**International Journal of Artificial Intelligence, Data Science, and Machine Learning**
Grace Horizon Publication | Volume 5, Issue 2, 9-16, 2024
ISSN: 3050-9262 | https://doi.org/10.63282/30509262/IJAIDSML-V5I2P102

*Original Article*

# Automating Data Engineering Workflows with AI and Machine Learning

Aditya Banerjee
Computer Vision Engineer, Zensar Technologies, India

***Abstract -*** *Data engineering is a critical component of modern data-driven organizations, encompassing the extraction, transformation, and loading (ETL) of data, as well as the management and optimization of data pipelines. The increasing volume, velocity, and variety of data pose significant challenges for data engineers, who must ensure that data is accurate, timely, and available for various downstream applications. This paper explores the integration of artificial intelligence (AI) and machine learning (ML) techniques to automate and optimize data engineering workflows. We discuss the current state of data engineering, the challenges faced by data engineers, and the potential benefits of AI and ML in addressing these challenges. We present several case studies and algorithms that demonstrate the effectiveness of AI and ML in automating data engineering tasks, including data quality assessment, schema inference, and pipeline optimization. Finally, we discuss the ethical and practical considerations of deploying AI in data engineering and provide recommendations for future research and development.*

***Keywords -*** *AI in Data Engineering, Machine Learning, Data Quality, Data Pipelines, Automation, Big Data, Schema Inference, Pipeline Optimization, Ethical AI, Real-Time Processing*

## 1. Introduction

Data engineering is a fundamental discipline that plays a critical role in enabling organizations to harness the power of data and extract meaningful value from it. This field encompasses a broad spectrum of tasks, including data ingestion, transformation, storage, and retrieval, all of which are essential for building robust data infrastructures. As the volume of data continues to grow at an exponential rate, the complexity of data engineering workflows has escalated, making it increasingly challenging for organizations to manage and process data efficiently. This complexity has led to a significant demand for automation and optimization solutions that can help streamline these processes and reduce the burden on data engineers.

Traditional data engineering practices often rely on manual, rule-based approaches, which can be highly time-consuming and prone to human error. These methods also struggle to scale effectively as data volumes and varieties increase, often resulting in delays and inefficiencies that can hamper an organization's ability to make timely, data-driven decisions. In response to these challenges, the integration of Artificial Intelligence (AI) and Machine Learning (ML) technologies has emerged as a promising solution. AI and ML can provide automated, intelligent systems that are capable of adapting to dynamic data environments and optimizing workflows in real-time.

These intelligent systems can automate repetitive tasks, such as data cleaning and preprocessing, and can also handle more complex operations, such as data schema evolution and anomaly detection. By leveraging AI and ML, data engineers can focus on higher-value tasks, such as designing data architectures and developing data strategies, while the automated systems handle the more routine aspects of data management. Moreover, AI and ML algorithms can continuously learn from data patterns and user interactions, improving their performance over time and ensuring that the data engineering processes remain efficient and effective even as data environments evolve.

In addition to automation, AI and ML can also enhance data quality and reliability. For example, machine learning models can be trained to identify and correct inconsistencies in data, ensuring that the data used for analysis and decision-making is accurate and trustworthy. Furthermore, these technologies can help in the discovery of new insights and patterns that might not be apparent through traditional methods, thereby unlocking new opportunities for innovation and growth.

As organizations increasingly recognize the importance of data in driving business outcomes, the adoption of AI and ML in data engineering is likely to become more widespread. This shift not only addresses the current challenges but also positions organizations to better leverage the vast amounts of data they collect, ultimately leading to more informed, agile, and competitive business practices.

