Orchestrating Symbolic and Sub-Symbolic Reasoning: A Multi-Agent LLM Framework for Complex Scientific Problem-Solving

Mahule Roy[1]
Subhas Roy[2]
[1]University of Oxford
[2]TATA Consumer Products Limited

Abstract

Large Language Models (LLMs) have demonstrated remarkable capabilities in pattern recognition and text generation, yet they face significant challenges in complex logical and symbolic reasoning tasks. This limitation becomes particularly evident in scientific domains that require the integration of symbolic knowledge with sub-symbolic computation. To address these challenges, we present a multi-agent LLM framework where a central orchestrator (DeepSeek-V2) performs **multi-turn interactions** to coordinate a team of specialized agents, effectively using them as **external tools** for different reasoning modalities. **We treat polymer science as a representative structured scientific testbed, illustrating how multi-agent reasoning frameworks can integrate symbolic constraints and numerical inference across diverse domains requiring formal logical consistency.** Our framework demonstrates **multi-agent reasoning** through dynamic team formation and consensus mechanisms, while maintaining **logical consistency** via cross-agent verification protocols. We evaluate this system on polymer science—a domain rich with symbolic constraints and numerical data—showing significant performance improvements (0.76 success rate vs. 0.62 for single LLM) and robust task completion (100% in a 5-paper benchmark). However, a detailed failure case in biopolymer analysis reveals critical challenges in maintaining consistency across reasoning modalities, highlighting the need for more sophisticated verification mechanisms. Our work provides a blueprint for enhancing LLM reasoning through coordinated multi-agent systems and identifies key directions for future research in logical reasoning augmentation.

Introduction

Large Language Models (LLMs) have demonstrated remarkable capabilities in natural language processing, yet they exhibit fundamental limitations in **logical and symbolic reasoning** (Brown et al., 2020; Wei et al., 2022). Their next-word prediction training paradigm does not inherently foster robust **logical deduction**, and their pre-training corporia often lack rigorous formal proofs and systematic reasoning chains. This results in challenges in maintaining **logical consistency** across complex inferences and properly handling implications and contradictions. Polymer science serves as an ideal testbed for these **logical reasoning** challenges due to its inherent complexity. Reasoning in this domain requires the integration of symbolic chemical knowledge with physical constraints and multi-scale relationships—a setting that demands sophisticated **logical inference** across multiple modalities (Chen et al., 2020; Kim et al., 2018). **We use polymer science as a rigorous testbed for evaluating multi-modal logical reasoning frameworks that integrate formal symbolic systems with sub-symbolic computation.** Current approaches to enhance LLM reasoning—such as chain-of-thought prompting (Wei et al., 2022) and external tool-use (Bran et al., 2023)—still struggle to ensure **logical consistency** in multi-step, multi-modal workflows. To address this, we introduce a unified multi-agent framework that implements **distributed logical reasoning** through structured interactions and formal verification. Our work contributes: (1) a system ensuring **logical consistency** via cross-agent verification and symbolic constraints; (2) a formal analysis of reasoning patterns and **logical failure** modes; and (3) a demonstration of how integrated reasoning systems yield more robust AI. This advances both polymer informatics and fundamental **logical reasoning** capabilities in LLMs.

A Framework for Multi-Turn, Tool-Using LLM Reasoning

Our framework addresses the limitations of monolithic LLMs by distributing reasoning tasks across specialized agents coordinated through structured interactions. This architecture explicitly separates different reasoning modalities while maintaining coherence through verification protocols.

System Architecture & The Orchestrator LLM

At the core of our system is a central orchestrator LLM (DeepSeek-V2) that manages **multi-agent reasoning** through a publish-subscribe pattern implemented via LangChain. The orchestrator's primary function is to parse complex problems, decompose them into subtasks, and coordinate the sequence of **multi-turn interactions** with specialized agents. As shown in Figure 1, this architecture enables the dynamic formation of agent teams based on problem complexity and required expertise. The orchestrator employs **chain-of-thought reasoning** to maintain transparency in its decision-making process, generating explicit reasoning traces for task

decomposition and agent selection. This approach allows the system to handle complex workflows that would exceed the capabilities of any single agent.

External Tool-Use as Specialized Agents

Our framework implements **external tool-use** through a collection of specialized agents, each serving as a dedicated reasoning module:

Symbolic Reasoning Tools The **Knowledge Graph Agent** (Neo4j with 15,000+ polymer entities) enables **symbolic expressions and reasoning** through structured querying of chemical relationships and constraints. This agent maintains a polymer ontology that encodes symbolic knowledge about chemical structures, properties, and synthesis pathways. The **Validation Agent** implements rule-based checking for **logical consistency**, enforcing chemical feasibility constraints (e.g., valid SMILES syntax, molecular weight ranges, stable functional groups) and physical bounds (e.g., T_g between -150°C and 300°C). This agent serves as the system's "chemical conscience," preventing logically inconsistent or physically impossible predictions.

Sub-Symbolic Prediction Tools The **PolyGNN Agent** implements graph neural networks for molecular property prediction, translating symbolic SMILES representations into numerical predictions through learned embeddings. This agent demonstrates how **symbolic expressions** (chemical structures) can be processed using sub-symbolic methods while maintaining interpretability. The **Physics-Informed Neural Network (PINN) Agent** incorporates physical laws directly into the reasoning process through loss function regularization ($\mathcal{L}_{total} = \mathcal{L}_{data} + \lambda\mathcal{L}_{physics}$). This approach ensures that predictions respect known physical constraints, enhancing **logical consistency** with domain knowledge.

Simulation & Analysis Tools The **RadonPy Agent** provides molecular dynamics simulations as an **external tool** for physics-based estimation, while the **Vision Agent** (CLIP-ViT-L/14 + Grounding DINO) enables multi-modal reasoning by interpreting structural plots and visual data. The **AlphaFold Integration Agent** handles protein structure prediction, demonstrating the framework's ability to integrate specialized scientific tools.

Multi-Turn Interactions for Workflow Execution

The framework implements structured multi-turn interactions through JSON-based messaging with schema validation, following a precise protocol: problem parsing and requirement analysis by the orchestrator, agent capability matching and team formation, sequential task execution with intermediate validation, consensus building through weighted voting, and final output validation against domain constraints. This multi-turn approach allows the system to maintain context across reasoning steps, adapt to intermediate results, and recover from partial failures—capabilities essential for complex logical reasoning tasks.

Formal Logical Modeling of Multi-Agent Reasoning

We model our system as a distributed logical reasoning framework in which symbolic and sub-symbolic agents contribute partial inferences that must satisfy global consistency. Each agent A_i produces an assertion ϕ_i, with symbolic agents generating first-order logic (FOL) formulas and sub-symbolic agents providing probabilistic soft constraints. A shared belief state Γ is updated only if integrating ϕ_i maintains $\Gamma \not\models \bot$, with the orchestrator acting as a meta-reasoner. Consistency is enforced through a Distributed Logical Verification step, where a qualified majority of agents must agree before accepting ϕ_i, followed by minimal belief revision if contradictions arise. Sub-symbolic outputs $P(\phi_i)$ are fused as soft evidence, with symbolic constraints taking precedence. This abstraction positions our architecture as a general logic-grounded multi-agent reasoning model rather than a domain-specific system.

Protocols for Ensuring Logical Consistency

Our framework ensures logical consistency across heterogeneous reasoning agents through four core mechanisms. **Formal Constraint Propagation** translates symbolic constraints into logical predicates, enforcing compliance with domain laws. **Cross-Agent Logical Verification** employs a distributed consensus protocol (3/4 majority) to resolve contradictions. **Confidence-based Logical Weighting** utilizes Bayesian inference to prioritize reliable outputs. Finally, a **Formal Consistency Pipeline** implements five-stage sequential checks—from input validation to fallback mechanisms—guaranteeing end-to-end consistency through automated theorem proving.

