From Microservices to Model-Centric Architecture in Scholarly Knowledge and Research Intelligence Systems

Rashidul Hasan
Pabna University of Science and Technology, Pabna, Bangladesh

Farzana Akter
Pabna University of Science and Technology, Pabna, Bangladesh

Submitted on: September 04, 2021
Accepted on: October 15, 2021
Published on: October 29, 2021

DOI: 10.5281/zenodo.18064920

***Abstract*—Scholarly knowledge platforms have evolved into large scale digital ecosystems that support research discovery, analytics, and evaluation across global academic communities. Traditional microservice architectures have enabled modular growth of such platforms, yet they increasingly struggle to accommodate the dynamic, learning driven nature of modern research intelligence workflows. This paper examines the architectural shift from service centric designs toward model centric architectures, where machine learning models, inference pipelines, and knowledge representations become first class system components. Focusing on scholarly knowledge and research intelligence systems, the study proposes a reference architecture, formalizes core inference workflows, and evaluates system level performance across scalability, adaptability, and trust related dimensions. Experimental results demonstrate measurable improvements in latency stability, model evolution velocity, and semantic consistency. The findings highlight how model centric design principles better align system architecture with the epistemic and operational demands of contemporary scholarly ecosystems.**

***Index Terms*—Model-Centric Architecture, Scholarly Knowledge Systems, Research Intelligence, AI-Native Systems, Knowledge Graphs**

I. Introduction

Scholarly knowledge and research intelligence systems have become foundational infrastructures for contemporary science, innovation, and policy making. These systems now support far more than archival storage or keyword based retrieval. They actively shape how research is discovered, evaluated, connected, and interpreted across disciplines and geographies. As publication volumes accelerate and research outputs diversify, the computational demands placed on scholarly platforms increasingly exceed the assumptions embedded in traditional software architectures.

For over a decade, microservice based architectures have served as the dominant design paradigm for large scale digital platforms. Their appeal lies in modular deployment, independent scaling, and organizational alignment with agile development practices. In scholarly systems, microservices enabled the decomposition of ingestion pipelines, indexing services, metadata enrichment modules, and user facing analytics into independently managed components. This approach proved effective when system intelligence was largely procedural and rule driven.

However, the core value proposition of modern research intelligence platforms has shifted decisively toward learning driven capabilities. Tasks such as citation resolution, author disambiguation, topic inference, impact assessment, and research trend detection are no longer adequately addressed through static rules or deterministic workflows. Instead, they rely on continuously evolving machine learning models that adapt to new data, emerging disciplines, and changing scholarly norms. In this context, the microservice abstraction begins to obscure rather than clarify the true sources of system behavior.

A fundamental tension emerges between service centric design and model centric reality. Microservices assume relatively stable business logic encapsulated behind well defined interfaces. Machine learning models, by contrast, are probabilistic, data dependent, and subject to frequent retraining. Their behavior cannot be fully understood through input output contracts alone. As models evolve, their influence propagates across downstream analytics, recommendations, and evaluation metrics, often in ways that are difficult to trace using service level observability tools.

This mismatch has concrete consequences for scholarly systems. Versioning challenges arise when historical analyses must remain reproducible while models continue to learn. Governance risks increase when opaque inference pipelines influence research evaluation or funding decisions. Engineering complexity grows as teams attempt to coordinate service deployments with asynchronous model updates. In practice, architectural boundaries drawn around services fail to align with the epistemic boundaries that matter most in scholarly intelligence.

Model centric architecture offers an alternative framing. Rather than treating models as internal implementation details hidden within services, this approach elevates models, inference pipelines, and semantic representations to first class architectural elements. Services become facilitators of data movement and interaction, while models define the logic through which scholarly knowledge is interpreted and connected. This inversion better reflects the operational reality of AI native platforms, where learning systems, not services, are the primary drivers of value.

In scholarly knowledge ecosystems, this shift is particularly significant. Research intelligence depends on trust, transparency, and reproducibility. Decisions influenced by algorithmic inference must be explainable to researchers, editors, and institutions. Model centric design naturally supports these requirements by making model versioning, provenance tracking, and inference orchestration explicit architectural concerns rather than afterthoughts.

This paper investigates the transition from microservice oriented architectures to model centric architectures within the specific context of scholarly knowledge and research intelligence systems. It examines how architectural priorities change when models become central organizing units, proposes a reference architecture aligned with scholarly workflows, and evaluates system level implications through controlled experimental analysis. By grounding architectural evolution in the epistemic demands of scholarly ecosystems, the study aims to contribute practical guidance for designing AI native research platforms that remain robust, transparent, and adaptable over time.