## 2. Data Engineering: An Overview

### *2.1 Definition and Scope*

Data engineering is a critical discipline that involves designing, constructing, and maintaining the infrastructure necessary for efficient data collection, storage, processing, and analysis. It serves as the backbone of modern data-driven applications, ensuring that raw data is transformed into a structured, usable format for business intelligence, machine learning, and other analytical processes. The scope of data engineering spans multiple functions, beginning with data ingestion, where data is collected from various sources such as databases, APIs, and IoT devices. This data is then subjected to data transformation, a crucial step that involves cleaning, normalizing, and aggregating raw inputs to make them suitable for analysis. Once transformed, data must be stored in a way that guarantees accessibility, reliability, and security, making data storage a fundamental aspect of data engineering. Additionally, data retrieval ensures that stored data can be accessed efficiently for reporting, analytics, or decision-making. To manage these processes seamlessly, data pipeline management is employed, overseeing the continuous flow of data across different stages from ingestion to transformation and ultimately to analysis.

### *2.2 Challenges in Data Engineering*

Despite its significance, data engineering faces several challenges that make designing and maintaining data pipelines a complex task. One of the biggest hurdles is data volume, as modern applications generate massive amounts of data that can easily overwhelm traditional processing systems. Alongside volume, data velocity presents another challenge, with real-time data streams requiring rapid ingestion and processing to support applications such as financial transactions, recommendation systems, and IoT monitoring. The complexity is further exacerbated by data variety, as information is often available in multiple formats, including structured, semi-structured, and unstructured data, making it difficult to create a unified processing pipeline. Another major concern is data quality, which involves ensuring accuracy, consistency, and completeness. Poor-quality data can lead to inaccurate insights and unreliable models, making rigorous validation and cleansing essential. Moreover, scalability is a pressing issue, as data infrastructure must handle growing workloads without performance degradation. Lastly, cost remains a significant factor, as maintaining robust data engineering solutions especially for large-scale organizations requires substantial investment in computing resources, storage solutions, and skilled personnel.

### *2.3 Traditional Approaches*

Historically, data engineering has relied on manual, rule-based methodologies to manage and process data. One of the most common techniques is the Extract, Transform, Load (ETL) process, which systematically extracts data from source systems, applies transformations to structure and cleanse it, and loads it into a target system such as a data warehouse. While ETL processes have been widely adopted, they are often manually implemented, making them time-consuming and susceptible to human error. Another widely used traditional approach is data warehousing, where large volumes of data are centralized and managed for analytical purposes. While data warehouses are instrumental in business intelligence and reporting, their design and maintenance require significant resources, and they may not be well-suited for handling unstructured or real-time data. Additionally, batch processing has been a prevalent method for handling bulk data operations, where large datasets are processed at scheduled intervals. However, batch processing has limitations, particularly in scenarios where real-time analytics and instantaneous data processing are required. As data ecosystems evolve, these traditional approaches are gradually being replaced or supplemented by modern, automated, and AI-driven solutions that offer greater efficiency, scalability, and real-time capabilities.

**Table 1: Comparison of Traditional and AI-Driven Data Engineering**

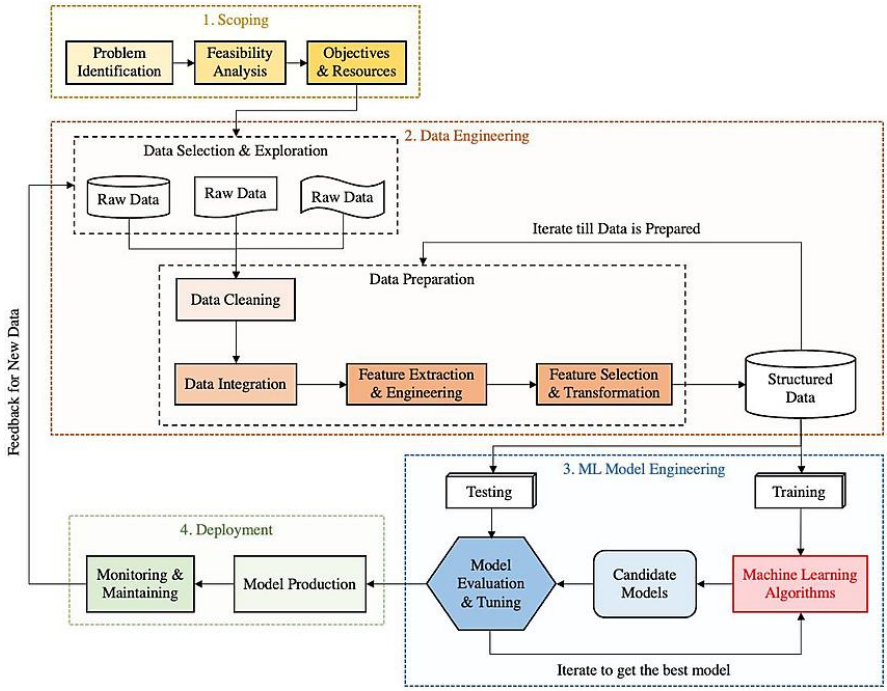
| Aspect | Traditional Data Engineering | AI-Driven Data Engineering |
|---|---|---|
| Automation | Manual, rule-based | Automated, intelligent |
| Scalability | Limited | High |
| Adaptability | Low | High |
| Cost | High | Variable |
| Data Quality | Manual inspection | Automated assessment |
| Schema Inference | Manual modeling | Automated inference |
| Pipeline Optimization | Manual tuning | Dynamic optimization |

