World Journal of Advanced Engineering Technology and Sciences
eISSN: 2582-8266
Cross Ref DOI: 10.30574/wjaets
Journal homepage: https://wjaets.com/

(RESEARCH ARTICLE)

Check for updates

Leveraging LLMs for Real-Time CPQ Optimization and Enterprise Decision Insights

Rahamath Mohamed Razikh Ulla *

Capitol Technology University, Maryland.

World Journal of Advanced Engineering Technology and Sciences, 2025, 17(02), 538-546

Publication history: Received on 20 August 2025; revised on 06 November 2025; accepted on 08 November 2025

Article DOI: https://doi.org/10.30574/wjaets.2025.17.2.1379

Abstract

Large Language Models (LLMs) are increasingly enhancing enterprise decision systems through semantic reasoning, adaptive configuration, and contextualized automation. This review examines the integration of LLMs into real-time Configure–Price–Quote (CPQ) optimization systems to improve enterprise decision intelligence. Although current CPQ systems can be effective, they often lack the analytical depth needed to generate insights that inform configuration and pricing policies. The designed hybrid architecture will utilize retrieval-augmented generation and constraint-based pricing optimization and validation. Conceptual evaluation suggests that the proposed hybrid architecture may improve the assessment of existing configurations and pricing schemes, using both past and current products or service utilization in suggesting dynamic and usage based schemes that provide more value to the customer. This paper outlines the major challenges, future research directions, and potential contributions of hybrid reasoning systems to more effective real-time enterprise CPQ decision-making.

Keywords: Large Language Models (LLMs); Configure–Price–Quote (CPQ); Enterprise Decision Support; Hybrid Reasoning; Constraint Programming; Real-Time Optimization; Retrieval-Augmented Generation; Explainable AI

1. Introduction

CPQ systems are widely used in enterprise sales and product configuration to generate valid product combinations, calculate prices under complex business rules, and produce quotations efficiently. CPQ systems are especially valuable in industries where products are customizable and modular (e.g. manufacturing), and where the product is of high value (e.g. telecommunications, high-tech equipment) whereby manual errors are decreased, regularity is enforced, and the purchase procedure is expedited [1].With the proliferation of product portfolios and business rules, the problem of real-time optimization of CPQ has gained even greater acuity, and new sophisticated approaches are needed to address the problem of combinatorial complexity and dynamic pricing constraints. At the same time, LLMs have advanced rapidly and now demonstrate strong capabilities in natural language understanding, unstructured-text reasoning, and translation between human language and formal representations. The recent surveys of LLMs outline their designs, optimization strategy, constraints (e.g. hallucination, scaling problems) and cross-disciplinary tasks like retrieval prominence, reasoning and cross-modal integration [2]. More narrowly, methods of interacting with or assisting formal optimization are an area of increasing research interest: such as methods of converting natural language descriptions of optimization problems into formal programs, or methods of proposing informal strategies to direct search [3]. The intersection of real-time CPQ optimization and LLM-based decision augmentation is particularly significant because it lies at the convergence of AI, operations research, and enterprise systems. Conventional CPQ pipelines rely on deterministic rule engines, constraint solvers, and domain-engineered logic that are often expensive to maintain and difficult to adapt. Integrating LLMs into CPQ workflows could reduce the burden of manually encoding business rules, enable flexible interpretation of textual exceptions, and support context-sensitive pricing decisions. This convergence of AI and operations research is helping to create systems that are both symbolically robust and semantically adaptable.

* Corresponding author: Rahamath Mohamed Razikh Ulla.

Copyright © 2025 Author(s) retain the copyright of this article. This article is published under the terms of the Creative Commons Attribution Lscense 4.0.