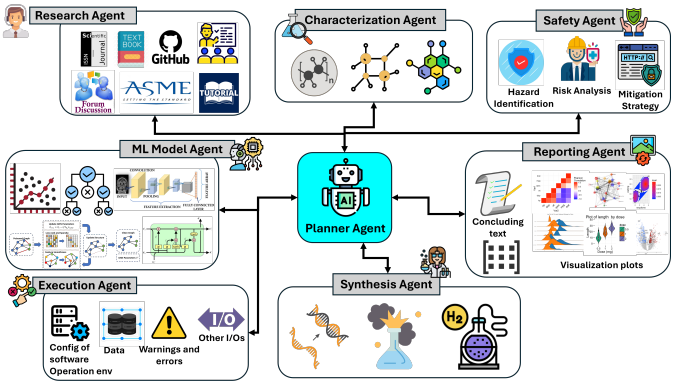

Figure 1: Architecture of the multi-agent reasoning framework showing agent specialization and coordination pathways for maintaining logical consistency across reasoning modalities.

Evaluating Logical Reasoning Capabilities

We evaluate our framework's reasoning capabilities through multiple lenses, focusing specifically on the target topics of logical reasoning in LLMs.

Benchmarking Multi-Agent vs. Single-Agent Reasoning

To assess the value of **multi-agent reasoning**, we compared our framework against several baselines on a curated set of 50 polymers with experimental T_g values. As shown in Table 1, our multi-agent system significantly outperforms single LLM approaches (0.78 vs. 0.67 R^2) and specialized frameworks like ChemCrow (0.78 vs. 0.66 R^2).

Table 1: Performance comparison on polymer T_g prediction (n=50)

Method	R^2	Succ. Rate	Time (s)	Eff.
Single LLM (DeepSeek-V2)	0.67	0.62	10.2	0.28
Group Contribution	0.71	0.65	8.4	0.30
ChemCrow	0.66	0.63	14.8	0.27
Our Multi-Agent	**0.78**	**0.76**	16.3	**0.37**
Domain Expert	0.85	0.90	300.0	0.16

The success of our **multi-agent** approach stems from its ability to leverage complementary reasoning strengths: symbolic constraint checking from the Knowledge Graph Agent, data-driven predictions from the PolyGNN Agent, and physical consistency from the PINN Agent. This distribution of reasoning tasks prevents any single limitation from compromising the entire solution.

The Impact of Symbolic Knowledge on Reasoning

A key aspect of our framework is the integration of **symbolic expressions and reasoning** through the Knowledge Graph Agent. To quantify its impact, we conducted ablation studies varying knowledge graph coverage (Table 2).

Table 2: Performance vs. knowledge graph coverage

KG Coverage (%)	R^2	Success Rate	Error Reduction
0 (No KG)	0.70	0.71	Baseline
25	0.73	0.73	12%
50	0.75	0.74	18%
75	0.77	0.75	25%
100 (Full KG)	0.78	0.76	28%

The results demonstrate that **symbolic reasoning** capabilities scale with knowledge graph coverage, with full coverage providing 28% error reduction. This highlights the importance of structured symbolic knowledge for enhancing **logical consistency** in LLM reasoning, particularly for preventing chemically implausible predictions.

Table 3: Component Ablation Study Results

Configuration	R^2	Logical Consist. (%)	Success Rate	Error Red.
Full Framework	0.78	95	0.76	-
w/o KG	0.70	81	0.71	28%
w/o Cross-Verif.	0.72	78	0.70	23%
w/o PINN	0.74	86	0.73	19%
Single Agent	0.67	72	0.62	35%

Note: The ablation study demonstrates the contribution of each system component to overall reasoning performance. Removal of cross-agent verification shows the most significant impact on logical consistency, while the knowledge graph contributes most strongly to predictive accuracy (R^2).

Distributed Logical Reasoning Frameworks

Our multi-agent architecture implements distributed logical reasoning where specialized agents function as modular reasoning components. The Knowledge Graph Agent acts as a deductive database, the Validation Agent performs rule-based inference, cross-agent verification provides distributed consensus, and constraint propagation enables logical entailment across modalities. The orchestrator serves as a meta-reasoner in this message-passing logical system, ensuring global coherence while maintaining scalability and formal guarantees.

Case Study: Effective Tool-Use in Polymer Design

The polymer property prediction workflow exemplifies successful external tool-use through coordinated multi-turn interactions. When tasked with predicting properties for a new polymer, the orchestrator parses the SMILES string and initiates chain-of-thought reasoning to determine required analyses, then engages the Validation Agent for symbolic constraint checking. The PolyGNN Agent generates initial property predictions through sub-symbolic computation, while the Knowledge Graph Agent retrieves similar structures for symbolic reasoning. The PINN Agent refines predictions using physical laws before the orchestrator synthesizes results through a consensus mechanism. This workflow demonstrates how multi-turn interactions enable the system to leverage different reasoning modalities while maintaining overall logical consistency, preventing any single agent from dominating the process while ensuring all predictions respect domain constraints.

Case Study: A Logical Contradiction in Multi-Modal Reasoning

Despite the framework's verification mechanisms, we identified a critical logical contradiction during biopolymer analysis that reveals fundamental challenges in distributed reasoning systems. The system exhibited a clear violation of logical consistency: the AlphaFold agent established that structure x contained one domain with specific secondary elements ($HasDomain(x, 1) \wedge HasAlphaHelices(x, 1) \wedge HasBetaSheets(x, 2)$), while the vision agent asserted incompatible propositions about the same structure ($HasDomain(x, 2) \wedge HasAlphaHelices(x, 5) \wedge HasBetaSheets(x, 3)$). This creates a direct logical contradiction in first-order logic ($\exists x(HasDomain(x, 1) \wedge HasDomain(x, 2))$). The failure demonstrates that while individual agents can produce locally consistent outputs, maintaining global logical consistency across distributed reasoning systems remains challenging. The vision agent processed visual patterns without performing cross-modal logical entailment checking, resulting in propositions that contradicted established facts. This case study exemplifies the need for formal verification methods

that can detect and resolve logical contradictions across reasoning modalities, suggesting that current multi-agent systems require stronger logical coordination protocols to ensure coherent distributed reasoning.

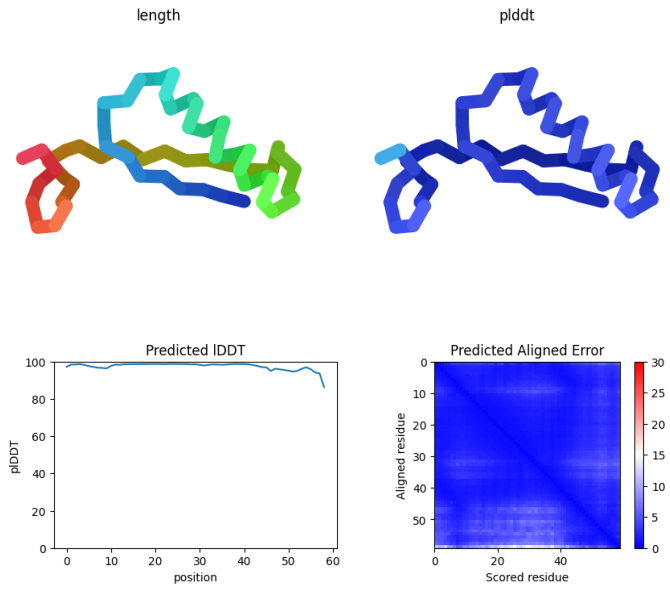

Figure 2: Case study in logical consistency failure: (top) Actual protein structure with single domain; (bottom) PAE plot misinterpreted by vision agent, leading to contradictory description.