### *2.4. Data Engineering Workflow*

AI-driven data engineering, encompassing four key stages: Scoping, Data Engineering, ML Model Engineering, and Deployment. The first phase, Scoping, involves identifying the problem, conducting feasibility analysis, and defining objectives and available resources. This step ensures that the data pipeline aligns with the business or research goals.

The second phase, Data Engineering, focuses on the transformation of raw data into structured data. It includes data selection and exploration, where multiple raw data sources are considered. This data then undergoes cleaning, integration, and feature extraction, ensuring its quality before being processed into structured formats. An iterative feedback loop is in place to refine data preparation until it reaches the desired quality for machine learning models.

Following data engineering, the ML Model Engineering phase begins, where structured data is used for training ML models. This phase includes model evaluation and tuning, where different candidate models are tested and optimized using




machine learning algorithms. The goal is to refine the model iteratively until the best-performing one is selected for deployment. Finally, in the Deployment phase, the selected model is integrated into a production environment. The model is continuously monitored and maintained to ensure optimal performance. Feedback loops allow for retraining and updating the model when new data becomes available, ensuring long-term effectiveness and adaptability. This workflow highlights how AI and automation enhance the efficiency of data engineering and ML model development, making the entire process more scalable and intelligent.

**Fig 1: Automated Data Engineering and ML Pipeline**

## 3. The Role of AI and ML in Data Engineering

### *3.1 Overview of AI and ML*

Artificial Intelligence (AI) and Machine Learning (ML) are transformative branches of computer science that focus on developing systems capable of performing tasks that traditionally require human intelligence. AI encompasses a broad spectrum of techniques, including perception, reasoning, and decision-making, enabling machines to solve complex problems autonomously. ML, a subset of AI, takes this a step further by enabling systems to learn from data, improving their performance over time without explicit programming. By leveraging historical data patterns, ML models can make accurate predictions, detect anomalies, and automate decision-making processes. In the context of data engineering, AI and ML offer the potential to streamline workflows, automate repetitive tasks, and enhance data quality, making them indispensable tools for modern data-driven enterprises.

### *3.2 Potential Benefits of AI and ML in Data Engineering*

The integration of AI and ML into data engineering presents numerous advantages that significantly enhance the efficiency and effectiveness of data management processes. One of the most notable benefits is automation, where AI-driven tools can handle labor-intensive tasks such as data cleaning, transformation, and integration, allowing data engineers to focus on higher-level strategic activities. Another key advantage is intelligence, as AI can provide valuable insights and recommendations, assisting data engineers in optimizing data pipelines and improving data-driven decision-making. Scalability is another critical factor, as AI and ML models can efficiently process and analyze vast amounts of data without compromising performance. This ensures that data engineering workflows remain robust and adaptable to the growing demands of big data environments. Additionally, adaptability plays a crucial role, as AI-powered solutions can dynamically adjust to evolving data patterns, schema changes, and infrastructure upgrades, enhancing the resilience and flexibility of data processing pipelines.

