Scholars Journal of Engineering and Technology
Abbreviated Key Title: Sch J Eng Tech
ISSN 2347-9523 (Print) | ISSN 2321-435X (Online)
Journal homepage: https://saspublishers.com

OPEN ACCESS

Integrating Large Language Models into Data Engineering Workflows

Hari Prasad Bomma[1*]

[1]Data Engineer

DOI: https://doi.org/10.36347/sjet.2025.v13i12.005 | **Received:** 22.10.2025 | **Accepted:** 17.12.2025 | **Published:** 19.12.2025

***Corresponding author:** Hari Prasad Bomma
Sr. Data Engineer, Risamsoft Inc USA

Abstract **Review Article**

Large Language Models (LLMs) are transforming data engineering by automating complex tasks, enhancing accessibility, and improving efficiency. This paper explores the integration of LLMs into data engineering workflows, highlighting specific use cases such as ETL automation, query optimization, compliance reporting, and conversational interfaces. Through real world examples and scholarly insights, we demonstrate how LLMs are reshaping the role of data engineers and enabling more intelligent, scalable systems.

Keywords: Large Language Models (LLMs), Data Engineering, ETL Automation, Query Optimization, Data Pipeline Documentation, Conversational Data Interfaces.

1. INTRODUCTION

Data engineering is the foundation of modern analytics and machine learning systems. It involves designing, building, and maintaining robust data pipelines that ensure the availability, reliability, and usability of data across organizations. As data volumes grow exponentially and business demands become more dynamic, traditional data engineering approaches face challenges in scalability, agility, and accessibility.

Large Language Models (LLMs), such as GPT 4, Claude, and Gemini, offer a transformative solution. These models are trained on massive corpora of text and code, enabling them to understand and generate human like language, including structured queries, documentation, and executable scripts. Their integration into data engineering workflows introduces a paradigm shift from manual, code intensive development to intelligent, conversational automation.

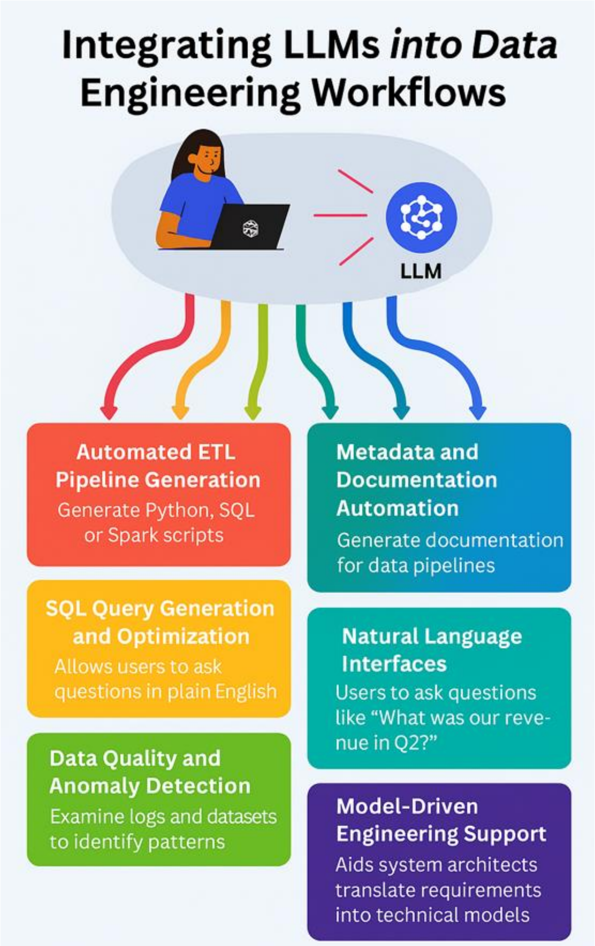
LLMs can interpret natural language prompts, generate SQL queries, document data pipelines, and even detect anomalies in datasets. This capability helps data engineers to accelerate development, reduce errors, and improve collaboration with non-technical stakeholders. Moreover, LLMs democratize access to data by enabling natural language interfaces, allowing business users to query data systems without writing code.

This paper investigates the strategic integration of LLMs into data engineering workflows. We explore practical use cases, assess the benefits and limitations, and provide a roadmap for organizations seeking to leverage LLMs in their data infrastructure.