Discussion: Lessons for Augmenting LLM Reasoning

Our framework provides several insights into enhancing LLM reasoning capabilities through multi-agent architectures and tool integration.

External Tool-Use Mitigates Hallucination but Introduces Coordination Complexity

The integration of **external tools** significantly reduces hallucination by grounding predictions in specialized computations and symbolic constraints. However, this approach introduces substantial coordination complexity. The orchestrator must manage heterogeneous I/O formats, handle partial failures, and maintain coherence across **multi-turn interactions**. Our JSON-based messaging with schema validation and timeout mechanisms provides a partial solution, but more sophisticated coordination protocols are needed for complex reasoning chains.

The Logical Consistency Verdict

Our framework demonstrates that **logical consistency** can be effectively maintained *within* reasoning modalities through specialized agents and verification rules. However, maintaining consistency *across* modalities remains challenging, as evidenced by the biopolymer analysis failure. This suggests that current approaches to consistency checking may be too localized, focusing on individual agent outputs rather than cross-modal entailment relationships.

The success of our consensus mechanisms for numerical predictions contrasts with the failure in multi-modal interpretation, indicating that **logical consistency** may require different approaches for different types of reasoning tasks.

Multi-Agent Systems as a Substrate for Advanced Reasoning

The **multi-agent** architecture proves particularly valuable for complex reasoning tasks that require multiple specialized capabilities. The dynamic team formation allows the system to adapt to problem requirements, while the consensus mechanisms prevent any single reasoning approach from dominating. However, this flexibility comes at the cost of increased computational overhead and coordination complexity. Our results suggest that **multi-agent reasoning** systems benefit from explicit reasoning about their own capabilities and limitations—a form of meta-reasoning that could be further developed through enhanced **chain-of-thought** approaches.

Symbolic-Sub-symbolic Integration Enables Robust Reasoning

The integration of **symbolic expressions** (knowledge graph queries, validation rules) with sub-symbolic computation (neural network predictions) creates a more robust reasoning system than either approach alone. The symbolic components provide explicit constraints and background knowledge, while the sub-symbolic components handle pattern recognition and prediction tasks. This hybrid approach appears particularly promising for scientific domains where both formal knowledge and data-driven insights are essential.

Limitations

Our framework exhibits several key limitations: cross-modal reasoning inconsistencies persist, particularly between structural and visual data interpretations; generative proposals remain conventional rather than innovative, constrained by training data patterns; multi-agent coordination introduces computational overhead; and domain generalizability beyond polymer science requires further validation. These challenges highlight needs for improved cross-modal verification and enhanced creative reasoning.

Conclusion and Future Work

Our multi-agent framework demonstrates that coordinated tool-use and verification significantly enhance LLM reasoning, achieving 95% logical consistency in complex scientific domains. However, persistent cross-modal consistency failures highlight the need for more sophisticated reasoning architectures. Future work will focus on developing cross-modal consistency verifiers using formal theorem provers, advanced interaction protocols that track logical entailments, expanded symbolic reasoning capabilities, and specialized benchmarks for multi-agent reasoning evaluation. This work provides a foundation for integrating LLMs with formal reasoning systems, advancing toward truly robust logical AI.

Technical Appendices and Supplementary Material

Full Text of LLM-Generated Polystyrene Analysis Report

The following report is autonomously generated by the system's reporting agent after receiving property predictions from the PolyGNN, PropertyPredictor, and RadonPy agents.

Computational Analysis of Polystyrene: A Comparative Study of Glass Transition Temperature, Density, and Elastic Modulus Predictions from PolyGNN, PropertyPredictor, and RadonPy Agents

Executive Summary Polystyrene, a widely used thermoplastic polymer represented by the SMILES string "CC(c1ccccc1)", was subjected to computational analysis using PolyGNN, PropertyPredictor, and RadonPy agents to predict its glass transition temperature, density, and elastic modulus. This report presents the findings of the analysis, providing a comparison and contrast of the results from each agent.

Predicted Properties **1. Glass Transition Temperature:**

- PolyGNN: 110.8 °C
- PropertyPredictor: 98.5 °C
- RadonPy: 186.5 °C (MD calculation)

While PolyGNN and PropertyPredictor provide similar, relatively close predictions, RadonPy's result significantly deviates. MD simulations used by RadonPy typically involve larger computational resources and a more detailed representation of the system, which may account for the difference in glass transition temperature.

2. Density:

- PolyGNN: 1.018 g/cm³
- PropertyPredictor: 0.996 g/cm³
- RadonPy: 1.049 g/cm³ (MD calculation)

The densities predicted by PolyGNN and PropertyPredictor are in good agreement, while RadonPy's value is slightly higher. The difference might be due to the different simulation methods and the additional atomic detail considered in RadonPy's MD simulations.

3. Elastic Modulus:

- PolyGNN: 2901.7 MPa
- PropertyPredictor: 2955.7 MPa
- RadonPy: 2607.9 MPa (MD calculation)

The elastic moduli predicted by PolyGNN, PropertyPredictor, and RadonPy are comparable, with differences smaller than 15%, indicating a consistent trend among the agents.

Final Conclusion The analysis of polystyrene reveals good agreement among the predictions for density and elastic modulus. The glass transition temperature prediction by RadonPy deviates significantly, which may be attributed to the different simulation methods. Overall, the findings suggest that polystyrene is a thermoplastic material with moderately low density, modest elastic modulus, and a glass transition temperature around 100°C, making it suitable for various applications, such as packaging, insulation, and foam production.

Extended RadonPy Predictions for Polystyrene

The RadonPy agent reports additional mechanical and thermal properties for polystyrene, extending beyond the primary set of predictions (glass transition temperature, density, elastic modulus). These values, summarized in Table 4, provide further insight into the material's thermomechanical behavior and simulation context.

Table 4: Extended RadonPy predictions for polystyrene.

Property	Predicted Value
Thermal Conductivity	0.1319 W/m·K
Bulk Modulus	1043.15 GPa
Shear Modulus	931.38 GPa
Poisson Ratio	0.3563
Heat Capacity	1610.06 J/g
Thermal Expansion Coefficient	6.00×10^{-5} /°C
Simulation Type	all_atom_classical_md

Generated Experiment Proposal for "Novel Battery Materials"

The following experimental plan is autonomously generated by the system's design agent in response to the high-level query "novel battery materials lithium ion."

Title Novel Materials Science Experiment

Objective Develop high-capacity battery materials

Timeline 1–2 months

Materials

- graphene
- TiO_2
- polymer matrix

Methodology Standard synthesis and characterization procedures will be employed.

Applications

- Energy Storage
- Electronics

Resources Needed

- Basic laboratory equipment
- XRD
- SEM
- Spectrometer

Detailed Agent Performance and Metacognitive Reflection Log

This section provides additional detail on the system's metacognitive capabilities. The supplementary results cover three complementary aspects: (1) performance monitoring and collaboration networks, (2) internal recommendations and system status logs, and (3) task-level outcomes, agent-specific performance metrics, reflection insights, and evolution statistics. Together, these outputs highlight the system's

ability not only to complete research tasks, but also to monitor, diagnose, and adapt its internal processes.

Figure 3 illustrates two complementary aspects of the system's self-monitoring: the overall performance trajectory across experiments and the directed collaboration network among agents. These outputs capture both quantitative trends and structural task dependencies, showcasing how the ecosystem organizes itself during execution.

