jin, Siru Ouyang, Yanzhen Shen, Qiao Jin, Yin Fang, Zhiyong Lu, and Jiawei Han. 2025. Benchmarking retrieval-augmented generation for chemistry. In *Second Conference on Language Modeling*.

A Data Processing and Statistics

In this section, we provide a detailed breakdown of the data processing pipeline, statistical characteristics, and the balancing strategy visualized in Figure 5.

A.1 Event Extraction and Filtering Pipeline

Our molecular event sequences originate from Reactive Molecular Dynamics (RMD) simulations of MoS_2 synthesis, as reported by Dang et al. (2025). The raw trajectories capture high-frequency atomic motions that do not directly correspond to symbolic chemical reactions. To align this data with language modeling, we employed a multi-stage pipeline:

- **Raw Extraction (Step 1):** We utilized a Depth-First Search (DFS) algorithm based on bond-order cutoffs to identify molecular species at every frame. This yielded an initial "Raw MD" dataset comprising 1,648,646 events (Figure 5(a)).
- **Thermal Noise Reduction (Step 2):** Raw trajectories are dominated by transient thermal fluctuations where bonds vibrate but do not break. We filtered out all events with a persistence duration of $\Delta t < 1$ ps. This resulted in the "Extracted" dataset of 237,673 events.
- **Temporal Band-Pass Filtering (Step 3):** To focus on the primary dynamic scales relevant for reaction forecasting, we further refined the dataset by retaining only events with durations $\Delta t \in [10, 500]$ ps. This step removes extremely short-lived noise while segmenting ultra-long stable states, resulting in the "Filtered" dataset of 130,900 events.

A.2 Handling Data Imbalance

A critical challenge in MD data is the long-tail distribution of species. As shown in the word cloud in Figure 5(c) and the "Origin Data" histograms in Figure 5(b), a few dominant species (e.g., reactants like MoS_x precursors) account for the vast majority of observations, while critical transition states are rare.

Training a language model directly on this skewed distribution leads to trivial solutions where the model simply memorizes the most frequent tokens. To address this, we applied stratified sampling to balance the dataset across both molecular identity and duration intervals.

- **Effect of Balancing:** Figure 5(b) (bottom panels) demonstrates that after sampling, the distributions of both molecular types and event durations become significantly more uniform.
- **Final Dataset:** This process yielded the final "Balanced" dataset containing 7,015 high-quality sequence pairs, which were used for fine-tuning EvoMD-LLM.

A.3 Temporal Characteristics

Figure 5(d) presents box plots of the existence durations for various molecular species in the final processed dataset. The distinct temporal distributions (e.g., some species consistently show shorter lifetimes than others) confirm that duration is a semantic property intrinsic to each chemical species, justifying our use of Temporal Scaffolding to capture these kinetic signatures.

B Implementation Details

To ensure the reproducibility of EvoMD-LLM, we provide detailed specifications of our software environment, hardware infrastructure, and training configurations.

B.1 Software and Hardware Environment

We implemented EvoMD-LLM using the `Unsloth` framework, which optimizes memory usage and training speed for Llama-based models. The core software dependencies include:

- **Python**: 3.10
- **PyTorch**: 2.7.0 (with CUDA 12.6)
- **Unsloth**: 2025.5.6
- **Transformers**: 4.51.3
- **Peft**: 0.15.2

