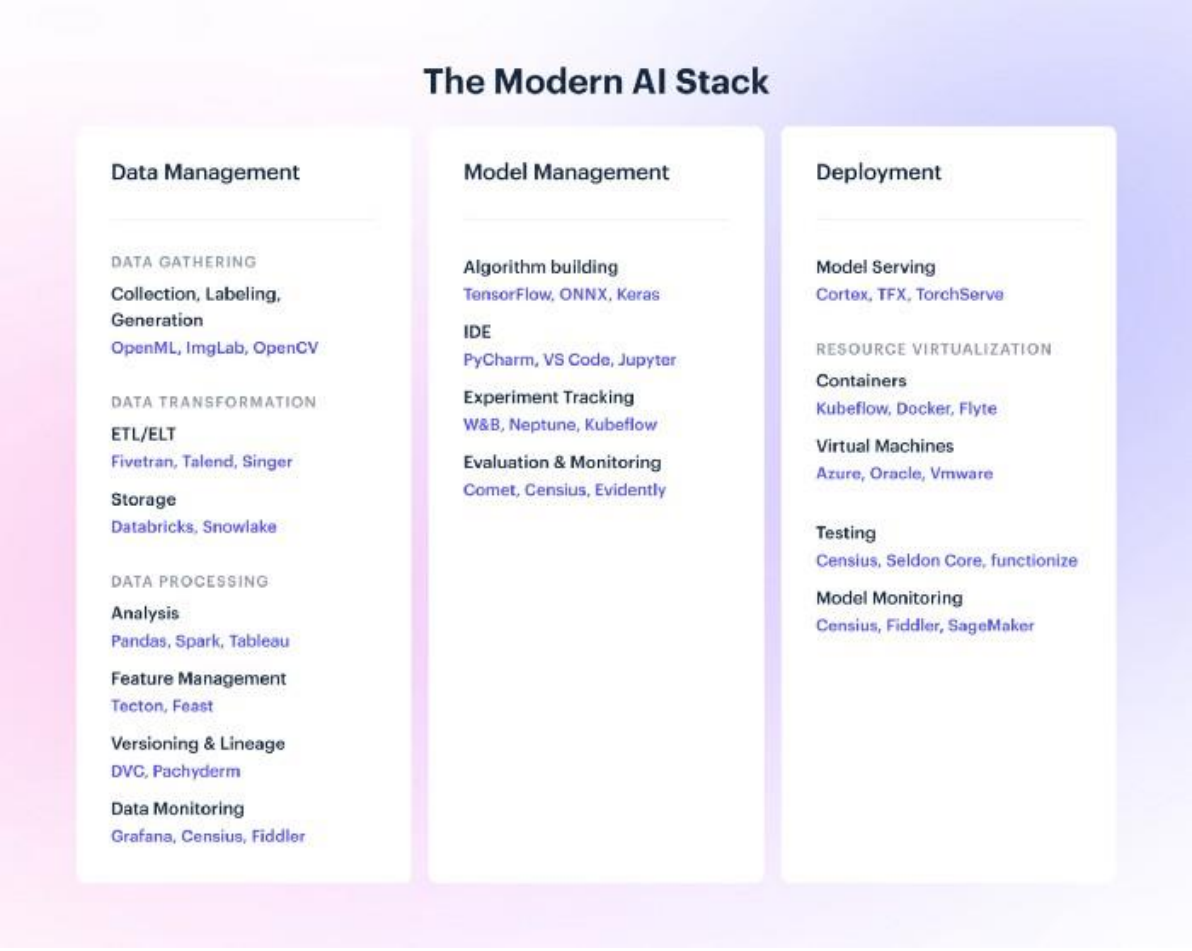
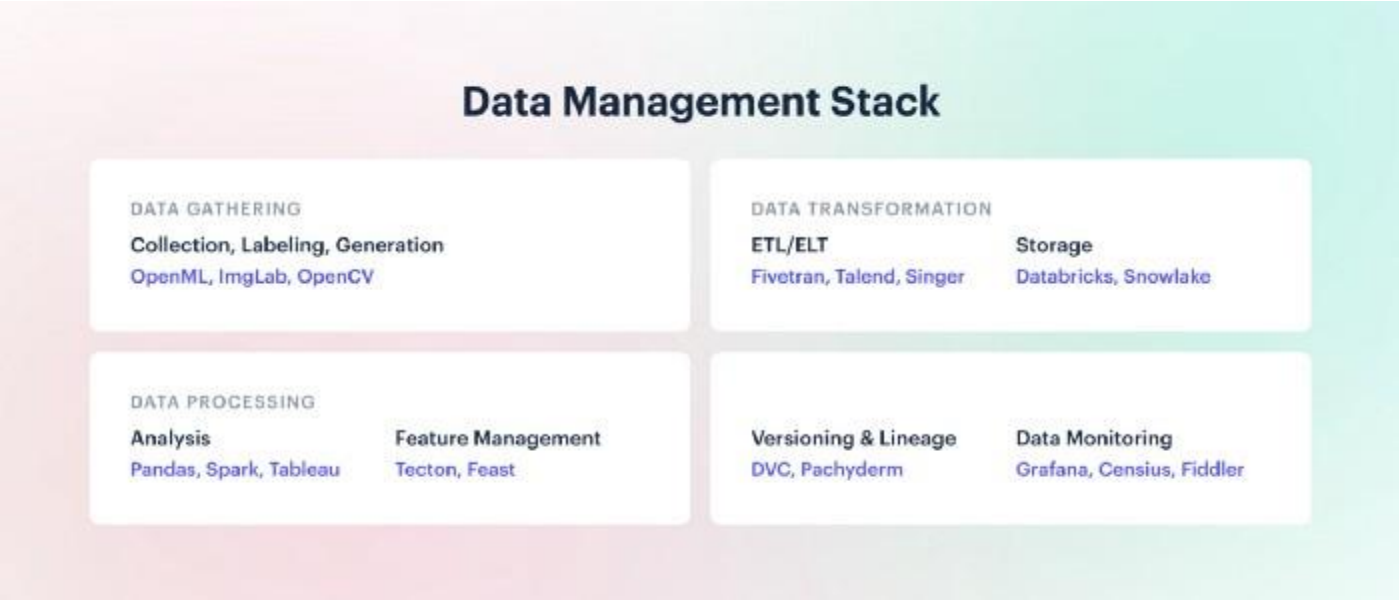
Stages of the modern AI Stack