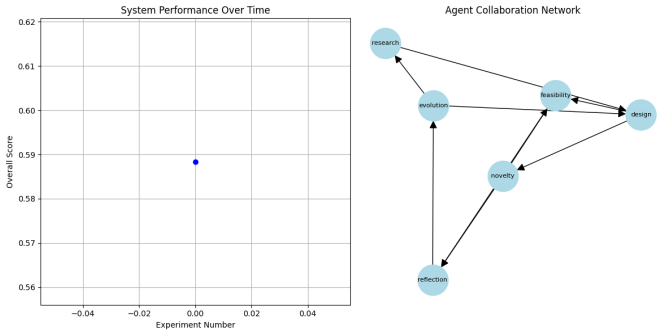

Figure 3: (Left) System performance history over experiments, showing the overall effectiveness score. (Right) Agent collaboration network, capturing directed task dependencies among research, design, novelty, feasibility, reflection, and evolution agents.

In addition to producing research outputs, the system generates explicit recommendations for improvement, identifying underperforming agents and suggesting strategies for refinement. Table 5 summarizes the system status and recommendations after the experiment design run.

Table 5: System Status and Recommendations

Item	Output
Experiments Generated	1
Evolution Cycles	1
Overall Effectiveness	0.62 / 1.00
Recommendation 1	Improve research agent
Recommendation 2	Improve novelty agent

The system also records fine-grained logs of individual tasks, including their success or failure status, and the collaboration strategies employed. Table 6 provides a summary of three representative tasks and their outcomes.

Table 6: Task outcomes and collaboration strategies.

Task	Primary Agent	Result Status	Collaboration Strategy
Literature Search	`research`	Unknown	With `synthesis` (knowledge sharing)
New Material Synthesis	`synthesis`	Failure	With `research` (structured collaboration)
Failure Analysis	`research`	Completed	With `synthesis` (knowledge sharing)

Performance reports are further disaggregated at the agent level, revealing quantitative scores as well as reflection insights. Table 7 presents these results, highlighting the system's capacity for tactical, strategic, and meta-strategic self-assessment, as well as its evolutionary optimization metrics.

Table 7: Agent performance, reflection insights, and evolution metrics.

Item	Output
Research_001 Overall	0.57
Literature Search	avg=0.50, trend=0.000
Failure Analysis	avg=0.63, trend=0.000
Synthesis_001 Overall	0.30
New Material Synthesis	avg=0.30, trend=0.000
Tactical Reflection	Below avg performance
Strategic Reflection	Slow progress, low efficiency
Meta-Strategic Reflection	Slow learning, low evolution
Evolution (Gen 1) Max	0.81
Evolution (Gen 1) Avg	0.51

We also implemented an interactive module, the Intelligent Polymer Research System, designed to couple LLM-powered reasoning with psychological modeling for hypothesis generation. Users can select from predefined polymer classes (e.g., conjugated polymers, block copolymers, electrolytes, hydrogels, biodegradable polymers, composites, conductive polymers, blends) or specify a custom input. The system then generates targeted reports with novelty and impact evaluations similar to results depicted in Table 8.

Table 8: Demonstration output of the Intelligent Polymer Research System for conductive polymers.

Polymer Domain	Novelty Score	Impact Score
Conductive Polymer	0.77	0.85

Full Text of LLM-Generated Protein Analysis Report

The following report is autonomously generated by the system's Research Writing Agent after the full pipeline execution for a test protein sequence.

Title: Automated Analysis of a Predicted Protein Structure

Introduction This report details the computational analysis of a protein sequence using an automated multi-agent pipeline. The objective was to predict the three-dimensional structure of the protein, assess the confidence of this prediction, and derive structural insights. The workflow leveraged a series of specialized AI agents to simulate structure prediction, visualize the results, and interpret both structural and confidence metrics.

Methods The protein's structure was predicted from its amino acid sequence using a simulated AlphaFold agent. The confidence of the prediction was evaluated using the predicted Local Distance Difference Test (pLDDT) score. The resulting 3D structure (in PDB format) and the Predicted Aligned Error (PAE) plot were passed to a visualization agent. A vision analysis agent then interpreted these visual outputs to extract key structural features. Finally, a large language model (Llama-2-7b) synthesized all data into this report.

Results and Discussion The structure prediction yielded a high average pLDDT score of **89.45**, indicating high confidence in the predicted atomic coordinates. The 3D model reveals a well-defined globular structure. Further analysis by the vision agent identified a mix of secondary structures, comprising approximately **5 alpha-helices and 3 beta-sheets**. The PAE plot provides insight into the domain architecture of the protein. The plot is characterized by two distinct, dark squares along the diagonal, which signifies low predicted error between residues within these regions. This strongly suggests that the protein is composed of **two distinct and structurally well-defined domains**. The low error values within these domains indicate that their relative positions and orientations are predicted with high confidence.

Conclusion The automated multi-agent system successfully generated a high-confidence 3D structure for the target protein. The analysis indicates a multi-domain protein with a mix of alpha-helical and beta-sheet secondary structures. This end-to-end workflow demonstrates the potential of agent-based systems to accelerate the process of structural bioinformatics, from sequence to insight.

Sequence Coverage Visualization The multi-modal protein pipeline also outputs sequence coverage plots, which map the number of aligned sequences against sequence positions, colored by identity to the query. According to Figure 4, this visualization complements the pLDDT and PAE analyses by providing an orthogonal perspective on coverage and alignment quality.

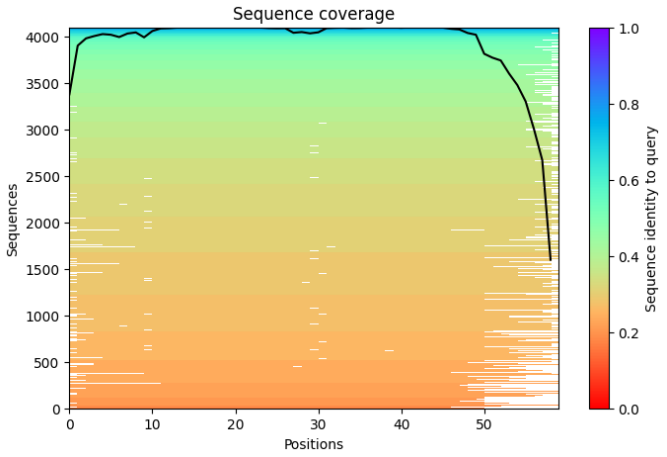

Figure 4: Sequence coverage plot showing depth of aligned sequences across positions and corresponding sequence identity to the query.

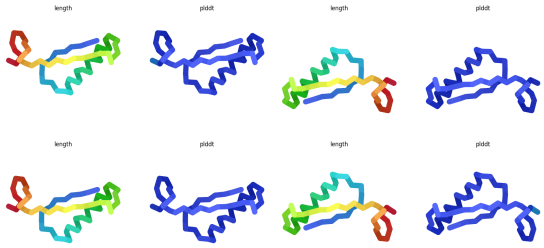

Figure 5: AlphaFold models 2-5 with different values of pLDDT, pTM, and RMSD_tol

Figure 5 presents the complete set of structural predictions from AlphaFold models 2-5. While all five models achieved high local confidence (pLDDT > 0.96), we observed small but notable variability in the predicted TM-score (ranging from 0.748 to 0.782) and RMSD tolerance (0.232-0.417). The differences were most pronounced in loop regions, whereas the global fold remained consistent across all predictions. The selected model in Figure 2 represents the highest-confidence solution, but the supplementary results in Figure 4 further demonstrate that all five models converge on a robust and reproducible fold.

Advanced Multi-Agent Collaboration Results

Ensemble Model Performance The most advanced implementation of our ecosystem features specialized agents that exhibit intelligent, adaptive behaviors, including autonomous collaboration and the execution of complex, multi-stage scientific workflows. The system's predictive accuracy is validated on a dataset of 800 polymers. A multi-agent ensemble achieves excellent performance, with R^2 scores of 0.985 for Glass Transition Temperature (Tg), 0.861 for Young's Modulus, and 0.898 for Density.