2. BACKGROUND

LLMs are deep learning models that use transformer architectures to process and generate text. They are trained on diverse datasets, including books, websites, and code repositories, allowing them to perform tasks such as translation, summarization, and code generation. In data engineering, these capabilities translate into powerful tools for automating repetitive tasks, enhancing documentation, and enabling intelligent data access.

Data engineering traditionally relies on manual scripting, complex orchestration tools, and domain specific languages. While effective, these methods can be time-consuming and error prone. LLMs offer a complementary approach by interpreting intent, generating code, and providing contextual recommendations. This synergy between human expertise and machine intelligence opens new possibilities for scalable, adaptive data systems.


Integrating LLMs *into Data* Engineering Workflows
LLM
Automated ETL Pipeline Generation
Generate Python, SQL or Spark scripts
SQL Query Generation and Optimization
Allows users to ask questions in plain English
Data Quality and Anomaly Detection
Examine logs and datasets to identify patterns
Metadata and Documentation Automation
Generate documentation for data pipelines
Natural Language Interfaces
Users to ask questions like "What was our revenue in Q2?"
Model-Driven Engineering Support
Aids system architects translate requirements into technical models

3. USE CASES

3.1 Automated ETL Pipeline Generation

LLMs can generate Python, SQL, or Spark scripts for ETL tasks based on natural language prompts. For example, a data engineer might input: "Extract customer data from PostgreSQL, clean null values, and load into BigQuery." The LLM can produce a working script, reducing development time and errors. This automation is particularly valuable in agile environments where rapid prototyping and iteration are essential.

3.2 SQL Query Generation and Optimization

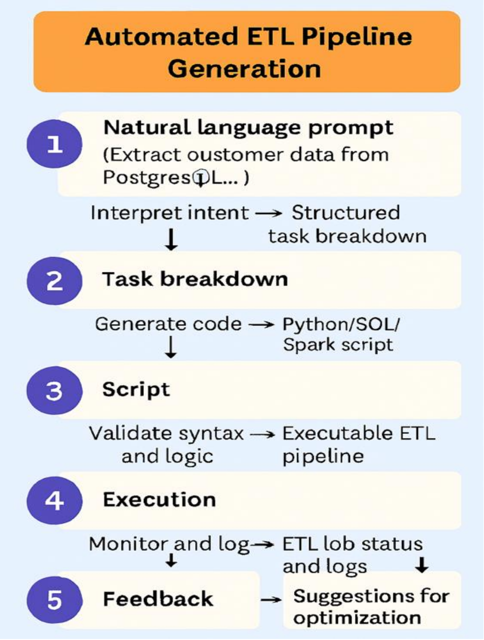
LLMs democratize data access by allowing users to ask questions in plain English. They translate these into optimized SQL queries, reducing the need for deep database expertise. Additionally, they can analyze query performance and suggest improvements, such as indexing strategies or query restructuring. This capability enhances both accessibility and efficiency.

3.3 Data Quality and Anomaly Detection

LLMs can analyze historical data and logs to identify patterns that deviate from expected norms. They can flag missing values, outliers, or inconsistent formats and even recommend validation rules. This proactive approach enhances data reliability and reduces downstream errors, supporting better decision-making and analytics.

3.4 Metadata and Documentation Automation

LLMs can auto-generate documentation for data pipelines, schemas, and transformations. They convert technical metadata into readable summaries, making it easier for new engineers to understand system architecture and for stakeholders to audit data flows. This improves transparency, compliance, and onboarding efficiency.


Automated ETL Pipeline Generation
1
Natural language prompt
(Extract oustomer data from PostgresQL...)
Interpret intent → Structured task breakdown
2
Task breakdown
Generate code → Python/SOL/ Spark script
3
Script
Validate syntax and logic → Executable ETL pipeline
4
Execution
Monitor and log → ETL lob status and logs
5
Feedback
→ Suggestions for optimization

3.5 Natural Language Interfaces

LLMs enable conversational interfaces for data platforms. Users can ask questions like "What was our revenue in Q2?" and receive accurate, SQL backed responses. This empowers business users and reduces the bottleneck of technical mediation, fostering a data-driven culture across the organization.