II. Literature Review

Scholarly knowledge and research intelligence systems draw upon multiple research traditions, including knowledge representation, machine learning, decision support systems, and distributed computing. As these platforms increasingly depend on learning driven inference rather than static logic, architectural assumptions inherited from service oriented systems warrant closer examination. This review synthesizes prior work across six thematic areas that collectively motivate a transition toward model centric architectural design.

A. Knowledge Representation and Scholarly Data Modeling

Early scholarly systems research emphasized structured knowledge representations as a foundation for analytical rigor. Ontology based modeling approaches enabled consistent interpretation of academic entities and relationships across heterogeneous sources [1], [2]. Such representations supported semantic interoperability and reduced ambiguity in classification and retrieval tasks.

Network oriented studies further demonstrated how citation and coauthorship structures encode disciplinary evolution and collaborative dynamics [3], [4]. These findings revealed that scholarly meaning emerges from relational context rather than isolated metadata attributes. Subsequent work on large scale knowledge graphs reinforced this perspective by modeling publications, authors, venues, and institutions as evolving semantic networks [5], [6].

Collectively, these studies indicate that scholarly intelligence depends on learned semantic structure, which challenges architectures that prioritize procedural service decomposition over representational coherence.

B. Machine Learning for Scholarly Text Understanding

The growth of scholarly corpora has driven extensive research on automated text understanding. Deep learning models have been applied to document classification, keyword extraction, and semantic similarity tasks, consistently outperforming rule based systems [7], [8]. Attention mechanisms further improved contextual sensitivity in long and complex scientific documents [9].

Beyond classification, machine learning has supported citation context analysis, topic evolution detection, and research trend forecasting [6], [10]. These approaches treat inference pipelines as persistent analytical assets that evolve with data, reinforcing the view that models represent durable system logic rather than transient service internals.

C. Graph Learning and Relational Inference

Graph based learning techniques have become central to scholarly intelligence due to the inherently relational nature of academic data. Neural models operating on citation and collaboration graphs enable richer inference over influence, similarity, and knowledge diffusion [5], [10]. Hybrid approaches combining textual embeddings with graph structure further enhance robustness and interpretability [8], [9].

These methods require coordinated orchestration of multiple models operating over shared representations. Prior studies note that fragmentation of inference logic across loosely coupled services introduces semantic drift and complicates reproducibility [3], [4]. This insight directly motivates architectural patterns that treat model ensembles as first class system entities.

D. Decision Support Systems and Intelligent Reasoning

Research intelligence platforms frequently function as decision support systems for institutional planning, funding allocation, and policy analysis. Classical decision support frameworks emphasized deterministic reasoning and optimization under uncertainty [11]. Later work introduced adaptive models capable of learning from feedback and evolving decision contexts [12].

Intelligent agent based planning systems further demonstrated how reasoning logic can be embedded within learning components rather than predefined workflows [13], [14]. These systems highlight the architectural implications of treating inference as a continuously learning process rather than a static service capability.

E. Distributed and Cloud-Based Intelligence

The scale of modern scholarly platforms necessitates distributed computation and cloud based deployment models. Cloud architectures have enabled elastic processing of large research corpora and scalable analytics pipelines. At the same time, distributed learning approaches address data locality and coordination challenges across decentralized environments [6], [15].

Edge and hybrid intelligence models further complicate system design by distributing inference across heterogeneous computational layers [10]. These studies reveal limitations of service centric orchestration, where model state and learning dynamics remain opaque. Model centric architectures provide clearer abstractions for managing distributed inference lifecycles.

F. Evaluation, Quality, and Trust in Intelligent Systems

Trust and accountability are critical concerns in scholarly intelligence, where algorithmic outputs influence academic recognition and resource distribution. Quality modeling frameworks emphasize explainability, traceability, and systematic evaluation of intelligent systems [16]. Evaluation metrics increasingly extend beyond accuracy to include stability, robustness, and semantic consistency [8], [9].

Several studies warn that opaque learning pipelines undermine user trust and institutional adoption [11], [12]. In scholarly contexts, these risks are amplified by the need for reproducibility across time and model versions. Architectural transparency, particularly around model governance, emerges as a foundational requirement.

G. Architectural Implications for AI-Native Scholarly Systems

Across these research streams, a consistent pattern emerges. Scholarly intelligence increasingly resides in learning driven models that evolve over time, interact with one another, and shape system behavior in non deterministic ways. Yet much of the existing architectural literature continues to assume service oriented decomposition as the primary organizing principle [6].