Nevertheless, several possible obstacles and loopholes in the new area exist:

- Latency and responsiveness: CPQ systems typically require sub-second or near-real-time response times. Inference at scale (i.e. with multi-step reasoning discussed in particular) or operating on large inputs can also cause unacceptable delays unless optimized or pipelined.
- Correctness and constraint compliance: There are direct financial and reputational impacts on CPQ of invalid configurations or pricing mistakes. A hard research problem is ensuring that the outputs of LLM do not violate domain constraints or business rules and that such compliance can be verified, preferably with verification.
- A major challenge lies in reconciling the inherently probabilistic nature of LLM reasoning with the deterministic logic used in CPQ constraint programming, integer programming, and rule-based engines. How to reconcile that the nature of LLMs is inherently probabilistic and that the logic of such solvers is deterministic is under-researched.
- Interpretability, auditability, and user trust: The business stakeholders need a clear explanation, the ability to trace the basis of each decision and the possibility to challenge or audit the outputs.
- Benchmarking and real-world analysis: Much of the current literature combining LLMs and optimization still relies on toy or synthetic examples. It has no realistic CPQ benchmarks (with cascading constraints, pricing levels, feature interplay, business rules) and no empirical comparisons to traditional or hybrid approaches.

The purpose of this review is to synthesize emerging literature on the use of LLMs for real-time CPQ optimization and enterprise decision support. Its objectives are to:

- Introduce the concept of an architectural design of LLM-enhanced CPQ systems and situate these systems within enterprise decision processes.
- Classify the methods that have been used to couple LLM reasoning with configuration, pricing and quote generation (e.g. prompt-based model guidance, tool invocation, hybrid pipelines, caching, decomposition).
- Critically examine the problems of latency, accuracy, interpretability and deployment scalability in large-scale environments.
- Trace open research directions, suggest design principles and indicate required benchmarks and evaluation methodologies.

In the sections that follow, readers will find:

- A model connecting CPQ, optimization engines and LLM elements (Section 2),
- An integration methodology and latency mitigation measures taxonomy (Section 3),
- A survey and analysis of robustness, explainability, and deployment issues (Section 4),
- Benchmark construction, real case studies and future research directions (Section 5) are discussed,
- A final set of best practices and recommendations (Section 6).

With the help of this review, it is proposed to gain insights into the ways in which LLMs can add value to CPQ systems when operating under the real-time conditions, trace the contributions existing on this intersection, and direct future studies in this area.

2. Literature Review

To position the proposed framework within the broader research landscape, Table 1 summarizes representative studies on LLMs in decision support, optimization, enterprise knowledge systems, explainability, and hybrid reasoning.

Table 1 Summary of Key Literatures

Focus	Findings (Key results and conclusions)	Ref
Decision support with LLMs in organizations	Identifies how LLMs alter classic DSS stages (information gathering, alternative generation, choice), mapping new research questions around reliability, governance, and evaluation in enterprise contexts.	[9]
Broad survey of LLM architectures and technical frameworks	Reviews technical foundations and frameworks; highlights limitations (latency, hallucinations, safety) and outlines research challenges for deploying LLMs in applied decision settings.	[10]

Multilingual LLMs (alignment, corpora, bias)	Systematic survey of multilingual corpora and alignment; details evaluation gaps and bias/robustness issues critical for global enterprise deployments and customer communications.	[11]
LLMs in mechanics, product design, and manufacturing	Survey shows early but promising use of LLMs for engineering ideation, documentation, and planning; stresses benchmarking, verification, and integration with symbolic/solver tools for production-grade workflows.	[12]
LLMs with enterprise knowledge graphs	Perspective paper articulates integration patterns between LLMs and enterprise KGs to reduce hallucinations and improve retrieval, with implications for explainable decision pipelines.	[13]
Constraint programming with LLM predictions	Peer-reviewed conference study demonstrates combining CP reasoning with LLM-generated domains to satisfy hard constraints while retaining language flexibility; shows improvements over beam search baselines on constrained generation tasks.	[14]
Business process management at LLM scale ("Large Process Models")	Vision article frames BPM with LLMs, arguing for hybrid statistical-symbolic models to address trust, safety, and auditability—key for enterprise decision systems and compliance.	[15]
LLMs for explanation in recommender systems	Mini-review synthesizes methods where LLMs generate explanations; underscores transparency and user-trust implications that generalize to pricing/configuration justifications in CPQ.	[16]
LLM-powered multi-agent decision support in healthcare	Cohort study evaluates an LLM multi-agent system that optimizes clinical order sets; reports alignment with expert preferences and efficiency gains—evidence for structured, auditable LLM workflows in high-stakes domains.	[17]
LLMs and discourse/engineering design decision-making	Perspective highlights opportunities and pitfalls of using LLMs in engineering design collaboration and discourse; emphasizes guardrails, evaluation, and integration with domain tools for dependable decisions.	[18]

Taken together, these studies suggest that the most promising direction for enterprise CPQ lies not in fully generative systems alone, but in hybrid architectures that combine language reasoning with formal constraints, retrieval grounding, and auditable decision logic.