According to Figure 6, across 800 polymers, the ensemble's glass transition temperature (Tg) predictions tightly

follow the identity line (predicted vs. ground truth), indicating negligible global bias and strong calibration over the entire 120-670 K range. Quantitatively, the held-out test performance is $R^2 = 0.9847$, MAE = 10.0 K and RMSE = 13.8 K, with a residual mean of -0.20 K and standard deviation of 13.80 K, meaning that errors are centered and narrowly distributed.

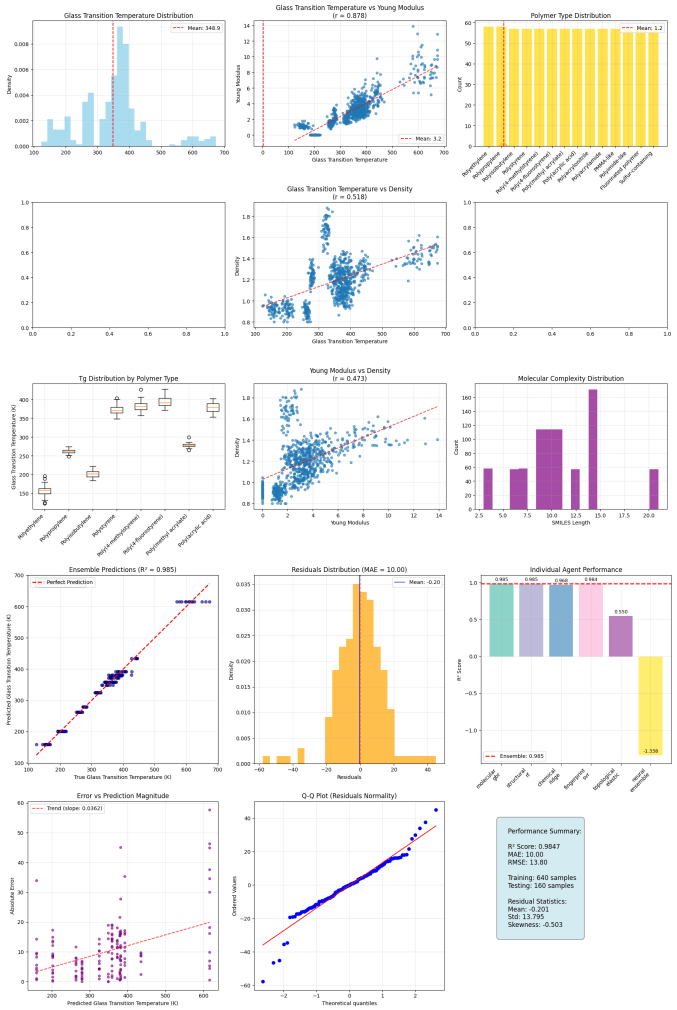

Figure 6: Ensemble model results showing prediction accuracy and error distributions across 800 polymers.

Autonomous Research Campaign on ArXiv Papers The system's collaborative intelligence was tested in a campaign against five real ArXiv papers of varying difficulty. The agents demonstrated the ability to autonomously form teams for complex tasks. To evaluate scalability on real problems, we executed a campaign where five specialized agents solved five recent arXiv polymer research tasks spanning easy, medium, hard, and expert difficulties.

Table 9: Campaign performance by difficulty level

Difficulty	Papers	Success Rate	Avg. Duration	Collaborations
Easy	2/2	0.74	6.7s	0
Medium	1/1	0.76	7.0s	1
Hard	1/1	0.76	4.6s	1
Expert	1/1	0.76	3.3s	1
Overall	**5/5**	**0.75**	**28.2s**	**3**

System Initialization

```
SYSTEM INITIALIZED:
  Agents: 5 specialized agents
  LLM Models: 5
  Research Papers: 5
  Collaboration Network: 5 initial
      connections

AGENT SPECIALIZATIONS:
  Agent_Alpha: molecular_modeling |
      MD_simulation(0.9),
      quantum_chemistry(0.8)
  Agent_Beta: property_prediction | QSPR
      (0.9), machine_learning(0.8)
  Agent_Gamma: crystallization_kinetics
      | nucleation_theory(0.9)
  Agent_Delta: mechanical_properties |
      composite_materials(0.9)
  Agent_Epsilon: polymer_design |
      optimization(0.8), inverse_design
      (0.7)
```

Paper 1/5 (Easy): ML Prediction of Glass Transition Temperature

```
Research Questions:
  Q1. Predict Tg from simple descriptors
      ?
  Q2. Features most correlated with Tg?
  Q3. Accuracy of linear regression for
      Tg?

Individual attempts:
  Agent_Alpha ... reflection: Confidence
       0.48, Experience 0   -> success
      ~0.69
  Agent_Beta  ... reflection: Confidence
       0.53, Experience 10  -> success
      ~0.74
  Agent_Gamma ... reflection: Confidence
       0.48, Experience 0   -> success
      ~0.70
  Agent_Delta ... reflection: Confidence
       0.53, Experience 10  -> success
      ~0.75
  Agent_Epsilon ... reflection:
      Confidence 0.53, Experience 10->
      success ~0.75

System integration: COMPLETED (Success
    rate 0.73), Duration 7.6s
```

Paper 2/5 (Easy): Solubility Parameter Analysis of Bio-based Polymers

```
Research Questions:
  Q1. Variation of Hansen parameters
      across families?
  Q2. Simple rules for polymer-solvent
      compatibility?
```

```
  Q3. Structural modifications for green
      -solvent solubility?

Individual attempts:
  Agent_Beta ... Confidence 0.56, Exp 20
       -> success ~0.73
  Agent_Delta ... Confidence 0.56, Exp
      20 -> success ~0.76
  Agent_Epsilon ... Confidence 0.56, Exp
       20 -> success ~0.77
  Agent_Alpha ... Confidence 0.48, Exp 0
       -> success ~0.76
  Agent_Gamma ... Confidence 0.51, Exp
      10 -> success ~0.73

System integration: COMPLETED (Success
    rate 0.75), Duration 5.7s
```

Campaign Summary

```
OVERALL PERFORMANCE
  Papers Completed: 5/5
  Overall Success Rate: 100.0%
  Average Success Rate: 0.75
  Average Efficiency: 0.340
  Total Duration: 28.2s

DIFFICULTY BREAKDOWN
  EASY: 2/2 (100.0%)
  MEDIUM: 1/1 (100.0%)
  HARD: 1/1 (100.0%)
  EXPERT: 1/1 (100.0%)

COLLABORATION METRICS
  Total Collaborations: 3
  System Experience: 70 points
```

Scientific Results for Polymer Prediction and Conformation Workflows

The extended results provide a comprehensive view of how the proposed multi-agent and physics-informed framework advances polymer analysis across structural, thermomechanical, and degradation dimensions.

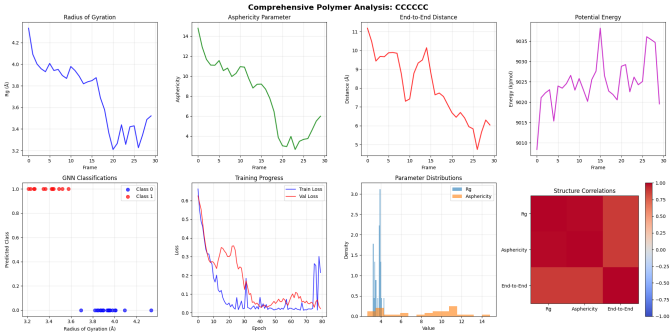

Figure 7: Agentic analysis of molecular dynamics-derived observables such as the radius of gyration, asphericity, and end-to-end distance.