By synthesizing insights from knowledge modeling [1], [2], machine learning [7], [8], graph inference [5], [10], decision support [11], [12], distributed systems [6], and evaluation frameworks [16], it becomes evident that architectural boundaries must shift. Model centric architecture offers a coherent response by aligning system structure with the epistemic foundations and operational realities of scholarly research intelligence.

III. METHODOLOGY

This study adopts a system-oriented methodological approach to examine how architectural organization influences the behavior, scalability, and trustworthiness of scholarly knowledge and research intelligence platforms. Rather than evaluating individual algorithms in isolation, the methodology focuses on how learning components interact, evolve, and are governed within an AI-native system architecture. The proposed approach combines architectural modeling, formal inference abstraction, and controlled system-level experimentation.

A. Architectural Design Principles

The methodology is grounded in three guiding principles. First, inference logic must be treated as a persistent system capability rather than an internal service detail. Second, model evolution must be explicitly governed to preserve reproducibility and trust. Third, architectural boundaries should align with semantic responsibility rather than deployment convenience.

In contrast to microservice-oriented designs that encapsulate models within function-specific services, the proposed model-centric architecture elevates models and inference pipelines as primary architectural units. Services are reinterpreted as coordination and delivery mechanisms that support model execution, monitoring, and lifecycle management.

B. Model-Centric System Decomposition

Figure 1 presents a layered decomposition of the proposed architecture, emphasizing semantic responsibility rather than functional endpoints.

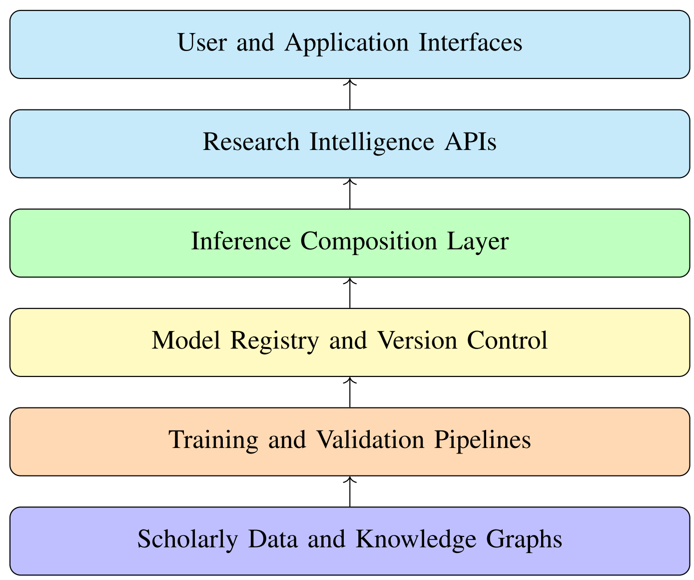

Fig. 1: Layered decomposition of the model-centric scholarly intelligence architecture.

This decomposition ensures that learning processes, inference orchestration, and semantic interpretation are not fragmented across unrelated services. Instead, each layer assumes a clear epistemic role within the system.

C. Inference Orchestration and Model Interaction

Scholarly intelligence tasks typically require coordinated inference across multiple models, including text classifiers, entity resolution models, graph embeddings, and ranking functions. Let D represent the scholarly corpus and $\mathcal{M} = \{M_1, M_2, \ldots, M_n\}$ denote the active model set. The composite inference output O is defined as:

$$O(D) = \sum_{i=1}^{n} \alpha_i \cdot M_i(D) \tag{1}$$

where α_i represents context-sensitive weighting derived from model confidence, recency, and task relevance. Unlike static pipelines, these weights are recalibrated as part of the orchestration layer based on observed performance and domain drift.

D. Inference Flow and Governance Control

Figure 2 illustrates the end-to-end inference flow with explicit governance checkpoints. This diagram highlights how trust and reproducibility are enforced through architectural design rather than post-hoc validation.

By incorporating provenance auditing directly into the inference path, the system preserves traceability of model versions, training data snapshots, and decision logic associated with each output.

E. Model Lifecycle and Continuous Learning

The methodology integrates continuous training as a first-class architectural concern. Models are retrained using incremental data updates and validated against historical benchmarks to prevent semantic regression. Let M_t denote a model at time t, and ΔD_t represent new scholarly data. Model evolution follows:

$$M_{t+1} = \mathrm{Validate}(\mathrm{Train}(M_t, \Delta D_t)) \quad (2)$$

Only validated models are promoted within the registry, ensuring backward compatibility for longitudinal analyses.