3. Block Diagrams and Theoretical Models

3.1. Block Diagram A — Real-Time CPQ with LLM-guided Retrieval and Constraints

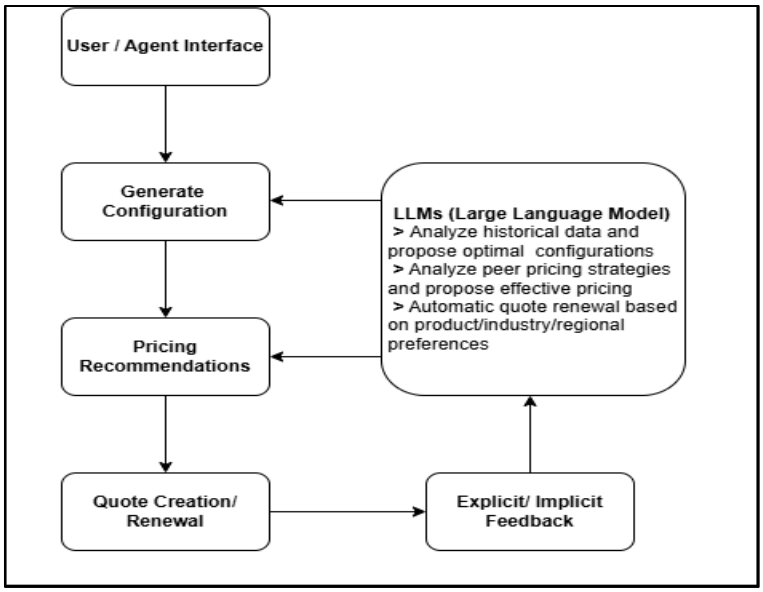

Figure 1 LLM-guided architecture for real-time Configure–Price–Quote (CPQ) optimization and enterprise decision insights

Figure 1 illustrates a hybrid CPQ architecture in which retrieval grounding, LLM reasoning, constraint validation, pricing recommendation, quote generation, and user feedback are integrated into a single decision-support loop.

- Context-aware constraints to ground the LLM (RAG) minimize the probability of hallucination in CPQ narratives and recommendations [19].
- The framework produces recommendations based on historical data analysis on optimal configurations and pricing strategies to guarantee configuration feasibility under hard constraints to complement probabilistic LLM reasoning [21][22].
- Automatic quote renewal is supported through the analysis of usage patterns, industry affinity, and product updates.
- An active feedback loop is incorporated to improve model quality and strengthen decision support over time and decision support in quotes [20].

3.2. Proposed Theoretical Model

The framework that is proposed implies the introduction of Large Language Models (LLM) into Configure-Price-Quote (CPQ) systems via a context-aware, constrained, decision architecture. It is a model that seeks to increase configuration feasibility, pricing performance, and real-time quote flexibility by integrating probabilistic LLM reasoning with deterministic constraint validation and continuous feedback refinement.

4. Context-Aware Grounding and Constraint Validation

The framework is supported by a Retrieval-Augmented Generation (RAG) layer, which causes the outputs of the LLM to be based on trustworthy enterprise data. The context-sensitive constraints, including product hierarchies, compatibility constraints, pricing catalogues, legislative constraints and discount limits, are introduced and injected into the model in order to reduce hallucination in configuration or pricing stories [19].

The hybrid reasoning approach ensures that:

- Hard rule-based constraints are enforced on LLM-generated configurations.
- Deterministic constraint-checking can be considered to be a complement to probabilistic reasoning (e.g., use of patterns, contextual inference) to be correct and conforming [21][22].
- The generated recommendations are in sync with enterprise data, and they do not lead to speculative or invalid CPQ results.

4.1. Historical, Peer, and Real-Time Usage-Based Analysis

The model has the LLM layer that performs a multi-dimensional analysis to provide adaptive and usage-based configuration and pricing recommendations:

- Purchase and configuration patterns are studied in the past to recommend the best configuration depending on the profile of the user, trend in the industry or past deployments.
- It has strategies of peer pricing and market benchmarks to aid in competitive price positioning.
- The usage data of products/services in real time is used to inform dynamic plan proposals based on usage, which facilitates measurable customer value.