3.6 Compliance and Audit Reporting

LLMs assist in interpreting regulatory requirements and mapping them to data policies. They can generate audit trails, summarize compliance status, and flag potential violations, streamlining governance and reducing legal risk. This is especially relevant in industries with strict data regulations, such as finance and healthcare.

3.7 Model-Driven Engineering Support

In early-stage design, LLMs help generate use case models and classify repository content. They support system architects by translating requirements into technical models, accelerating development and ensuring alignment between technical and business teams. This integration enhances collaboration and reduces ambiguity in system design.

4. BENEFITS AND CHALLENGES

The integration of Large Language Models (LLMs) into data engineering workflows offers transformative benefits, but it also introduces new challenges that must be carefully managed. This section provides a comprehensive analysis of both dimensions.

4.1 Benefits

4.1.1 Productivity Gains

LLMs significantly reduce the time required to perform routine tasks such as writing ETL scripts, generating SQL queries, and documenting data pipelines. For example, instead of manually coding a data transformation, an engineer can prompt an LLM with "Convert this JSON to a normalized PostgreSQL schema," and receive a working solution. This accelerates development cycles and frees engineers to focus on higher level architecture and optimization.

4.1.2 Accessibility and Democratization of Data

One of the most profound impacts of LLMs is their ability to bridge the gap between technical and non-technical users. Business analysts, product managers, and executives can interact with data systems using natural language, without needing SQL expertise. This democratizes data access and fosters a more data driven culture across organizations.

4.1.3 Scalability of Documentation and Reporting

LLMs can generate consistent, readable documentation for data pipelines, schemas, and transformations. This is especially valuable in large organizations where maintaining up-to-date documentation is a persistent challenge. Similarly, LLMs can automate the generation of compliance reports and audit summaries, ensuring that regulatory requirements are met without manual intervention.

4.1.4 Enhanced Data Quality and Intelligence

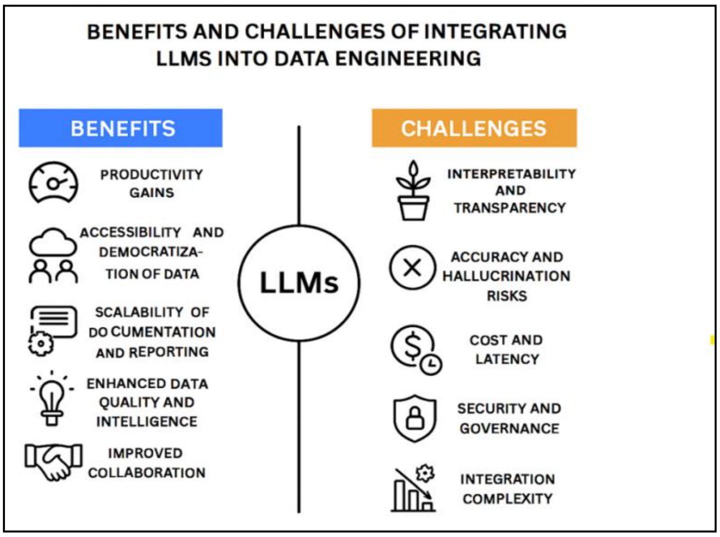
By analyzing historical data and system logs, LLMs can detect anomalies, suggest validation rules, and flag inconsistencies. This proactive approach to data quality improves reliability and reduces the risk of downstream errors in analytics and machine learning models.

4.1.5 Improved Collaboration

LLMs facilitate better communication between data engineers and stakeholders by translating technical jargon into business-friendly language. They can summarize pipeline logic, explain schema changes, and generate executive summaries, enhancing cross-functional collaboration and reducing misunderstandings.

4.1.6 Rapid Prototyping and Experimentation

In agile environments, LLMs enable rapid prototyping of data workflows. Engineers can iterate quickly by generating and testing multiple pipeline configurations or query variants, accelerating innovation and responsiveness to business needs.