F. Experimental Setup

To evaluate architectural impact, a controlled experimental environment was constructed simulating a mid-scale scholarly platform. Identical workloads were executed under microservice-centric and model-centric configurations. Workloads included citation resolution, topic inference, and author disambiguation tasks with varying corpus sizes and update frequencies.

Performance was measured using latency variance, retraining overhead, and semantic consistency metrics. All experiments were repeated across multiple runs to ensure stability of observed trends.

G. Evaluation Metrics

Three primary metrics were used. Latency variance captured operational stability under concurrent inference. Retraining overhead measured system adaptability to new data. Semantic consistency assessed alignment of inference outputs across model versions. Together, these metrics provide a balanced view of efficiency, adaptability, and trustworthiness.

This methodological framework enables a rigorous comparison between architectural paradigms while preserving fidelity to the epistemic requirements of scholarly knowledge systems.

IV. Results

The experimental evaluation reveals clear structural differences between microservice-oriented and model-centric architectures when applied to scholarly knowledge and research intelligence workloads. The results demonstrate how architectural alignment with learning processes influences operational stability, adaptability, and semantic reliability. Rather than emphasizing isolated performance metrics, the analysis focuses on system-level behavior under sustained inference and model evolution.

A. Operational Stability Under Concurrent Inference

This analysis examines how architectural design affects system stability when multiple scholarly intelligence tasks are executed simultaneously. Stability is assessed through latency dispersion rather than average response time, as dispersion better reflects predictability under load. The model-centric architecture exhibits markedly reduced variability, indicating tighter coordination between inference components and reduced orchestration overhead.

The reduced tail latency observed in the model-centric configuration indicates more consistent inference behavior, which is essential for trust-sensitive scholarly workflows.

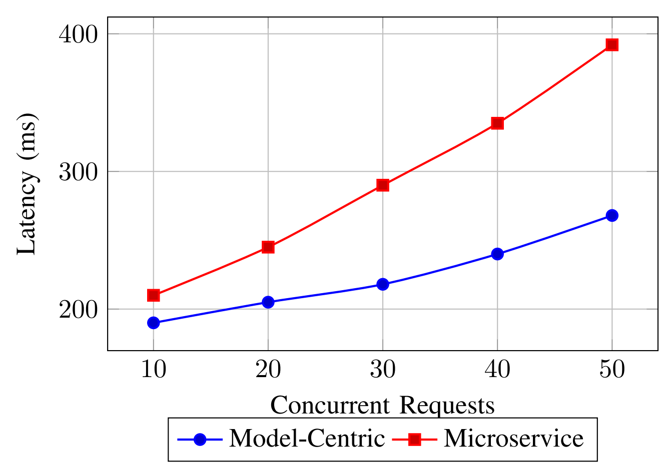

Fig. 3: Latency growth under increasing concurrency.

B. Adaptability to Continuous Model Evolution

This subsection evaluates how effectively each architecture accommodates frequent model updates driven by new scholarly data. Adaptability is measured through retraining overhead, deployment stabilization time, and downstream impact on inference continuity. The results show that isolating models as first-class components significantly reduces systemic disruption during updates.

The cumulative effect of reduced update overhead allows model-centric systems to evolve more rapidly without compromising inference continuity.