All experiments were conducted on a single consumer-grade NVIDIA RTX 4090D (24GB

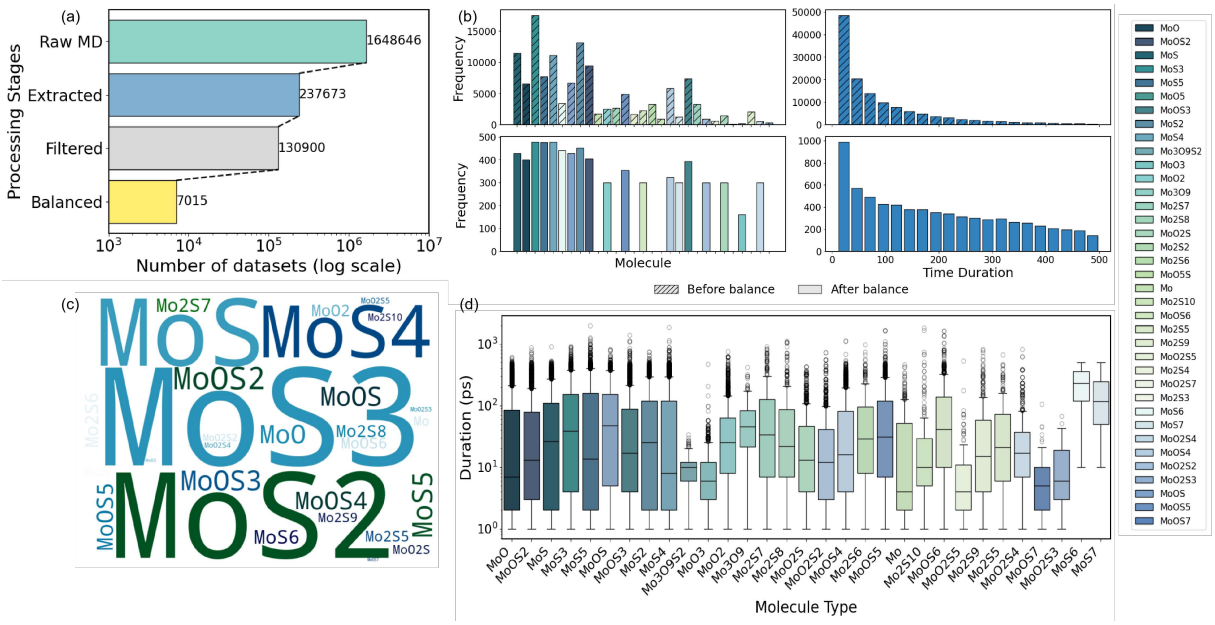

Figure 5: Data processing visualization. (a) Evolution of dataset scale across successive preprocessing stages, showing the reduction from raw events to high-quality balanced sequences. (b) Histograms comparing molecular type and duration distributions before (Origin) and after (Processed) stratified sampling, highlighting the mitigation of the long-tail problem. (c) Word cloud visualizing the dominance of specific species in the raw dataset. (d) Box plots showing the distinct distribution of existence durations (in ps) for different molecular species, reflecting their varying kinetic stabilities.

VRAM) GPU. We utilized mixed-precision training (`bfloat16`) to maximize computational efficiency without compromising numerical stability.

B.2 Details on Quantitative Evaluation

All reported quantitative results are computed as the mean over multiple runs with different random seeds to ensure robust and reproducible evaluation. Specifically, for each experiment, we performed 10 runs with distinct random seeds and report the averaged metrics.

For the additional reasoning assessment on the ARC benchmark, we utilized the OpenCompass evaluation platform (Contributors, 2023). The model was evaluated using the `HuggingFaceCausalLM`.

B.3 Hyperparameters and Training Costs

The detailed hyperparameters used for fine-tuning are listed in Table 5, optimized by grid search. We employed the LoRA technique, targeting all linear layers in the attention and feed-forward blocks.

With the configuration specified below, the full training process (covering both structured forecasting and Q&A tasks) took approximately 2.5 hours. The peak memory usage was controlled under 16GB thanks to the 4-bit quantization support from Unsloth during the gradient calculation.

B.4 Licenses and Terms of Use.

We use the LLaMA 3.1 model released by Meta under the LLaMA community license. The ARC-e and ARC-c benchmarks are publicly available for research purposes. All reactive molecular dynamics simulations and derived symbolic datasets were generated by the authors and do not contain personal or sensitive information. We plan to release the processed datasets and model checkpoints under a permissive research license upon acceptance.

C Prompt Templates

In this section, we present the exact prompt templates used for training EvoMD-LLM and for eliciting qualitative reasoning. We employed a consistent system prompt to define the model's role, while task-specific instructions were appended to the user queries to constrain the output format.

C.1 Training and Prediction Prompts

For the supervised fine-tuning (SFT) stage and standard prediction tasks (1-step, N-step, and Backward), we used the following template structure.

System Message. This prompt sets the general behavioral constraints and defines the data format

Table 5: Detailed hyperparameters and training configuration for EvoMD-LLM. The model was fine-tuned using the LoRA method with the Unsloth framework for memory optimization.