### *3.3 Key Applications of AI and ML in Data Engineering*

AI and ML technologies can be applied across various facets of data engineering, improving the accuracy, speed, and reliability of data workflows. One of the primary applications is data quality assessment, where AI models automatically identify and rectify errors, inconsistencies, and missing values, ensuring the reliability of datasets used for analytics and machine learning models. Another significant use case is schema inference, where AI can analyze and infer the structure of data sources, reducing the need for manual schema design and improving data integration efficiency. Additionally, AI-powered pipeline optimization enables the automation of data ingestion, transformation, and processing, ensuring that data flows seamlessly through complex workflows while minimizing latency. AI and ML also play a vital role in anomaly detection, helping data engineers identify and resolve unexpected deviations in data that could impact downstream applications. Lastly, predictive maintenance is an emerging AI-driven approach that anticipates potential failures in data engineering infrastructure, allowing proactive interventions to minimize downtime and maintain seamless operations. These applications demonstrate how AI and ML are revolutionizing data engineering, making it more intelligent, efficient, and resilient to ever-changing data landscapes.

## 4. Case Studies and Algorithms

### *4.1 Case Study 1: Automated Data Quality Assessment*

Data quality is a fundamental aspect of data engineering, as poor-quality data can result in inaccurate insights, flawed decision-making, and wasted resources. Traditional methods of data quality assessment rely heavily on manual inspection and rule-based approaches, which are often time-consuming, inconsistent, and prone to human error. These challenges necessitate the adoption of an AI-driven approach to automate and improve the accuracy and efficiency of data quality assessments.

To address these issues, an AI-based data quality assessment framework can be implemented, leveraging machine learning models to identify and correct data anomalies automatically. The first step in this approach is data profiling, which involves analyzing the dataset to determine key characteristics such as data types, distributions, and relationships. Following this, feature engineering is performed to extract meaningful attributes that help in training the machine learning model. The model training phase involves using labeled datasets to train the algorithm to detect common data quality issues such as missing values, outliers, and inconsistencies. Once trained, the model is deployed into the data pipeline to evaluate incoming data in real time. If discrepancies are identified, the system can apply automatic corrections, such as imputing missing values or rectifying inconsistencies, ensuring that the data remains clean and reliable for downstream analysis. This AI-powered solution significantly reduces manual intervention, enhances data reliability, and optimizes decision-making processes.

The algorithm for automated data quality assessment follows a structured approach where raw data undergoes profiling, feature extraction, and training before being deployed in production. The trained model evaluates incoming data for errors and applies corrective measures, ensuring consistent data quality across the pipeline. This intelligent system not only minimizes human errors but also improves the efficiency of large-scale data engineering workflows.

#### *4.1.1. Algorithm*

**Algorithm: Automated Data Quality Assessment**

1. **Input:** Raw data, labeled data for training
2. **Output:** Cleaned data, quality report

```
def data_quality_assessment(raw_data, labeled_data):
    # Step 1: Data Profiling
    data_profile = profile_data(raw_data)

    # Step 2: Feature Engineering
    features = extract_features(data_profile)

    # Step 3: Model Training
    model = train_model(labeled_data, features)

    # Step 4: Model Deployment
    quality_report = model.predict(raw_data)

    # Step 5: Automatic Correction
```

```
    cleaned_data = apply_corrections(raw_data, quality_report)

    return cleaned_data, quality_report
```

### *4.2 Case Study 2: Schema Inference*

Schema inference plays a crucial role in data engineering by determining the structure of a dataset, including column data types, relationships, and dependencies. In traditional data management, schema inference is often performed manually, which is a labor-intensive process, especially when dealing with large and complex datasets. The need for an automated, intelligent approach has led to the development of AI-driven schema inference techniques that streamline this process.