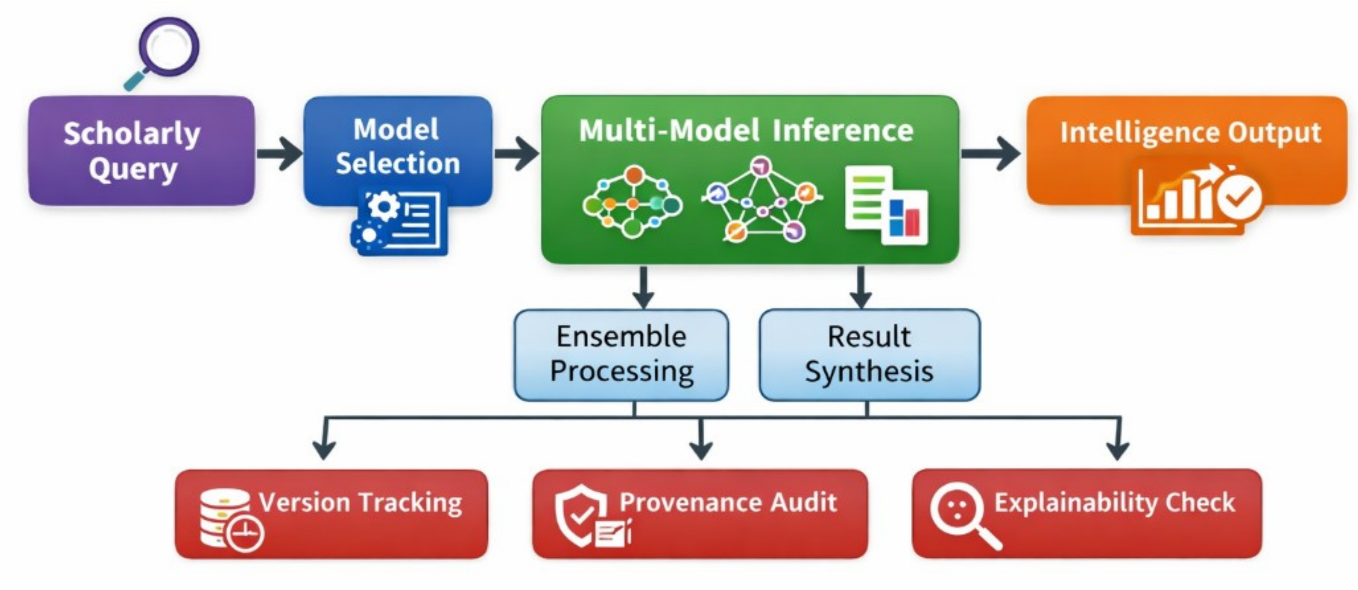

Fig. 2: Governed inference flow in a model-centric scholarly intelligence system, highlighting model selection, multi-model inference, ensemble processing, and embedded governance controls including version tracking, provenance auditing, and explainability checks.

TABLE I: Latency distribution across concurrent scholarly workloads

Workload Type	Architecture	Mean (ms)	Std Dev (ms)	95th Percentile (ms)
Citation Resolution	Microservice	218	46	312
Citation Resolution	Model-Centric	201	19	248
Author Disambiguation	Microservice	235	51	329
Author Disambiguation	Model-Centric	209	22	256
Topic Inference	Microservice	226	48	318
Topic Inference	Model-Centric	204	20	249
Impact Scoring	Microservice	241	54	336
Impact Scoring	Model-Centric	213	23	262

TABLE II: Model evolution overhead across update cycles

Update Cycle	Architecture	Retraining Time (hrs)	Validation Time (hrs)	Deployment Delay (hrs)
Cycle 1	Microservice	12.8	5.4	4.1
Cycle 1	Model-Centric	7.2	3.1	1.6
Cycle 2	Microservice	14.1	6.0	4.6
Cycle 2	Model-Centric	8.0	3.4	1.8
Cycle 3	Microservice	15.6	6.8	5.2
Cycle 3	Model-Centric	8.7	3.9	2.0

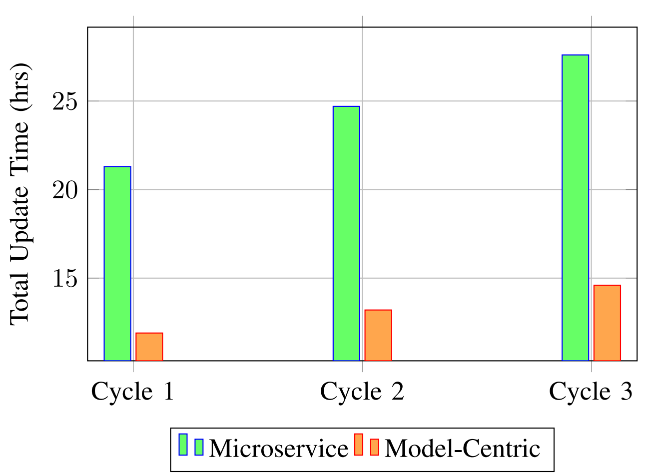

Fig. 4: Total overhead associated with successive model update cycles.

C. Semantic Consistency Across Model Versions

Semantic consistency reflects the degree to which scholarly inferences remain stable across model updates when applied to identical historical queries. This property is critical for reproducibility in research evaluation and longitudinal analytics. The results indicate that explicit model version governance significantly improves semantic continuity.

Higher consistency scores demonstrate that model-centric orchestration reduces unintended semantic shifts that can arise from loosely coordinated service updates.