Hyperparameter	Value
General Configuration	
Base Model	Llama 3.1 8B Instruct
Framework	Unsloth (TRL)
Precision	bfloat16 (bf16)
Random Seed	3407
Max Sequence Length	2048
Optimization	
Optimizer	AdamW (8-bit)
Learning Rate	2×10^{-4}
Weight Decay	0.01
LR Scheduler	Linear
Warmup Steps	400
Num Epochs	2
Batch Size Configuration	
Per-Device Batch Size	2
Gradient Accumulation Steps	4
Effective Batch Size	8
LoRA Configuration	
Rank (r)	16
Alpha (α)	16
Dropout	0
Bias	None
Target Modules	q, k, v, o, gate, up, down_proj

(molecule, duration).

> You are an AI assistant to help me predict molecular sequence progression based on given molecular compositions and their existence durations and analysis. Each data point consists of a molecule and the duration it persists in the system, the unit of duration is ps. If the question is about predicting molecular sequences, format your answer as (molecule, time). Otherwise, answer normally.

Task-Specific Instructions. Different prediction tasks are distinguished by specific suffixes appended to the historical sequence.

1. Single-Step Prediction (Forward):

> **Input:** The history sequence is {SEQUENCE_HISTORY}, What is the next element? Output ONLY the next element in the format: (molecule, time). No explanation. No code. No extra words!

2. Multi-Step Prediction (N=2):

> **Input:** The history sequence is {SEQUENCE_HISTORY}, What are the next two elements? Output ONLY the next two elements in the format: (molecule, time). No explanation. No code. No extra words!

3. Backward Prediction:

> **Input:** The history sequence is {SEQUENCE_HISTORY}, What is the previous element? Output ONLY the previous element in the format: (molecule, time). No explanation. No code. No extra words!

D Reasoning and Explanation Prompts

To assess the emergent explanatory capabilities of EvoMD-LLM, we utilized a structured prompt designed to constrain the output format.

It is important to note that while the model was fine-tuned with a mixture of symbolic MD sequences and general scientific Q&A pairs (to prevent catastrophic forgetting, see Section 2.5), it was never supervised on paired samples of (trajectory, textual explanation).

The training data for MD trajectories consisted solely of symbolic sequences (e.g., molecule tokens and duration values). Therefore, the detailed reasoning elicited by the prompt below reflects the model's emergent ability to ground its general chemical knowledge (acquired from pre-training and Q&A regularization) into the specific context of the learned physical dynamics.

The prompt acts as a structural scaffold, directing the model to articulate its learned sequence patterns into explicit linguistic reasoning.

Expert Simulator System Context.

> You are an expert scientific simulator specializing in Reactive Molecular Dynamics (RMD) for Chemical Vapor Deposition (CVD) synthesis.
>
> **System Context:** The reaction system involves the sulfidation of Mo_3O_9 precursors by S_2 gas. Key dynamics include Oxygen-Sulfur exchange, structural relaxation, and thermal decomposition.
>
> **Task Definition:** Your goal is to forecast the trajectory of chemical evolution. Each data point (Molecule, Duration) represents a distinct chemical state and its kinetic persistence (stability).
>
> - A short duration implies a transient intermediate or transition state.
> - A long duration implies a thermodynamically stable product or metastable trap.

Reasoning Instruction. After the model generates a prediction, we prompt it to explain the physical rationale using the following template:

> **Task:** You are provided with a historical trajectory of molecular species and their durations.

History Sequence: {history_seq}
Your Model Prediction: ({predict_res})

Instructions: Provide a scientific explanation for this transition. Your response must:

1. **Mechanism:** Analyze the change in stoichiometry from the last history step to the predicted step. What specific chemical process drives this transformation?
2. **Stability:** Analyze the predicted duration ({duration}). What does this specific timescale imply about the thermodynamic state or kinetic stability of the predicted molecule?
3. **Format:** Write in strict, concise Academic English.

Your answer must be in academic English, concise, and only include the reasoning (no extra content, no repetition).

E Sample Efficiency Analysis

To investigate whether our dataset size is a bottleneck for performance, we conducted a scaling analysis by training EvoMD-LLM on subsets of the training data ranging from roughly 100 to 20,000 samples. Figure 6 illustrates the 1-step prediction accuracy as a function of data quantity.