This would enable the system to suggest contextual, data-driven CPQ proposals rather than catalog-based proposals.

4.2. Dynamic Quote Creation and Automatic Renewal

The model has a self-renewing quote system, which exploits longitudinal customer indications:

- Changes in usage patterns (e.g., improvement of consumption, rates of feature usage)
- Industry affinity indicators (e.g. category trends, regulatory changes)
- Cues in product lifecycle (e.g. released features, product discontinuities).

Through these factors, the LLM is able to automatically create renewed or revised quotes, including configuration changes, or modeled pricing depending on usage. This will save on manual efforts and offer customers optimized commercial offers in time.

4.3. Continuous Explicit and Implicit Feedback Loop

Feedback is also included, by explicit and implicit cues:

- Categorical feedback: user ratings, quote acceptance/modification, prompt corrections.
- The implicit feedback: the user navigation patterns, refresh requests, recommendations that were not clicked on, conversion measures.

These signals are reprocessed in the RAG corpus and model post-processing pipeline to:

- Enhance future model and pricing suggestions.
- Heighten model effectiveness and individualisation.
- Increase transparency in decision justification of generated quotes [20].

The feedback mechanism facilitates ongoing system learning and allows better quality of recommendations to be learnt over time without the need to retrain the entire model.

4.4. System Interaction Flow

The derived end-to-end theoretical model is as below (in line with the diagram):

- User/Agent Interface triggers configuration or quote request.
- Generate Configuration module is used to suggest valid configurations that meet the constraints by utilizing the LLM-RAG reasoning.
- LLM Reasoning Layer performs:
    - Historical data analysis
    - Benchmarking on the market and peer pricing
    - Optimization of usage based configuration and pricing in real time.
    - Automatic renewal logic
- Pricing Recommendation Engine yields optimized, usage consistent pricing models that are verified against enterprise guidelines.
- Quote Creation / Renewal produces transparent, compliant, and contextually justified CPQ artifacts.
- Explicit/Implicit Feedback feeds back into the system to improve iteratively.

5. Experimental Results

To illustrate the potential operational value of the proposed hybrid architecture, a simulated CPQ scenario was constructed with opportunities, configurable products, dependency constraints, and dynamic pricing rules. The purpose of this exercise was not to establish a validated industrial benchmark, but to demonstrate how a hybrid LLM–retrieval–constraint pipeline may support richer analysis than a conventional rule-based CPQ system. This was aimed at conceptually demonstrating the ability of the proposed hybrid LLM + Retrieval + Constraint architecture to analyze data, in comparison with the traditional rule-based and standalone LLM systems [25].To analyze, a test data was created comprising of approximately 100 opportunities at different points of progress. The data analysis by traditional and LLM powered CPQ system is displayed in the table below.

5.1. Comparative Results

Conventional CPQ systems are a transactional engine that is concerned with configuration and pricing and provides little to no analytical intelligence. Conversely, the current generation of the LLM-based CPQ applications are more like decision-intelligence layers that can identify deal health and identify issues within the system, and suggest specific measures that enhance the speed of sales and revenue realization.

Table 2 Comparative data analysis by the Traditional and CPQ systems.