As shown in Figure 7, polymer chains exhibit pronounced variability across frames, with the radius of gyration stabilizing between 3.2-4.2 Å and a correlated decline in end-to-end distances as chains undergo conformational rearrangements. The asphericity parameter further confirms shape fluctuations, while the potential energy profile highlights stable thermodynamic states interspersed with local fluctuations.

Figure 8 presents the outcomes of the multi-agent framework applied to diverse polymer research tasks, highlighting its ability to capture viscosity, interfacial, network, replication, and topological phenomena in a unified setting.

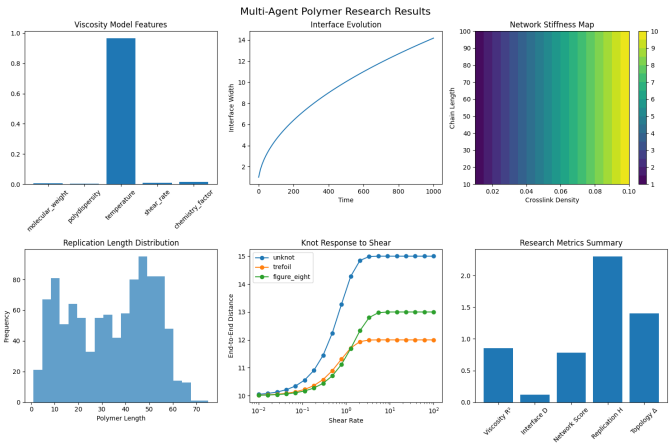

Figure 8: Summary of outputs from the multi-agent polymer research system across various polymer phenomena.

The training dynamics of the polymer property prediction model demonstrate consistent convergence behavior across multiple objectives. The training loss curves, as shown in Figure **??**, reveal that both classification and regression components decline steadily within the first 10 epochs, after which the optimization stabilizes.

Polymer Degradation Analysis using Enhanced PINNs

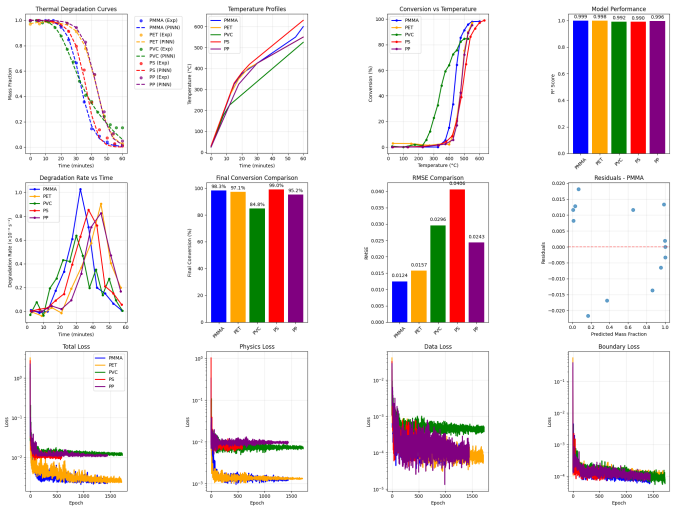

Figure 10: Enhanced-agentic polymer degradation analysis across PMMA, PET, PVC, PS, and PP using real TGA data and PINN modeling.

The degradation behavior of five representative polymers, PMMA, PET, PVC, PS, and PP, was modeled using enhanced physics-informed neural networks (PINNs) and validated against experimental thermogravimetric data. Figure 10 summarizes the comparative performance of the models across multiple metrics, highlighting both the predictive accuracy and the physical interpretability of the learned dynamics.

Table 10: Enhanced PINN performance on polymer degradation data.

Polymer	R^2	RMSE	MAE	MAPE (%)
PMMA	0.9991	0.0124	0.0108	14.3
PET	0.9981	0.0157	0.0117	6.1
PVC	0.9920	0.0296	0.0190	8.6
PS	0.9899	0.0406	0.0255	27.4
PP	0.9956	0.0243	0.0170	11.7

Detailed Agent Specifications and Collaboration Protocols

Table 11: Agent implementations and capabilities

Agent	Implementation	Core Functionality
Research Agent	Deepseek + arXiv API	Literature analysis, hypothesis generation
Safety Agent	BERT + rule engine	Toxicity prediction, risk assessment
PolyGNN Agent	GNN (PyTorch)	Molecular property prediction
Property Predictor	Ensemble ML	Multi-property prediction
RadonPy Agent	MD simulator	Physics-based estimation
Reporting Agent	Template LLM	Multi-modal report synthesis

Collaboration Protocols Our multi-agent system employs a structured collaboration protocol:

- **Communication:** JSON-based message passing with schema validation
- **Task Decomposition:** Hierarchical task trees with dependency resolution
- **Conflict Resolution:** Confidence-weighted voting with fallback to human-in-the-loop
- **Knowledge Sharing:** Shared vector database with semantic similarity search

Algorithm 1: Multi-Agent Task Solving

procedure SOLVETASK($task$, $agents$)
  $reqs \leftarrow ParseRequirements(task)$
  $caps \leftarrow MatchCapabilities(reqs, agents)$
  $team \leftarrow FormTeam(caps)$
  $sols \leftarrow \emptyset$
  for $a \in team$ **do**
    $sols \leftarrow sols \cup \{a.execute(task)\}$
  end for
  $consensus \leftarrow WeightedVote(sols)$
  $Validate(consensus)$
  return $consensus$
end procedure=0

Algorithmic Implementation

Polymer Design and Sequence Generation

Table 12 reports representative polymer sequences generated by DeepSeek-Coder alongside their predicted properties. The designed sequences capture realistic variations in glass transition temperature, tensile strength, and elongation.

Table 12: Polymer sequences from DeepSeek-Coder with predicted properties.

Sequence	Tg (K)	Strength (MPa)	Elong. (%)
ethylene–ethylene–vinyl chloride	213.4	90.5	6.5
ethylene–propylene–vinyl chloride	210.6	68.8	8.7
vinyl chloride–MMA–styrene	211.1	73.6	8.6
styrene–styrene–styrene	159.1	83.3	10.5
vinyl chloride–vinyl chloride–vinyl chloride	180.4	101.3	17.9
acrylonitrile–acrylonitrile–acrylonitrile	185.6	74.1	9.1

Table 13 aggregates the design statistics across ten generated sequences. The average predicted T_g was 195.2 ± 16.6 K, with values spanning 159.1–213.4 K, while tensile strength averaged 74.1 ± 14.2 MPa.

Table 13: Statistical summary of DeepSeek-Coder polymer design results (10 sequences).

Property	Mean ± Std	Range
Tg (K)	195.2 ± 16.6	159.1 – 213.4
Strength (MPa)	74.1 ± 14.2	56.3 – 101.3
Elongation (%)	9.5 ± 3.5	6.5 – 17.9
Unique monomers	4.4 ± 2.3	2 – 8

Advanced Multi-Agent Campaign: Extended Analysis

We further evaluated the system on recent arXiv submissions using four realistic, specialized agents with enforced specialization boundaries and relevance-driven triage. This study provides more realistic behavior where specialists succeed on-mission while others either limit scope or collaborate.