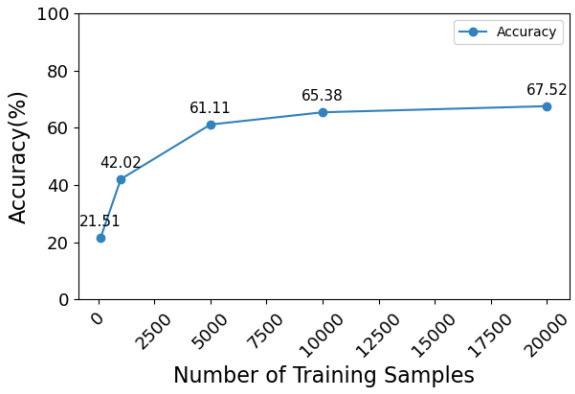

Figure 6: Learning curve of EvoMD-LLM. The plot shows 1-step prediction accuracy scaling with training data size. The model exhibits strong few-shot generalization, reaching over 60% accuracy with only 5,000 samples, and shows signs of performance saturation beyond 10,000 samples, indicating that the current dataset size is sufficient for capturing the core dynamics.

As shown in Figure 6, the model exhibits high sample efficiency.

- **Rapid Syntax Acquisition (0-5k):** Accuracy surges from 21.5% to 61.1% within the first 5,000 samples. This steep rise suggests that the LLM, leveraging its pre-trained capabilities, rapidly aligns with the "grammar" of molecular evolution (syntax and basic stoichiometry) with minimal data.

- **Performance Saturation (10k-20k):** As data volume doubles from 10,000 to 20,000, accuracy gains moderate (from 65.4% to 67.5%). This plateau indicates that the model has effectively captured the majority of the learnable patterns within the current domain.

This analysis confirms that our dataset size ($\sim$20k total samples) is robust. The constraint on further performance improvement is likely not the quantity of raw data, but the inherent stochasticity of the chemical system itself.

F Detailed Error Analysis

To evaluate the reliability of EvoMD-LLM beyond standard accuracy metrics, we conducted a fine-grained analysis of the prediction errors on the test set.

F.1 Zero Hallucinations and Chemical Validity

A critical finding of our analysis is that EvoMD-LLM exhibits zero hallucinations regarding chemical validity. 100% of the incorrect predictions correspond to chemically valid molecular formulas that exist within the reaction network (e.g., predicting a valid intermediate like MoS_3 but at an incorrect time step).

This stands in sharp contrast to generic LLMs, which often generate physically impossible stoichiometry when fine-tuned on scientific data. The absence of hallucinations confirms that our symbolic tokenization strategy has successfully grounded the model in the compositional grammar of the chemical system, constraining its errors to the domain of physical kinetics rather than generative syntax.

F.2 Physical Symmetry of Kinetic Mismatch

Since all errors are valid "Kinetic Mismatches," we further decomposed them to determine if the model exhibits systematic bias. Given that the primary reaction mechanism is sulfidation (replacing Oxygen with Sulfur), we classified the errors based on the stoichiometry of the predicted species relative to the ground truth:

- **Under-Sulfidation (Lagging):** The predicted molecule contains fewer Sulfur atoms than the ground truth ($S_{pred} < S_{true}$). The model predicts a precursor state, effectively "lagging" behind the true trajectory.

- **Over-Sulfidation (Too Fast):** The predicted molecule contains more Sulfur atoms ($S_{pred} > S_{true}$). The model anticipates the reaction progressing faster than reality.

- **Oxygen Deviation:** The Sulfur content is correct, but the Oxygen stoichiometry differs, reflecting minor inaccuracies in secondary deoxidation steps.

As illustrated in Figure 7, the error distribution is remarkably balanced. The near-perfect symmetry between Under-Sulfidation (45.0%) and Over-Sulfidation (44.1%) suggests that EvoMD-LLM does not suffer from a systematic drift towards either accelerating or retarding the reaction kinetics. Instead, the errors likely reflect the inherent stochasticity of Molecular Dynamics simulations, where the exact timing of atomic transitions fluctuates around the mean path. The model effectively captures the central tendency of the reaction, with errors representing unbiased variance.

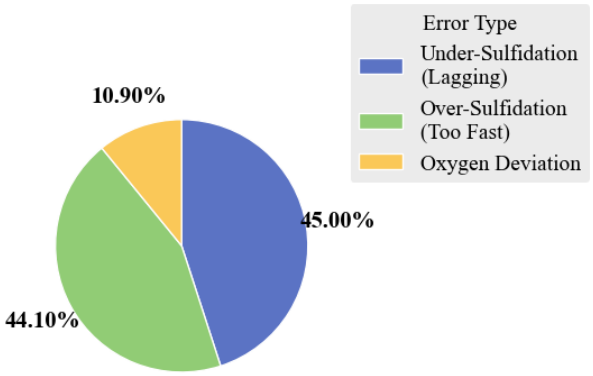

Figure 7: Breakdown of Kinetic Mismatch Errors. The distribution shows a balanced split between Under-Sulfidation (Blue, 45.0%) and Over-Sulfidation (Green, 44.1%). This symmetry indicates that the model's temporal errors are stochastic rather than biased. Oxygen Deviation (Yellow, 10.9%) represents minor stoichiometric noise.