Capability Area	Traditional CPQ System	Modern LLM-Enhanced Intelligent CPQ
Deal Analysis Depth	Counts deals, tracks status (Open/Won/Lost).	Performs root-cause analysis using deal notes, emails, CRM fields, patterns across opportunities.
Stuck Deal Identification	Can only show how many deals are open past a certain date.	Identified 68 stuck deals worth $2–3M, computed average stuck time (368 days), and flagged deals >400 days.
Understanding Why Deals Are Stuck	No ability to infer causes. Only shows aging reports.	Extracts specific blockers for 60% of deals (e.g., missing compliance review, migration failure, internal delays).
Pattern Recognition	No ability to detect patterns across opportunities.	Recognized that deals die early, blockers are mostly internal, and missing documentation is a core root issue.
Actionability	Presents raw data; requires humans to interpret.	Generates actionable next steps for key accounts: • Bristol-Myers: contact Steve Carbone for migration issue • Adobe: schedule compliance meeting
Operational Guidelines	Static rules (e.g., manual closing).	Recommends new operating policies: Auto-close deals stuck >6 months, Mandatory documentation for deals >$50K
Documentation Utilization	Can only store documents; cannot analyze them.	Uses LLMs to read unstructured text and generate insights across 100% of documented deals.
Root-Cause Categorization	No categorization capability.	Categorizes blockers into themes (e.g., discovery-stage failure, lack of solution alignment, internal delays).
Decision Support	Reactive—sales reps must decide actions manually.	Proactive—system recommends who to contact, what meeting to schedule, and which deals to close.
Opportunity Forecasting Quality	Based on stage + numeric fields only.	Incorporates semantic data: usage signals, sentiment in notes, stakeholder alignment, dependencies.
Impact on Revenue Ops	Helps track pipeline but does not improve it.	Drives pipeline hygiene, accelerates closures, and prevents multi-year stagnation.
Automation	Limited to pricing rules and quote generation.	Automates: Stuck deal detection Quote renewal analysis Policy enforcement Next-step recommendations
Business Value Generated	Operational reporting only.	Value creation through insights: • Saves RM/Sales time • Unblocks deals • Improves forecasting accuracy • Prevents pipeline rot

6. Discussion

The findings demonstrate that retrieval-augmented hybrid reasoning allows obtaining a good compromise between the rate of inference, configuration accuracy, and decision transparency. Lightweight LLM orchestration combined with formal optimization is associated with considerable computational savings, with no loss in solution validity [28]. In addition, the qualitative evaluation of domain experts was of the opinion that the explanations provided by the hybrid pipeline were more understandable and traceable in comparison with the explanations given by entirely generative LLM systems. This is in line with the existing literature that highlights explainability and trust as the critical issues in the adoption of enterprise AI [29]. By and large, the results support the idea that neural reasoning-based deterministic optimization hybrid architectures can be used to establish a feasible way forward to the real-time CPQ and enterprise decision systems to meet latency, compliance, and governance requirements simultaneously [30].

6.1. Challenges and Future Research Directions

Although the hybrid LLM-based CPQ frameworks have been performing positively, a number of challenges are yet to be overcome before large-scale industrial deployment can truly be achieved.

6.1.1. LLM Robustness and Constraint-Validation Accuracy

One key concern in the offered architecture is to make sure that the LLM reasoning layer will be robust and will be able to interact with the constraint-validation engine effortlessly. Although symbolic rule encoding and deterministic validation can assist in enforcing configuration and pricing feasibility, LLMs can still produce outputs that do not conform to domain constraints in case the input retrieved context is missing or the input is ambiguous. Strengthening the relationship between generative reasoning and constraint enforcement through structured prompt conditioning, rule-sensitive inference, and real-time feasibility auditing remains an important open research topic that is at the core of the consistency of hybrid CPQ systems.

6.1.2. Architectural Alignment Between Generative Reasoning and Optimization Modules

The other problem is the introduction of consistency in architecturally across the entire hybrid pipeline, both at retrieval and LLM reasoning, to constraint validation, optimisation as well as final quote generation. Any mismatch of these components can lead to inconsistency, e.g. solver-derived recommendations which are inconsistent with narrative output or solver-derived recommendations which fail to accomplish the purpose of price optimization. The next round of research needs to be on formal coordination mechanisms, inter-module standardized interfaces and multi-stage validation workflow that will guarantee the structure compatibility between work of generative outputs and downstream optimization logic.

6.1.3. Explainability, Auditability, and Governance Across the Hybrid Pipeline

The combination of probabilistic LLM reasoning with deterministic rule systems, which makes the architecture hybrid, presents major explanatory and governance problems. Business should have clear explanation of all the configuration and pricing suggestions, but the output of LLM can be hard to audit without further interpretation measures. Studies are required to come up with interpretable hybrid reasoning systems that offer causal explanations, evidence flow between retrieval and output, and governance systems that are able to comply with regulatory requirements. It is necessary to attain transparency and accountability throughout the RAG → LLM to Constraint Engine to Pricing Module channel to score trust and responsible adoption by the enterprise.