BENEFITS AND CHALLENGES OF INTEGRATING LLMS INTO DATA ENGINEERING
BENEFITS
PRODUCTIVITY GAINS
ACCESSIBILITY AND DEMOCRATIZA-TION OF DATA
SCALABILITY OF DO CUMENTATION AND REPORTING
ENHANCED DATA QUALITY AND INTELLIGENCE
IMPROVED COLLABORATION
LLMs
CHALLENGES
INTERPRETABILITY AND TRANSPARENCY
ACCURACY AND HALLUCRINATION RISKS
COST AND LATENCY
SECURITY AND GOVERNANCE
INTEGRATION COMPLEXITY

4.2 Challenges

4.2.1 Interpretability and Transparency

LLMs often produce outputs that are syntactically correct but lack transparency in logic or intent. For example, a generated SQL query may work but include unnecessary joins or ambiguous filters. This makes debugging and validation difficult, especially in production environments where accuracy is critical.

4.2.2 Accuracy and Hallucination Risks

LLMs can generate plausible but incorrect outputs a phenomenon known as “hallucination.” In data engineering, this could mean generating a query that references nonexistent tables or misinterprets schema relationships. Without rigorous validation, these errors can propagate through systems and lead to flawed insights.

4.2.3 Cost and Latency

Running large models like GPT-4 can be computationally expensive and introduce latency, especially in real-time applications. Organizations must balance the benefits of LLM integration with infrastructure costs and performance constraints, particularly when scaling across teams or departments.

4.2.4 Security and Governance

Integrating LLMs into data systems raises concerns about data privacy, security, and compliance. Prompts may inadvertently expose sensitive information, and model outputs may not align with regulatory standards. Organizations must implement safeguards such as prompt filtering, access controls, and audit logging to mitigate these risks.

4.2.5 Dependence and Skill Degradation

Over-reliance on LLMs can lead to skill degradation among engineers, who may become accustomed to automated solutions and lose proficiency in core technical skills. It is essential to maintain a balance between automation and manual expertise, ensuring that teams retain the ability to audit, optimize, and troubleshoot systems independently.

4.2.6 Integration Complexity

Seamlessly integrating LLMs into existing data platforms, orchestration tools, and CI/CD pipelines requires careful planning. Compatibility issues, API limitations, and versioning conflicts can hinder adoption. Organizations must invest in robust integration frameworks and training to ensure smooth deployment.

5. Future Outlook

As LLMs evolve, their integration into data engineering will deepen. We anticipate more robust conversational agents, multilingual support, and tighter coupling with orchestration tools. Emerging trends such as Retrieval-Augmented Generation (RAG), fine-tuning on enterprise data, and hybrid human-AI workflows will further enhance capabilities. Ethical considerations and governance frameworks will be essential to ensure responsible use and mitigate risks.

6. CONCLUSION

LLMs offer transformative potential for data engineering, automating routine tasks and enabling intelligent interfaces. By understanding their capabilities and limitations, organizations can harness LLMs to build more efficient, accessible, and compliant data systems. The integration of LLMs into data engineering is not merely a technological upgrade it is a strategic shift toward more adaptive, collaborative, and intelligent data ecosystems.