G Additional Qualitative Examples

To provide a more comprehensive view of the model's qualitative behavior, we present additional prediction and reasoning examples in Table 6.Unlike the selected cases in the main paper, these examples include both successful predictions and characteristic failure modes, and are intended to illustrate typical patterns rather than exhaustive coverage.

Input History Sequence	Prediction	Generated Reasoning
Case 1: Over-sulfidation Drift		
(MoS$_4$,8); (Mo$_2$S$_7$,12); (MoS$_3$,41)	**Mol:** MoS$_6$ **Time:** 10 ps	The transition is attributed to **continued sulfur adsorption** leading to higher sulfur coordination. However, the extremely short predicted duration suggests a **transient and unstable configuration**.
Case 2: Stability Overestimation		
(Mo$_2$S$_5$,7); (MoS$_3$,19); (MoS$_4$,32)	**Mol:** MoS$_4$ **Time:** 125 ps	The model interprets repeated appearance of MoS$_4$ as evidence of **kinetic stability**, potentially overestimating persistence due to frequency bias.
Case 3: Reduction Shortcut		
(MoOS,4); (MoO$_2$,9); (MoO$_2$,12)	**Mol:** MoO **Time:** 488 ps	The transition is explained as **oxygen loss under reducing conditions**. The long duration reflects a **stable oxide plateau**, though intermediate pathways are omitted.
Case 4: Missed Cluster Growth		
(Mo$_2$S$_7$,5); (Mo$_2$S$_5$,11); (MoS$_4$,18)	**Mol:** MoS$_4$ **Time:** 495 ps	The model favors **structural persistence** over further aggregation, overlooking potential **cluster growth** toward higher sulfur content.
Case 5: Correct Stability Tracking		
(MoS$_4$,2); (MoOS$_4$,9); (MoOS$_5$,14)	**Mol:** MoOS$_5$ **Time:** 62 ps	The prediction reflects **incremental sulfur incorporation** while correctly identifying a **moderately persistent intermediate**.
Case 6: Linear Growth Bias		
(MoS$_3$,6); (MoS$_4$,11); (MoS$_5$,9)	**Mol:** MoS$_6$ **Time:** 83 ps	The reasoning assumes monotonic sulfur accumulation, exhibiting a **linear growth bias** that neglects possible sulfur loss or rearrangement.
Case 7: Transient Misclassification		
(MoO$_2$,3); (MoOS$_2$,7); (MoS$_2$,5)	**Mol:** MoS$_2$ **Time:** 214 ps	Although MoS$_2$ is predicted as long-lived, the explanation likely overstates **kinetic trapping** for what may be a transient configuration.
Case 8: Rare Species Generalization		
(Mo$_3$S$_{11}$,4); (Mo$_3$S$_{12}$,6); (Mo$_3$S$_{13}$,3)	**Mol:** Mo$_3$S$_{13}$ **Time:** 57 ps	The model produces a generic explanation invoking **coordination saturation**, reflecting limited specificity for **rare cluster species**.
Case 9: Oxygen Retention Bias		
(MoO$_3$,5); (MoO$_2$,18); (MoOS$_2$,7)	**Mol:** MoO$_2$ **Time:** 301 ps	The prediction favors oxygen-rich species, suggesting a bias toward **oxide persistence** despite emerging sulfidation signals.
Case 10: Competing Pathway Suppression		
(MoS$_2$,14); (MoOS$_3$,9); (MoS$_3$,11)	**Mol:** MoS$_4$ **Time:** 92 ps	The explanation emphasizes sulfur addition while suppressing alternative **desulfurization or rearrangement pathways**.

Table 6: Additional qualitative examples generated by EvoMD-LLM. These cases complement the main-paper examples by covering a broader range of behaviors, including correct predictions, stability overestimation, linear growth bias, and generic reasoning for rare species. Highlights indicate inferred reaction mechanisms (**Red**) and stability or metastability judgments (**Blue**).