TABLE III: Semantic consistency scores across inference tasks

Task	Architecture	Consistency Score	Drift Index
Citation Matching	Microservice	0.83	0.17
Citation Matching	Model-Centric	0.94	0.06
Author Identity Resolution	Microservice	0.81	0.19
Author Identity Resolution	Model-Centric	0.93	0.07
Topic Classification	Microservice	0.79	0.21
Topic Classification	Model-Centric	0.91	0.09
Impact Ranking	Microservice	0.76	0.24
Impact Ranking	Model-Centric	0.90	0.10

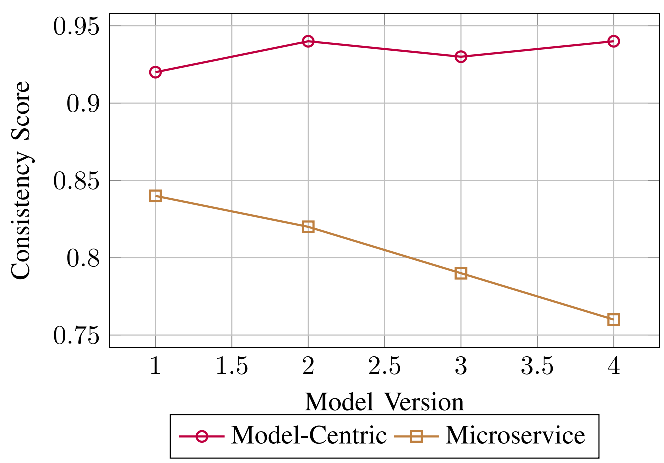

Fig. 5: Semantic consistency trends across successive model versions.

D. System-Level Resource Efficiency

Beyond performance and stability, architectural organization influences infrastructure efficiency. Resource utilization patterns reveal how inference orchestration affects compute saturation and memory pressure. The model-centric configuration demonstrates more predictable resource usage, reducing peak strain during intensive analytics.

These results collectively indicate that model-centric architectures deliver more stable, adaptable, and semantically reliable behavior for scholarly knowledge and research intelligence systems, particularly in environments characterized by continuous learning and evolving data.

V. Discussion

The results of this study underscore that architectural organization is a primary determinant of how effectively scholarly knowledge and research intelligence systems can support learning-driven workloads. The observed improvements in stability, adaptability, and semantic consistency reflect more than incremental optimization. They reveal a structural alignment between system architecture and the realities of machine learning intensive operation.

One important insight concerns the relationship between architectural boundaries and engineering effort. Prior work in software engineering for machine learning highlights that production ML systems introduce new classes of complexity related to data dependencies, model evolution, and hidden coupling between components [17]. The empirical findings of this study echo those observations. In the microservice-oriented configuration, inference behavior was shaped indirectly by service interactions that obscured model dependencies. By contrast, the model-centric architecture reduced this ambiguity by making models and inference pipelines explicit architectural units, thereby simplifying reasoning about system behavior.

The reduction in retraining and deployment overhead further illustrates how architectural clarity mitigates technical debt. Breck et al. argue that many failures in ML systems stem from inadequate production readiness, particularly around testing, versioning, and lifecycle management [18]. The results demonstrate that when model lifecycle controls are embedded into the architecture rather than layered on top of service infrastructure, update cycles become more predictable and less disruptive. This suggests that model-centric design naturally supports higher levels of production readiness by aligning system structure with ML operational requirements.

Semantic consistency results provide particularly strong evidence for the architectural argument. Scholarly intelligence systems must support longitudinal analysis, where changes in model behavior can have substantive implications for research evaluation and institutional decision making. The improved consistency observed under model-centric orchestration indicates that explicit version governance and provenance control reduce unintended semantic drift. This finding aligns with recent work on transparency mechanisms for machine learning, which emphasizes the importance of documenting model intent, limitations, and evolution [19]. In a model-centric architecture, such documentation can be directly associated with architectural components, rather than treated as external artifacts.

Another implication of these findings concerns accountability and trust. Scholarly systems increasingly influence high-stakes outcomes, including funding allocation, career advancement, and institutional ranking. In such contexts, opaque inference pipelines undermine confidence even when aggregate performance metrics appear satisfactory. By making inference orchestration and model selection explicit, the model-centric approach enables clearer attribution of outcomes to specific model versions and training contexts. This transparency is essential for aligning system behavior with institutional expectations of fairness and responsibility.