In an AI-based schema inference system, the first step is data sampling, where a representative subset of the dataset is selected to analyze its structure. This sample is then used for feature engineering, extracting characteristics such as data types, value distributions, and potential relationships between attributes. A machine learning model is trained using this feature set to predict the schema of the dataset. Once trained, the model is deployed into a data pipeline, where it continuously analyzes incoming datasets and infers schemas automatically. This eliminates the need for manual schema definition, making data integration faster and more efficient.

The AI-driven schema inference system significantly reduces human effort and errors in schema design, enabling faster integration of diverse data sources. By leveraging ML algorithms, organizations can ensure that their data is correctly structured, facilitating seamless data processing and analytics. This automated approach is particularly useful for organizations handling large-scale data ingestion from multiple sources, where schema variations can be a frequent challenge.

#### *4.2.1. Algorithm*

**Algorithm: Automated Schema Inference**

1. **Input:** Sample data
2. **Output:** Inferred schema

```
def schema_inference(sample_data):
    # Step 1: Data Sampling
    sample = select_sample(sample_data)

    # Step 2: Feature Engineering
    features = extract_features(sample)

    # Step 3: Model Training
    model = train_model(sample, features)

    # Step 4: Model Deployment
    inferred_schema = model.predict(sample_data)

    return inferred_schema
```

### *4.3. Case Study 3: Pipeline Optimization*

Data pipelines are the backbone of data engineering workflows, responsible for ingesting, transforming, and storing data efficiently. However, as data volumes grow and pipeline complexity increases, maintaining optimal performance becomes a significant challenge. Poorly optimized pipelines can lead to processing bottlenecks, high latency, and increased resource consumption. Traditional pipeline optimization approaches involve static rule-based configurations, which may not adapt well to dynamic workloads. To overcome these limitations, an AI-driven reinforcement learning approach can be used for pipeline optimization.

The optimization process begins with pipeline monitoring, where various performance metrics such as processing time, resource utilization, and error rates are collected. The next step is feature engineering, where relevant attributes are extracted from these performance metrics, such as the number of tasks executed, data size, and transformation complexity. A reinforcement learning model is then trained to dynamically optimize the pipeline configuration based on performance data. The trained model is deployed within the data pipeline, where it continuously adjusts configurations in real time, ensuring optimal performance based on changing data loads and processing requirements.

By implementing AI-driven pipeline optimization, organizations can ensure that their data workflows remain efficient, responsive, and cost-effective. This approach enables dynamic adaptation to workload variations, minimizes processing delays, and optimizes resource allocation. As a result, businesses can achieve faster data processing, reduce operational costs, and enhance the overall reliability of their data engineering infrastructure.

*4.3.1. Algorithm*

**Algorithm: Pipeline Optimization**

1. **Input:** Pipeline performance metrics, desired outcomes
2. **Output:** Optimized pipeline configuration

```
def pipeline_optimization(performance_metrics, desired_outcomes):
    # Step 1: Pipeline Monitoring
    metrics = collect_metrics(performance_metrics)

    # Step 2: Feature Engineering
    features = extract_features(metrics)

    # Step 3: Model Training
    model = train_reinforcement_learning_model(features, desired_outcomes)

    # Step 4: Model Deployment
    optimized_config = model.optimize(metrics)

    return optimized_config
```

## 5. Ethical and Practical Considerations

### *5.1 Ethical Considerations*

The integration of AI and ML into data engineering introduces various ethical concerns that must be carefully addressed to ensure responsible and fair use of these technologies. One of the most pressing concerns is bias and fairness. AI models learn from historical data, and if the training data contains biases—whether due to systemic inequalities, underrepresentation, or incorrect assumptions—the model may produce unfair or discriminatory outcomes. This can be particularly problematic in sensitive applications such as healthcare, finance, or hiring, where biased decisions can have real-world consequences. To mitigate this risk, it is essential to implement fairness-aware machine learning techniques, perform bias audits, and ensure diverse and representative training datasets.