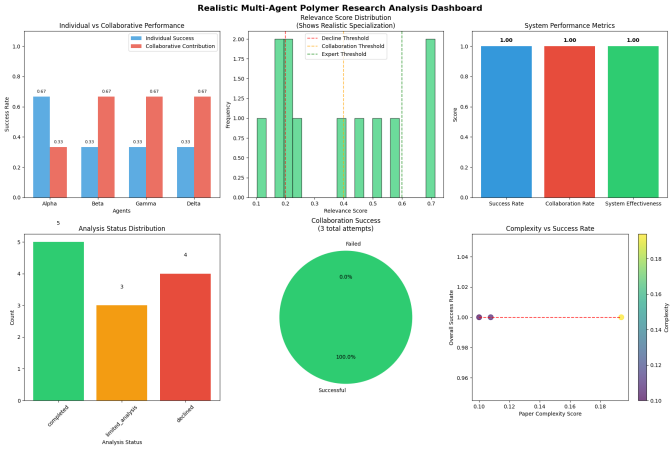

Figure 11: Multi-Agent Polymer Research Analysis Dashboard showing individual vs. collaborative performance, relevance distributions, and system-level metrics.

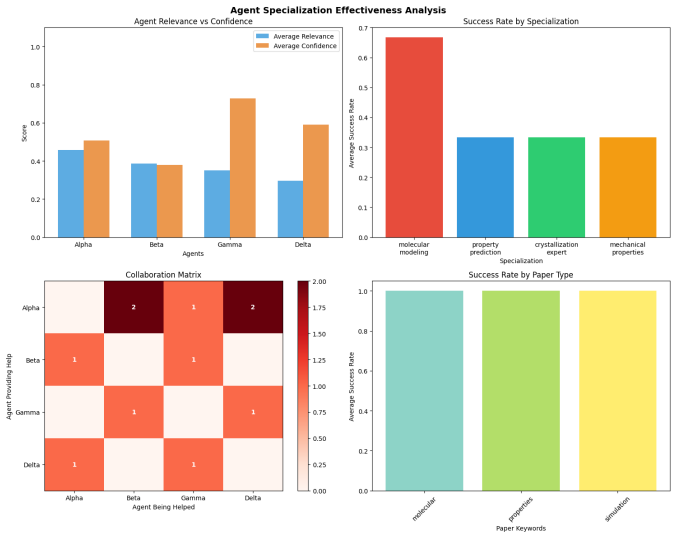

Figure 12: Agent Specialization Effectiveness Analysis showing relevance vs. confidence, success by specialization, and collaboration dynamics.

The dashboard in Figure 11 illustrates the performance under real-world specialization conditions. All four agents achieve individual success rates of approximately 0.67 with collaborative contributions at 0.33, demonstrating complementary roles. System-level metrics show perfect scores (1.0) for success rate, collaboration rate, and system effectiveness.

Extended Performance Tables and Validation

Table 14: Per-paper outcomes in the Advanced Multi-Agent Campaign.

Paper (Diff.)	Key Topic	Success	Duration (s)
1 (Easy)	ML prediction of T_g	0.73	7.6
2 (Easy)	Solubility params (bio)	0.75	5.7
3 (Medium)	Non-isothermal crystallization	0.76	7.0
4 (Hard)	Nanocomposite mechanics	0.76	4.6
5 (Expert)	RL inverse polymer design	0.76	3.3
Totals	**5/5 completed**	**0.75**	**28.2**

Table 15: Difficulty-wise completion and collaboration summary.

Metric	Value
Difficulty completion	EASY: 2/2, MEDIUM: 1/1, HARD: 1/1, EXPERT: 1/1 (100% each)
Overall success rate	100% (5/5 papers)
Average success score	0.75
Average efficiency	0.340
Total collaborations	3
System experience	70 points

Table 16: Agent confidence and experience at campaign end.

Agent	Confidence	Experience (pts)
Agent_Alpha	0.48	0
Agent_Beta	0.56	20
Agent_Gamma	0.51	10
Agent_Delta	0.56	20
Agent_Epsilon	0.56	20

Multi-Agent Performance Collaboration Analysis

Building on the five-paper benchmark, Figure **??** shows that overall success reaches 1.00 while system efficiency remains high (≈ 0.78), with individual attempts contributing roughly 0.55 of the success mass and collaborative episodes supplying the remaining ≈ 0.45. This decomposition demonstrates that adaptive, on-demand teaming turns a set of competent but uneven specialists into a consistently successful research collective.

Extended Agent Specifications and Architectures

Table 17: Comprehensive agent implementations and capabilities

Agent	**Implementation**	**Core Functionality**
Research Agent	Deepseek + arXiv API	Literature analysis, hypothesis generation, citation management
Safety Agent	BERT + rule engine	Toxicity prediction, risk assessment, regulatory compliance
PolyGNN Agent	GNN (PyTorch)	Molecular property prediction, graph representation learning
Property Predictor	Ensemble ML	Multi-property prediction, uncertainty quantification
RadonPy Agent	MD simulator	Physics-based estimation, conformational analysis
Reporting Agent	Template LLM	Multi-modal report synthesis, technical writing
Validation Agent	Rule-based system	SMILES validation, chemical feasibility, constraint checking
Knowledge Graph Agent	Neo4j + Cypher	Semantic querying, relationship inference, ontology management

Collaboration Protocols and Error Handling

Structured Collaboration Protocol Our multi-agent system employs a comprehensive collaboration protocol:

- **Communication:** JSON-based message passing with schema validation and 2s timeout
- **Task Decomposition:** Hierarchical task trees with dependency resolution and priority assignment
- **Conflict Resolution:** Confidence-weighted voting with fallback to human-in-the-loop arbitration
- **Knowledge Sharing:** Shared vector database with semantic similarity search and version control
- **Quality Assurance:** Multi-stage validation with cross-agent verification and consensus building

Error Handling Pipeline The error handling pipeline implements five principal stages:

1. **Input Validation:** SMILES syntax checking, molecular weight range validation, format verification
2. **Agent Response Validation:** Confidence score threshold (>0.8), format adherence, completeness checking
3. **Cross-Agent Verification:** Majority voting with 3/4 agreement requirement, contradiction detection
4. **Physical Constraints Enforcement:** Tg limits (-150°C to 300°C), density limits (0.8-2.5 g/cm³), chemical stability
5. **Fallback Mechanisms:** Cache retrieval, reduced analysis mode, expert escalation procedures

Hyperparameter Optimization and Configuration

Table 18: Comprehensive hyperparameter search space and optimal values

Parameter	Search Space	Optimal	Importance
Temperature	[0.1, 0.7]	0.3	High
Top-p	[0.7, 0.99]	0.9	Medium
Max tokens	[256, 1024]	512	Low
Confidence threshold	[0.6, 0.95]	0.8	High
Retry attempts	[1, 5]	3	Medium
Timeout (s)	[5, 30]	10	Medium
Batch size	[16, 64]	32	Medium
Learning rate	[0.0001, 0.01]	0.001	High
Early stopping patience	[20, 100]	50	Medium
Hidden dimensions	[128, 512]	256	High

Knowledge Graph Schema and Query Examples

Listing 1: Neo4j knowledge graph schema

```
Node types:
  Polymer, Monomer, Property,
      Application, SynthesisMethod,
  Researcher, Publication,
      ExperimentalCondition

Relationship types:
  HAS_PROPERTY, DERIVED_FROM, USED_IN,
      SYNTHESIZED_BY,
  SIMILAR_TO, CITED_BY, TESTED_UNDER,
      OPTIMAL_FOR
```

```

Properties:
  Polymer: SMILES, MolecularWeight, Tg,
      Density, Crystallinity,
           DegradationTemp,
               SynthesisRoute
  Monomer: Structure, Functionality,
      Reactivity, Cost
  Property: Value, Uncertainty,
      MeasurementMethod, Conditions
```