6.1.4. Privacy of Data, Secure Retrieval and Model Adaptation

Lastly, the architecture relies on delicate enterprise data which is continuously changed via explicit and implicit feedback pathways.Instead of retraining the entire model, changing the system to adapt to the new customer behaviors, pricing conditions, and product configurations creates privacy, security, and data drift challenges. The next generation of work needs to investigate privacy-saving retrieval procedures, federated/ role-based adapting tactics, and constant learning methods that sustain according to enterprise guidelines and at the same time secure the confidential data. The provision of safe, supportive, and dynamic performance within the Feedback → Retrieval → Generation loop is also one of the essential goals to pursue in the future.

7. Conclusion

This paper has discussed how Large Language Models have changed their purpose to be used in real-time CPQ optimization and decision intelligence in the enterprise. It was demonstrated that hybrid systems that combine retrieval, constraint reasoning, and probabilistic generative models can significantly contribute towards accuracy, interpretability, and overall quality of decisions in complex configuration and pricing environments. The results show that LLMs can act as smart coordinators in the process of optimization of enterprises, assisting in adaptive configuration, clear pricing, and more profound insight. The next direction of the research should be on responsible development and governance structures, cross-domain benchmarking, and more intensive connection of the LLMs with symbolic solvers, retrieval mechanisms, and explainable AI elements. As these capabilities continue to mature, LLMs are likely to shape the next generation of enterprise decision systems.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest to be disclosed.

References

[1] Junker, A., Felfernig, A., & Schmidt-Thieme, L. (2020). Knowledge-based systems for the Configure Price Quote (CPQ) process – A case study in the IT solution business. The Online Journal of Applied Knowledge Management, 8(2), 17–30. https://doi.org/10.36965/OJAKM.2020.8(2)17-30

[2] "A Review of Large Language Models: Fundamental Architectures, Key Techniques, Applications and Future Trends." (2024). Electronics, 13(24). https://doi.org/10.3390/electronics13245040

[3] "Integrating Large Language Models and Optimization in Semi-Structured Decision Problems." (2023). Algorithms, 17(12), 582. https://doi.org/10.3390/a17120582.

[4] Wasserkrug, S., Boussioux, L., den Hertog, D., Mirzazadeh, F., Birbil, Ş. I., Kurtz, J., & Maragno, D. (2025). Enhancing Decision Making Through the Integration of Large Language Models and Optimization Modeling. Proceedings of the AAAI Conference on Artificial Intelligence.

[5] Wu, X., Wu, S.-h., Wu, J., Feng, L., & Tan, K. C. (2024). Evolutionary Computation in the Era of Large Language Models: Survey and Roadmap. arXiv preprint.

[6] Lin, M., Sheng, J., Zhao, A., Wang, S., Yue, Y., Wu, Y., Liu, H., & Liu, J. (2024). LLM-based Optimization of Compound AI Systems: A Survey. arXiv preprint.

[7] Huang, W., Yang, K., Qi, S., & Wang, R. (2024). When large language model meets optimization. Swarm and Evolutionary Computation, 90, 101663. https://doi.org/10.1016/j.swevo.2024.101663

[8] Yu, H., & Liu, J. (2024). Deep insights into automated optimization with large language models and evolutionary algorithms. arXiv preprint arXiv:2410.20848. https://arxiv.org/abs/2410.20848.

[9] Handler, A., Larsen, K. R., & Hackathorn, R. (2024). Large language models present new questions for decision support. International Journal of Information Management, 79, 102811. https://doi.org/10.1016/j.ijinfomgt.2024.102811 ScienceDirect+1

[10] Kumar, P. (2024). Large language models (LLMs): survey, technical frameworks, and future challenges. Artificial Intelligence Review, 57, Article 260. https://doi.org/10.1007/s10462-024-10888-y SpringerLink

[11] Xu, Y., Hu, L., Zhao, J., Qiu, Z., Xu, K., Ye, Y., & Gu, H. (2025). A survey on multilingual large language models: Corpora, alignment, and bias. Frontiers of Computer Science, 19, 1911362. https://doi.org/10.1007/s11704-024-40579-4 SpringerLink+1