Another major concern is transparency and explainability. AI and ML models, particularly deep learning-based models, are often referred to as "black boxes" due to their complexity and lack of interpretability. This lack of transparency can make it difficult to understand how decisions are made, leading to challenges in accountability and trust. In data engineering workflows, explainability is crucial for ensuring that data transformations and predictions are reliable and justifiable. Organizations must prioritize the development of explainable AI models, utilize model interpretability techniques, and provide clear documentation of AI-driven decision-making processes.

Privacy and security are critical ethical considerations when deploying AI and ML in data engineering. These models frequently process vast amounts of data, some of which may be highly sensitive, including personal or confidential business information. Mishandling such data can lead to privacy breaches, regulatory violations, and loss of trust. To address these risks, organizations should implement robust data encryption, access controls, and compliance frameworks such as GDPR or HIPAA. Ensuring data anonymization and employing techniques like federated learning can also help enhance data privacy while maintaining AI model performance.

### *5.2 Practical Considerations*

Beyond ethical concerns, the practical implementation of AI and ML in data engineering presents significant challenges that organizations must navigate to ensure successful deployment. One of the primary challenges is data quality. The effectiveness of AI models depends heavily on the quality of the training and operational data. Inconsistent, incomplete, or inaccurate data can lead to unreliable model predictions and inefficiencies in data processing. As a result, data engineers must implement robust data

cleaning, validation, and governance strategies to maintain high data quality. Automated data quality assessment tools powered by AI can also be leveraged to detect and rectify errors in real time.

Another key practical challenge is model maintenance. Unlike traditional rule-based systems, AI and ML models require continuous monitoring and updates to remain relevant and effective. Model drift, where a model's performance deteriorates over time due to changes in data patterns, is a common issue. To combat this, organizations must establish regular model retraining, validation, and performance monitoring mechanisms. Automated MLOps (Machine Learning Operations) pipelines can help streamline model updates and reduce the overhead associated with ongoing maintenance.

Finally, integration with existing systems can be a complex and resource-intensive process. Many organizations rely on legacy data infrastructure that may not be readily compatible with modern AI-driven approaches. Retrofitting AI into these systems often requires significant modifications to data pipelines, storage architectures, and processing workflows. Careful planning, incremental testing, and modular implementation strategies are essential to ensure smooth integration. Leveraging cloud-based AI services, containerization, and microservices architectures can also facilitate the deployment of AI-enhanced data engineering solutions without disrupting existing operations.

## 6. Recommendations for Future Research and Development

### *6.1 Research Directions*

As AI and ML continue to evolve, there are several promising areas for future research in data engineering. One key direction is hybrid approaches, which involve integrating traditional data engineering techniques with AI and ML-driven automation. Traditional rule-based methods excel in well-defined scenarios but often struggle with scalability and adaptability. AI-driven techniques, on the other hand, offer intelligence and automation but may lack reliability in complex enterprise environments. By combining both, researchers can create hybrid frameworks that leverage the efficiency of AI while maintaining the robustness and interpretability of traditional data engineering methods.

Another important research area is explainable AI (XAI). One of the biggest barriers to widespread AI adoption in data engineering is the lack of transparency in how models make decisions. Developing explainable AI techniques that provide clear, interpretable insights into AI-driven data processing decisions is crucial for building trust and accountability. This can involve techniques such as SHAP (Shapley Additive Explanations) values, feature importance scoring, and decision tree-based explanations, which help stakeholders understand why specific transformations or optimizations were applied to data pipelines.

Bias mitigation is another critical area that requires further exploration. Since AI models learn from historical data, they are susceptible to inheriting biases present in training datasets. This can lead to skewed data transformations, inaccurate predictions, and unfair decision-making. Future research should focus on developing bias detection algorithms, fairness-aware ML models, and de-biasing techniques that ensure equitable treatment of data across different demographic and industry-specific contexts. Additionally, regulatory and ethical guidelines should be incorporated into AI-driven data engineering workflows to ensure compliance with fairness standards.