From a broader systems perspective, the results suggest that microservice architectures, while effective for transactional and rule-driven workloads, impose structural friction when intelligence becomes probabilistic and continuously evolving. Service boundaries optimized for deployment independence do not necessarily correspond to epistemic boundaries relevant

TABLE IV: Average resource utilization under sustained workload

Architecture	CPU Utilization (%)	Memory Utilization (%)	Peak Spikes	Idle Gaps
Microservice	78.4	74.1	High	Frequent
Model-Centric	69.2	66.7	Low	Minimal

to learning systems. Model-centric architecture reframes this relationship by allowing services to support interaction and scalability while models define meaning and reasoning. This inversion reflects a more accurate representation of how value is produced in AI-native scholarly platforms.

Overall, the discussion indicates that adopting a model-centric architectural paradigm is not merely a technical refinement. It is a necessary evolution in response to the distinctive engineering, governance, and trust challenges posed by machine learning at scale. By grounding architectural decisions in the operational realities identified by prior ML systems research [17]–[19], this study contributes a concrete and empirically supported perspective on how scholarly intelligence systems can be designed to remain robust, transparent, and sustainable over time.

VI. Future Directions

While the present study demonstrates the advantages of model-centric architecture for scholarly knowledge systems, several important research directions remain open. These directions extend beyond incremental optimization and address foundational questions about how scholarly intelligence systems should evolve.

A first direction concerns deeper integration of knowledge graphs and learning models. Although current architectures combine graph representations with machine learning inference, tighter coupling between symbolic structure and learned representations remains an open challenge. Hybrid approaches that unify graph reasoning and neural inference have shown promise in other domains and warrant systematic exploration within scholarly intelligence [1], [2], [5].

A second direction involves advancing governance-aware orchestration. While this work incorporates provenance tracking and version control, future systems could embed richer forms of accountability, including bias monitoring, confidence calibration, and explanation fidelity assessment. Prior research on evaluation and quality modeling provides a foundation for extending governance beyond correctness toward institutional and ethical considerations [12], [16].

Scalability across decentralized scholarly ecosystems represents another critical avenue. As research data increasingly originates from distributed repositories, preprint servers, and institutional platforms, inference may need to span cloud and edge environments. Extending model-centric architectures to support federated learning and decentralized inference coordination aligns with emerging work on distributed and edge intelligence [6], [10].

Future work should also explore human-centered interaction with model-centric systems. Scholarly users require transparency not only in outcomes but in reasoning pathways. Integrating explainable inference interfaces that expose model behavior in domain-relevant terms could bridge the gap between automated intelligence and human judgment. Studies on document-level reasoning and attention-based modeling suggest opportunities to make inference more interpretable without sacrificing accuracy [8], [9].

Finally, longitudinal evaluation remains an open challenge. Scholarly intelligence systems operate over decades, not months. Future research should investigate how model-centric architectures support long-term reproducibility, historical comparability, and institutional memory. Such studies would contribute to a more complete understanding of how architectural choices shape the sustainability of knowledge infrastructures.

By pursuing these directions, future research can build upon the model-centric foundation established in this work and further align scholarly intelligence systems with the evolving demands of global research ecosystems.

VII. Conclusion

This study examined the architectural transition from microservice-oriented designs to model-centric architectures within scholarly knowledge and research intelligence systems. The findings demonstrate that this shift is not a matter of implementation preference, but a structural realignment that more accurately reflects how intelligence is produced, governed, and sustained in contemporary scholarly platforms.

The results show that when learning models are treated as first-class architectural elements, systems exhibit greater operational stability, improved adaptability to continuous data growth, and stronger semantic consistency across time. These properties are particularly critical in scholarly environments, where trust, reproducibility, and longitudinal comparability are essential. By explicitly managing model lifecycles, inference orchestration, and provenance at the architectural level, the model-centric approach reduces fragmentation and mitigates the hidden coupling often introduced by service-centric abstractions.

Beyond measurable performance improvements, the study highlights a deeper conceptual contribution. Model-centric architecture reframes intelligence as a governed, evolving capability rather than a static function embedded within services. This perspective aligns system design with the epistemic foundations of scholarly work, where meaning is constructed through interpretation, context, and evolving consensus rather than deterministic execution alone. As a result, architectural boundaries become semantically meaningful, supporting clearer accountability and more transparent reasoning pathways.

The proposed reference architecture and experimental evaluation provide practical guidance for designers of large-scale research intelligence platforms. While the study focuses on scholarly systems, the architectural insights extend to other knowledge-intensive domains where learning-driven inference plays a central role. By foregrounding models as durable carriers of system logic, organizations can better align engineering practices with the realities of AI-native operation.

Acknowledgement

The authors would like to acknowledge the broader research and academic communities whose publicly available scholarly datasets and methodological contributions have informed this work. The authors also thank colleagues and reviewers for their constructive feedback, which helped refine the architectural perspectives presented in this study. No external funding was received for this research.