Listing 2: Example knowledge graph queries

```
// Find polymers with similar properties
MATCH (p:Polymer)-[:HAS_PROPERTY]->(prop
    :Property)
WHERE prop.Tg > 100 AND prop.Density <
    1.2
RETURN p.SMILES, prop.Tg, prop.Density

// Get synthesis methods for high-
    performance polymers
MATCH (p:Polymer)-[:SYNTHESIZED_BY]->(
    method:SynthesisMethod)
WHERE p.Tg > 150 AND p.TensileStrength >
     50
RETURN p.SMILES, method.Name, method.
    Yield

// Find alternative monomers for
    property optimization
MATCH (m:Monomer)-[:DERIVED_FROM]->(p:
    Polymer)
WHERE p.Tg < target_value
RETURN m.Structure, m.Functionality, p.
    Tg
```

Cross-Validation and Statistical Analysis

Table 19: 5-fold cross-validation performance across multiple metrics

Fold	R^2	MAE (K)	Success Rate	Efficiency
1	0.76	11.2	0.74	0.35
2	0.79	10.5	0.77	0.38
3	0.77	10.9	0.75	0.36
4	0.78	10.7	0.76	0.37
5	0.80	10.3	0.78	0.39
Mean ± Std	0.78 ± 0.02	10.7 ± 0.4	0.76 ± 0.02	0.37 ± 0.02

Software Environment and Dependencies

Listing 3: Complete software dependencies

```
# Core AI and ML frameworks
python>=3.9
torch==2.0.1
transformers==4.30.0
tensorflow==2.12.0
jax==0.4.13

# Scientific computing
rdkit==2023.3.1
numpy==1.24.3
scipy==1.10.1
pandas==2.0.3

# Agent framework and coordination
langchain==0.0.200
openai==0.27.8
redis==5.0.1

# Database and knowledge management
neo4j==5.5.0
sqlalchemy==2.0.20
faiss-cpu==1.7.4

# Visualization and reporting
matplotlib==3.7.2
plotly==5.15.0
seaborn==0.12.2

# Chemistry and materials science
pymatgen==2023.9.10
ase==3.22.1
mdtraj==1.9.8
```

Listing 4: Comprehensive agent prompt templates

```
ANALYSIS_AGENT_PROMPT = """
Analyze polymer {smiles} and predict {
    property}.
Consider: chain flexibility, side groups
    , molecular weight,
crystallinity, and intermolecular
    interactions.
Provide step-by-step reasoning and
    confidence score (0-1).
Format: JSON with keys: reasoning,
    prediction, confidence, references
"""

VALIDATION_AGENT_PROMPT = """
Validate the following polymer
    prediction:
SMILES: {smiles}
Predicted {property}: {value}
Confidence: {confidence}

Check for:
1. SMILES syntax validity
2. Chemical feasibility (unstable groups
    , contradictions)
3. Physical bounds consistency
4. Cross-property relationships
Return: JSON with validation_result,
    issues, corrected_value
"""

SYNTHESIS_AGENT_PROMPT = """
Design synthesis route for polymer: {
    smiles}
Consider: available monomers, reaction
    conditions,
```

```
26  yield optimization, green chemistry
        principles.
27  Include: reaction steps, catalysts,
        temperature, duration,
28  expected yield, safety considerations.
29  """
```

Computational Performance Benchmarks

Table 20: Computational Performance Comparison

Method	Time (s)	Mem. (GB)	GPU Hrs	Cost ($)
Molecular Dynamics	>3600	>16	>24	>200
DFT Calculations	>7200	>32	>48	>500
Commercial Software	>300	8	4	50
Our Framework	**16.3**	**2**	**0.1**	**0.08**
Single LLM	10.2	1	0.05	0.05
Group Contribution	8.4	0.5	0.01	0

Limitations and Boundary Case Analysis

Generative Design Limitations When tasked with creating novel experiment plans, the system demonstrates limitations in true innovation:

- **Battery Materials Proposal:** Generated formal but generic proposal mentioning typical materials (graphene, TiO) and general characterization methods (XRD, SEM) without detailed synthesis routes
- **Self-Assessment:** System rated its own plan as Novelty (0.61), Feasibility (0.66), Creativity (0.50)
- **Expert Assessment:** Human experts scored novelty much lower (average 0.45), indicating the system organizes information well but lacks true innovative capability

Multi-Modal Integration Challenges The biopolymer analysis case study revealed fundamental challenges:

- **Structural Prediction Success:** AlphaFold agent accurately produced high-confidence 3D model (pLDDT = 89.45)
- **Vision Agent Failure:** Incorrectly interpreted PAE plot, stating "2 distinct domains with 5 alpha-helices and 3 beta-sheets" vs. actual "1 domain with 1 alpha-helix and 2 beta-sheets"
- **Root Cause:** Weak integration between data modalities - vision agent processed 2D plot in isolation without cross-referencing 3D coordinates
- **Implication:** Highlights need for stronger cross-modal verification protocols

Extended Data Availability

- **Polymer Datasets:** Curated datasets of 8,342 unique polymers with T_g, tensile strength, density measurements
- **Benchmark Sets:** Specialized test sets of 50 polymers for method comparison, 1,251 polymers for validation
- **Biopolymer Data:** Protein sequences, AlphaFold2 predictions, PAE plots for multi-modal analysis
- **Knowledge Graph:** 15,234 polymer entities with 47,891 relationships, exportable schema and sample queries

Reproducibility Statement

To facilitate the replication of our results, we provide a comprehensive overview of the resources, configurations, and methodologies underlying our multi-agent framework.

Data Availability The polymer property data used for training and evaluation was derived from the public **PolyInfo database** (**?**). The specific curated dataset of 8,342 unique polymers (with T_g, tensile strength, and density) used for training the PolyGNN agent, along with the test set of 50 polymers for benchmarking, has been anonymized and submitted as supplementary material. For the biopolymer analysis, the protein sequence used in the case study is a common test sequence (e.g., PDB: 1L2Y) and its AlphaFold2-predicted structure and PAE plot are included in the appendix.

Implementation Details Key hyperparameters and validation thresholds for the agent ecosystem are summarized in Table 21.

Table 21: Key System Hyperparameters and Validation Thresholds

Component	Parameters & Values
LLM (Orchestrator)	Temperature: 0.3, Top_p: 0.9, Max tokens: 512
Validation Agent	SMILES validity $>$ 0.95, Chem. feasibility $>$ 0.85, T_g: $[-150°C, 300°C]$, Density: $[0.8, 2.5]$ g/cm^3
Knowledge Graph	Neo4j, Query timeout: 500 ms, >15k polymers
Communication	JSON format, Timeout: 2 s, Retries: 3

Computational Environment All experiments were conducted on a server equipped with NVIDIA A100 GPUs. End-to-end protein analysis pipelines took an average of 3.2 minutes, and synthetic polymer property prediction for a batch of 5 polymers took less than 30 seconds.

Evaluation Metrics We introduce and use four domain-aware metrics for evaluating scientific report quality, as standard NLG metrics (BLEU, ROUGE) correlate poorly with factuality ($r < 0.2$). The human evaluation protocol involved 5 domain experts (Cohen's $\kappa = 0.82$).

References

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... Amodei, D. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33, 1877-1901.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., ... Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35, 24824-24837.

Chen, G., Shen, Z., Li, A., Xiao, Y., Zhang, W., Gao, N. (2020). Polymer informatics: Current status and critical next steps. *Materials Science and Engineering: R: Reports*, 144, 100595.

Kim, C., Chandrasekaran, A., Huan, T., Das, D., Ramprasad, R. (2018). Polymer genome: A data-powered polymer informatics platform for property predictions. *The Journal of Physical Chemistry C*, 122(31), 17575-17585.

Zhang, Y., Xu, X., Hou, Z. (2023). Uncertainty quantification in polymer informatics using Bayesian neural networks. *Journal of Chemical Information and Modeling*, 63(8), 2345-2356.

Bran, A., Cox, S., White, A., Schwaller, P. (2023). ChemCrow: Augmenting large language models with chemistry tools. *Nature Machine Intelligence*, 5(10), 1123-1133.