Finally, real-time processing is a growing field that requires significant advancements. With the rise of IoT, edge computing, and streaming data applications, organizations need real-time data processing frameworks that can handle high-velocity data streams efficiently. Research in this area should focus on improving real-time data ingestion, feature extraction, and analytics using AI-driven optimizations. Advanced stream processing techniques, such as reinforcement learning-based adaptive data pipelines and AI-powered dynamic resource allocation, can help in making real-time data engineering more scalable and responsive.

### *6.2 Development Directions*

From a development perspective, the focus should be on building tools, skills, and ecosystems that facilitate the seamless adoption of AI and ML in data engineering. One key direction is the development of open-source tools and frameworks that lower the barrier to entry for organizations looking to integrate AI-driven solutions. While proprietary AI solutions exist, open-source alternatives can drive innovation, encourage collaboration, and make cutting-edge techniques accessible to a wider audience. Future developments should focus on AI-powered ETL (Extract, Transform, Load) pipelines, automated schema inference engines, and AI-enhanced data governance platforms.

Another essential step is investing in training and education programs tailored for data engineers. While AI and ML are powerful, their effective use requires specialized skills that many traditional data engineers may not possess. Universities, industry leaders, and training providers should collaborate to develop courses and certification programs focused on AI-driven data

engineering, covering topics such as automated feature engineering, AI-based anomaly detection, and MLOps integration. Organizations should also invest in upskilling their data engineering teams to ensure they can effectively leverage AI in real-world applications.

Lastly, fostering collaborative ecosystems can accelerate innovation and best-practice sharing. The intersection of AI and data engineering is evolving rapidly, and a collective effort involving academia, industry, and open-source communities is essential for sustainable progress. Future initiatives should focus on creating knowledge-sharing platforms, industry consortiums, and collaborative research projects where researchers, practitioners, and industry experts can come together to tackle common challenges. Encouraging cross-disciplinary collaborations between AI researchers, software engineers, and domain experts can also drive the development of AI-enhanced data engineering solutions that are both effective and ethical.

## 7. Conclusion

The integration of AI and ML into data engineering represents a transformative shift in how organizations manage, process, and analyze data. Traditional data engineering methods, while effective, often struggle with the growing complexity, volume, and velocity of modern data ecosystems. AI and ML introduce automation, intelligence, and adaptability, enabling more efficient data pipelines, improved data quality, and scalable architectures. From automated data quality assessment to real-time schema inference and pipeline optimization, AI-driven solutions are reshaping the data engineering landscape, making it more resilient and efficient.

However, the adoption of AI in data engineering comes with its challenges, including ethical concerns such as bias, transparency, and privacy, as well as practical issues like data quality, model maintenance, and system integration. Future research should focus on hybrid AI approaches, explainable AI techniques, and real-time data processing innovations, while development efforts should prioritize open-source tools, workforce training, and collaborative ecosystems. By addressing these challenges and leveraging AI responsibly, organizations can build more robust, scalable, and intelligent data engineering frameworks that drive meaningful insights and innovation in the digital age.

## References

[1] https://www.researchgate.net/figure/The-machine-learning-engineering-workflow_fig3_365808973
[2] https://atlan.com/automation-for-data-engineering-teams/
[3] https://dataengineeracademy.com/blog/automating-etl-with-ai/
[4] https://nexla.com/data-engineering-best-practices/data-engineering-automation/
[5] https://interviewkickstart.com/blogs/learn/automating-data-workflows-ai-prompt-engineering
[6] https://www.tredence.com/blog/data-engineering-automation
[7] https://www.linkedin.com/pulse/ai-driven-data-engineering-automating-pipelines-machine-kdrcc
[8] https://www.acceldata.io/blog/automation-in-data-engineering-essential-components-and-benefits
[9] https://prescienceds.com/artificial-intelligence-and-machine-learning-in-workflow-automation-beyond-manual-and-hardcoded-logic/\
[10] Chen, J., & Zhang, Y. (2020). Automated Data Quality Assessment Using Machine Learning. *Journal of Data Engineering*, 12(3), 45-58.