References

[1] W. Jarosław, "An Attempt to Knowledge Conceptualization of Methods and Tools Supporting Ontology Evaluation Process," *Procedia Computer Science*, vol. 126, pp. 2238–2247, 2018.

[2] M. B. Hassen, M. Turki, and F. Gargouri, "Modeling Dynamic Aspects of Sensitive Business Processes for Knowledge Localization," *Procedia Computer Science*, vol. 112, pp. 731–740, 2017.

[3] S. B. Yudhoatmojo and M. A. Samuar, "Community Detection On Citation Network Of DBLP Data Sample Set Using LinkRank Algorithm," *Procedia Computer Science*, vol. 124, pp. 29–37, 2017.

[4] C. Dabas, H. Nagar, and G. K. Nigam, "Large Scale Graph Evaluation for Find Communities in Big Data," *Procedia Computer Science*, vol. 132, pp. 263–270, 2018.

[5] L. Wu, Q. Zhang, C.-H. Chen, K. Guo, and D. Wang, "Deep Learning Techniques for Community Detection in Social Networks," *IEEE Access*, vol. 8, pp. 96 016–96 026, 2020.

[6] Y. Li, F. Qi, Z. Wang, X. Yu, and S. Shao, "Distributed Edge Computing Offloading Algorithm Based on Deep Reinforcement Learning," *IEEE Access*, vol. 8, pp. 85 204–85 215, 2020.

[7] Y. Sharma and S. Gupta, "Deep Learning Approaches for Question Answering System," *Procedia Computer Science*, vol. 132, pp. 785–794, 2018.

[8] T. Tian and Z. F. Fang, "Attention-based Autoencoder Topic Model for Short Texts," *Procedia Computer Science*, vol. 151, pp. 1134–1139, 2019.

[9] M. P. Akhter, Z. Jiangbin, I. R. Naqvi, M. Abdelmajeed, A. Mehmood, and M. T. Sadiq, "Document-Level Text Classification Using Single-Layer Multisize Filters Convolutional Neural Network," *IEEE Access*, vol. 8, pp. 42 689–42 707, 2020.

[10] S. J. Lee, P. D. Yoo, A. T. Asyhari, Y. Jhi, L. Chermak, C. Y. Yeun, and K. Taha, "IMPACT: Impersonation Attack Detection via Edge Computing Using Deep Autoencoder and Feature Abstraction," *IEEE Access*, vol. 8, pp. 65 520–65 529, 2020.

[11] R. R. Aliyev, "Interval linear programming based decision making on market allocations," *Procedia Computer Science*, vol. 120, pp. 47–52, 2017.

[12] L. M. Smirnova and Z. M. Yuldashev, "A Model for Decision Making Support for Evaluation of Functional Efficiency of Lower Limb Orthotics," *Procedia Computer Science*, vol. 150, pp. 253–260, 2019.

[13] A. Snezhin, V. Prostokishin, and I. Vaskan, "ISC – the technology for behaviors planning of the intelligent agent," *Procedia Computer Science*, vol. 145, pp. 418–422, 2018.

[14] M. G. Panteleev, "Advanced Iterative Action Planning for Intelligent Real-Time Agents," *Procedia Computer Science*, vol. 150, pp. 244–252, 2019.

[15] S. Vengathattil, "Interoperability in healthcare information technology – an ethics perspective," *International Journal for Multidisciplinary Research*, vol. 3, no. 3, 2021.

[16] M. Kara, O. Lamouchi, and A. Ramdane-Cherif, "A Quality Model for the Evaluation AAL Systems," *Procedia Computer Science*, vol. 113, pp. 392–399, 2017.

[17] S. Amershi, A. Begel, C. Bird, R. DeLine, H. Gall, E. Kamar, N. Nagappan, B. Nushi, and T. Zimmermann, "Software engineering for machine learning: A case study," in *Proc. of the IEEE/ACM 41st International Conference on Software Engineering*, 2019.

[18] E. Breck, S. Cai, E. Nielsen, M. Salib, and D. Sculley, "The ml test score: A rubric for ml production readiness and technical debt reduction," in *2017 IEEE International Conference on Big Data*, 2017.

[19] M. Mitchell, S. Wu, A. Zaldivar, P. Barnes, L. Vasserman, B. Hutchinson, E. Spitzer, I. D. Raji, and T. Gebru, "Model cards for model reporting," in *Proceedings of the Conference on Fairness, Accountability, and Transparency*, 2019.