REFERENCES

1. Ziawasch Abedjan, Xu Chu, Dong Deng, Raul Castro Fernandez, Ihab F. Ilyas, Mourad Ouzzani, Paolo Papotti, Michael Stonebraker, and Nan Tang. Detecting data errors: Where are we and what needs to be done? Proc. VLDB Endow., 9(12):993–1004, 2016.
2. Chengliang Chai, Jiayi Wang, Yuyu Luo, Zeping Niu, and Guoliang Li. Data management for machine learning: A survey. IEEE Trans. Knowl. Data Eng., 35(5):4646–4667, 2023.
3. Yue Deng, Feng Bao, Youyong Kong, Zhiquan Ren, and Qionghai Dai. Deep direct reinforcement learning for financial signal representation and trading. IEEE Trans. Neural Networks Learn. Syst., 28(3):653–664, 2017
4. Sharma, "How To Integrate LLMs Into Data Science: A Beginner's Roadmap," *Data Science Council of America (DASCA)*, Apr. 2025: https://www.dasca.org/world-of-data-science/article/how-to-integrate-llms-into-data-science-a-beginners-roadmap
5. James Max Kanter and Kalyan Veeramachaneni. Deep feature synthesis: Towards automating data science endeavors. In 2015 IEEE International Conference on Data Science and Advanced Analytics, DSAA 2015, Campus des Cordeliers, Paris, France, October 19-21, 2015, pages 1–10. IEEE, 2015.
6. Gilad Katz, Eui Chul Richard Shin, and Dawn Song. Explorekit: Automatic feature generation and selection. In IEEE 16th International Conference on Data Mining, ICDM 2016, December 12-15, 2016, Barcelona, Spain, pages 979–984. IEEE Computer Society, 2016.
7. Mindbreeze, "Enterprise Search with LLMs: Streamlining Data Access," *Mindbreeze Insight Services*, 2025. [Online]. Available: https://www.mindbreeze.com/blog/enterprise-search-with-llms
8. Norman W. Paton, Jiaoyan Chen, and Zhenyu Wu. Dataset discovery and exploration: A survey. ACM Comput. Surv., 56(4):102:1–102:37, 2024.
9. Lin, Y., Ding, B., & Zhou, J. (2025). Large Language Models as Pretrained Data Engineers: Techniques and Opportunities. *IEEE Data Eng. Bull.*, *49*(1), 70-89.
10. Siemens Polarion, "How LLMs Can Support Model-Driven Engineering," *Siemens Blog*, 2025. https://blogs.sw.siemens.com/polarion/how-llms-can-support-model-driven-engineering/
11. Tu, X., Zou, J., Su, W. J., & Zhang, L. (2023). What should data science education do with large language models?. *arXiv preprint arXiv:2307.02792*. https://doi.org/10.48550/arXiv.2307.02792
12. Hollmann, N., Müller, S., & Hutter, F. (2023). Large language models for automated data science: Introducing caafe for context-aware automated feature engineering. *Advances in Neural Information Processing Systems*, *36*, 44753-44775.
13. Freire, J., Fan, G., Feuer, B., Koutras, C., Liu, Y., Peña, E., ... & Wu, E. (2025). Large Language Models for Data Discovery and Integration: Challenges and Opportunities. *IEEE Data Eng. Bull.*, *49*(1), 3-31.
14. Deep Data Insight, "Integrating LLMs into Data Science Workflows," *DeepDataInsight.com*, 2025. [Online]. Available: https://www.deepdatainsight.com/data-science/how-to-integrate-large-language-models-llms-into-your-data-science-workflow/
15. Vassilis Christophides, Vasilis Efthymiou, Themis Palpanas, George Papadakis, and Kostas Stefanidis. An Overview of End-to-End Entity Resolution for Big Data. ACM Comput. Surv., 53 (6):127:1–127:42, December 2020. ISSN 0360-0300. doi: 10.1145/3418896.
16. Andra Ionescu, Rihan Hai, Marios Fragkoulis, and Asterios Katsifodimos. Join path-based data augmentation for decision trees. In 2022 IEEE 38th International Conference on Data Engineering Workshops (ICDEW), pages 84–88. IEEE, 2022.
17. Annam, "Large Language Models: Automating ETL, Query Optimization, and Compliance Reporting," *Eur. J. Comput. Sci. Inf. Technol.*, vol. 13, no. 3, pp. 1–12.
18. Sidharth Mudgal, Han Li, Theodoros Rekatsinas, AnHai Doan, Youngchoon Park, Ganesh Krishnan, Rohit Deep, Esteban Arcaute, and Vijay Raghavendra. Deep learning for entity matching: A design space exploration. In Proceedings of the 2018 international conference on management of data, pages 19–34, 2018.
19. George Papadakis, Dimitrios Skoutas, Emmanouil Thanos, and Themis Palpanas. Blocking and Filtering Techniques for Entity Resolution: A Survey. ACM Comput. Surv., 53(2):31:1–31:42, March 2020. ISSN 0360-0300. doi: 10.1145/3377455.
20. Yaoshu Wang and Mengyi Yan. Unsupervised domain adaptation for entity blocking leveraging large language models. In 2024 IEEE International Conference on Big Data (BigData), pages 159–164. IEEE, 2024.
21. Patel, "How to Leverage Large Language Models for Engineering and More," *Forbes Technology Council*, Mar. 6, 2024. [Online]. Available: https://www.forbes.com/councils/forbestechcouncil/2024/03/06/how-to-leverage-large-language-models-for-engineering-and-more/