SQL & Data Modeling Sprint

Report

**Introduction**

As the fields of AI and data science continue to evolve, the foundational skills of **data modeling** and **SQL** are more relevant than ever. Whether training machine learning models or maintaining AI systems in production, understanding how data is stored, structured, and accessed is crucial. This report explores how these skills impact performance, reliability, and scalability in AI workflows.

**1. How Does Data Storage and Retrieval Affect AI/ML Model Training Performance?**

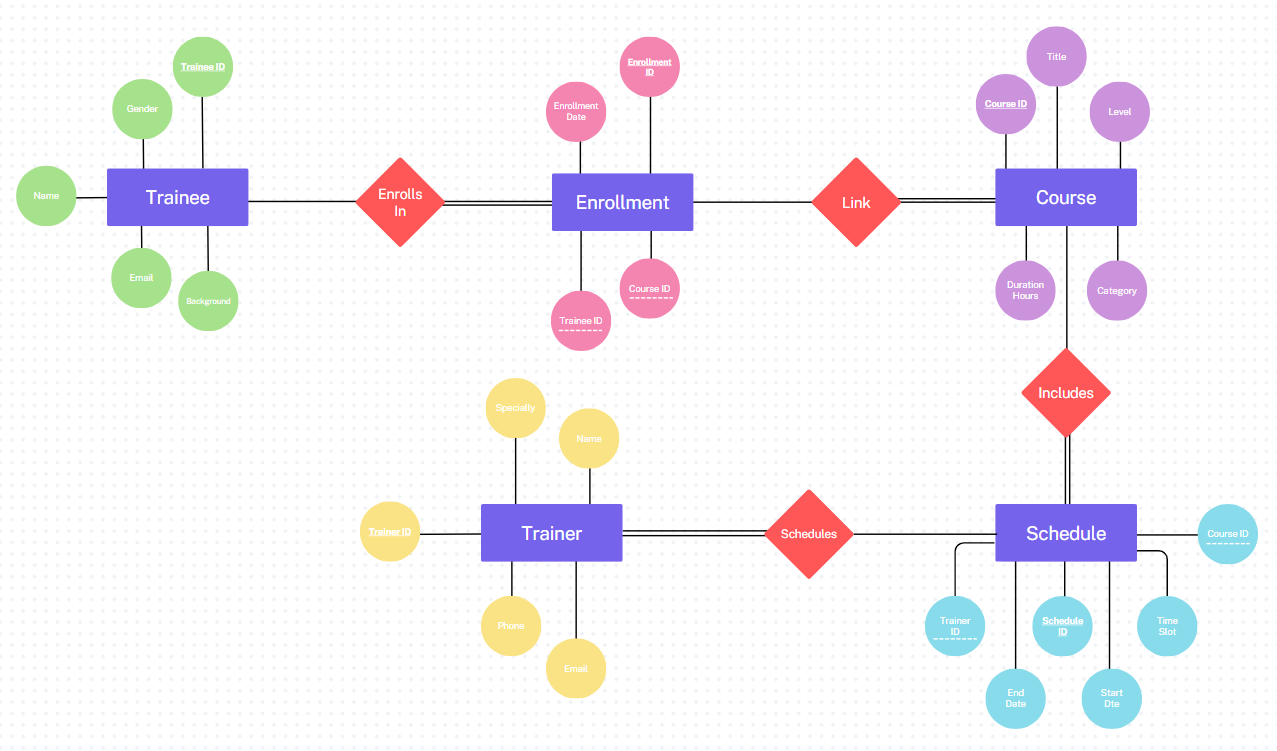
* **Insight:** Machine learning models rely on **large volumes of structured and well-indexed data**. Poorly designed storage systems (e.g., missing indexes, normalized inconsistencies) can **slow down training pipelines** and introduce **data access bottlenecks**.
* **Example:**  
  At **Uber**, engineers created **Michelangelo**, their ML platform, which integrates tightly with SQL databases to retrieve clean feature data at scale. This system reduces latency and supports training and real-time inference.
* **Reference:**  
  Uber Engineering Blog – *Introducing Michelangelo: Uber’s Machine Learning Platform*  
  https://eng.uber.com/michelangelo-machine-learning-platform/