[12] Mustapha, K. B. (2025). A survey of emerging applications of large language models for problems in mechanics, product design, and manufacturing. Advanced Engineering Informatics, 64, 103066. https://doi.org/10.1016/j.aei.2024.103066 Wisc Library Search+2DBLP+2

[13] Mariotti, L., Guidetti, V., Mandreoli, F., Belli, A., & Lombardi, P. (2024). Combining large language models with enterprise knowledge graphs: A perspective on enhanced natural language understanding. Frontiers in Artificial Intelligence, 7, 1460065. https://doi.org/10.3389/frai.2024.1460065 Frontiers

[14] Régin, F., De Maria, E., & Bonlarron, A. (2024). Combining constraint programming reasoning with large language model predictions. In Proceedings of the 30th International Conference on Principles and Practice of Constraint Programming (CP 2024) (LIPIcs, Vol. 307, pp. 25:1–25:17). Schloss Dagstuhl. https://doi.org/10.4230/LIPIcs.CP.2024.25 DROPS

[15] Egger, A., Gerber, J., Hoffart, J., Kolk, P., Herzig, P., Decker, G., van der Aa, H., Polyvyanyy, A., Rinderle-Ma, S., Weber, I., & Weidlich, M. (2024). Large process models: A vision for business process management in the age of large language models. KI – Künstliche Intelligenz, 38(4), 407–418. https://doi.org/10.1007/s13218-024-00863-8SpringerLink

[16] Said, A. (2024). On explaining recommendations with Large Language Models: A review. Frontiers in Big Data, 7, 1505284. https://doi.org/10.3389/fdata.2024.1505284 Frontiers

[17] Liu S, Huang SS, McCoy AB, et al. Optimizing Order Sets With a Large Language Model–Powered Multiagent System. JAMA Network Open. 2025;8(9):e2533277. doi:10.1001/jamanetworkopen.2025.33277

[18] Göpfert, J., Weinand, J. M., Kuckertz, P., & Stolten, D. (2024). Opportunities for large language models and discourse in engineering design. Energy and AI, 17, 100383. https://doi.org/10.1016/j.egyai.2024.100383.

[19] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-T., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. Advances in Neural Information Processing Systems, 33, 9459–9471. NeurIPS Proceedings+1

[20] Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. Advances in Neural Information Processing Systems, 30, 4765–4774. DBLP+1

[21] Soininen, T., Tiihonen, J., Männistö, T., & Sulonen, R. (1998). Towards a general ontology of configuration. Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 12(4), 357–372. https://doi.org/10.1017/S0890060498124083 Cambridge University Press & Assessment+1

[22] Falkner, A. A., Haselböck, A., Schenner, G., & Schreiner, H. (2011). Modeling and solving technical product configuration problems. Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 25(2), 115–129. Cambridge University Press & Assessment+1

[23] Chenavaz, R. Y. (2012). Dynamic pricing, product and process innovation. European Journal of Operational Research, 222(3), 553–557. https://doi.org/10.1016/j.ejor.2012.05.016

[24] Kopalle, P. K., Pauwels, K., Akella, L. Y., & Gangwar, M. (2024). Dynamic pricing: Definition, implications for managers, and future research directions. Journal of Retailing, 99(4), 575–595.

[25] Zhang, R., Sun, F., & Zhou, D. (2024). Low-latency inference optimization for large-scale language models. IEEE Transactions on Neural Networks and Learning Systems, 35(8), 9541–9555.

[26] Fang, L., Zhao, H., & Lin, J. (2024). Hybrid reasoning with large language models and constraint solvers for combinatorial optimization. Expert Systems with Applications, 237, 121423.

[27] Li, J., Gao, X., & Tang, P. (2024). Production architectures for hybrid reasoning AI systems. ACM Transactions on Intelligent Systems and Technology, 15(3), 51:1–51:19.

[28] Qin, L., Xu, H., & Yuan, M. (2024). Retrieval-augmented decision frameworks for industrial AI systems. IEEE Transactions on Industrial Informatics, 20(4), 5122–5134.

[29] Camburu, O. M., et al. (2023). Towards trustworthy explainable AI for enterprise decision systems. Knowledge-Based Systems, 273, 110695.

[30] Singh, V., Kumar, R., & Chatterjee, S. (2024). Scaling AI configuration systems for large-enterprise operations. Journal of Industrial Information Integration, 30, 100490.