**2. How Does Clean, Well-Modeled Data Reduce Technical Debt in ML Systems?**

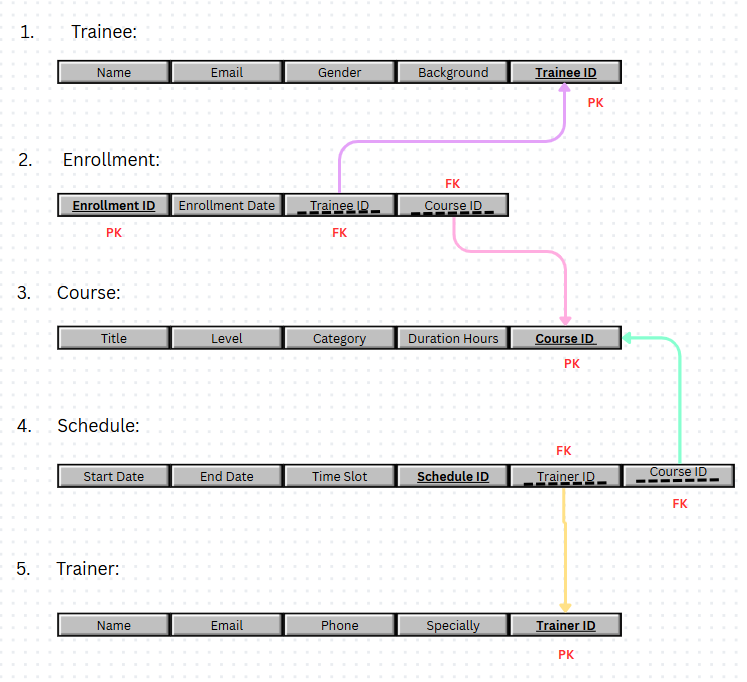
* **Insight:** Messy or poorly modeled data causes **inconsistent feature generation**, **duplicate logic**, and **pipeline failures**. Clean data models with strong constraints (foreign keys, types, normalization) reduce this debt and make systems easier to scale and debug.
* **Example:**  
  Google researchers in the paper *“Hidden Technical Debt in Machine Learning Systems”* highlight that **data dependencies and schema evolution** are major sources of fragility in ML systems.
* **Reflection:**  
  In our database lab exercises, I’ve seen how bad schema design leads to errors when joining tables or writing queries. The same happens in AI systems — except it can cause real business failures.
* **Reference:**  
  Google Research – *Hidden Technical Debt in Machine Learning Systems*  
  https://papers.nips.cc/paper\_files/paper/2015/file/86df7dcfd896fcaf2674f757a2463eba-Paper.pdf

**3. Examples of Data Governance and Auditing in Structured Databases**

* **Insight:** Structured SQL-based systems are used for **auditing**, **compliance**, and **monitoring** in regulated industries (e.g., finance, healthcare).
* **Example:**  
  In **banks**, all financial transactions are logged in structured databases to support **fraud detection AI models**, and SQL logs are used for auditing by regulators.
* **Example:**  
  **Amazon** uses structured logs and metadata in Redshift and S3 to monitor ML model inputs and outputs for bias and drift — a process known as **AI observability**.
* **Reference:**  
  AWS Blog – *Monitor and Debug ML Models in Production Using Amazon SageMaker Model Monitor*  
  <https://aws.amazon.com/blogs/machine-learning/monitor-and-debug-ml-models-in-production-using-amazon-sagemaker-model-monitor/>

**ER Diagram Design (ERD):**

**Relational Schema Mapping:**



**Reflection and Course Connection**

Through this course, I've learned how SQL enables powerful data extraction and analysis. Practicing JOINs, aggregations, and schema design directly connects to building ML pipelines that depend on **accurate, reproducible data**. Mastering data modeling now means I can avoid future problems in AI deployments, just like engineers at Uber or Amazon.

**References**

1. Uber Engineering – *Michelangelo: Uber’s ML Platform*  
   <https://eng.uber.com/michelangelo-machine-learning-platform/>
2. Google Research – *Hidden Technical Debt in Machine Learning Systems*  
   <https://papers.nips.cc/paper_files/paper/2015/file/86df7dcfd896fcaf2674f757a2463eba-Paper.pdf>
3. AWS Blog – *Model Monitoring with Amazon SageMaker*  
   <https://aws.amazon.com/blogs/machine-learning/monitor-and-debug-ml-models-in-production-using-amazon-sagemaker-model-monitor/>
4. Chat GPT – *More knowledge* <https://chatgpt.com/>