The modern AI stack is a collection of tools, services, and processes imbibed with MLOps practices that allow developers and operations teams to build ML pipelines efficiently in terms of resource utilization, team efforts, end-user experience, and maintenance activities.



Stage 1: Data Management

Data management has five primary counterparts: Gathering, Transformation, Processing, Versioning, and Monitoring.



Data Gathering

The data gathering process experiences the intersection of several third-party tools and services that integrate with the internal tools to assemble usable data.

- Data collection
- Data labeling

The collected data needs to be processed and annotated so machines can learn the appropriate relationships in supervised solutions.

However, this stage still remains a manually-intensive process since algorithms have a tendency to miss specific cases and reviews are time-taking. You need to have clear parameters on what types of data to collect and be very rigorous in the labeling process. Tools like Amazon's Mechanical Turk and Ground Truth are also available to offer support through outsourcing.

- Synthetic data generation

In spite of high data volumes, sometimes data is not always available for very specific use cases or is not used directly due to privacy concerns, such as rare disease data. Even though such data is hard to come by, the demand for such models with specific data requirements is comparatively high. There are ample tools and libraries that support data generation across a wide range of data types including images, text, tables, etc.

Data Transformation and storage

Data storage requires reliable systems that can support a variable volume of data over the long term without corrupting it. To accommodate this wide variety of needs, organizations are increasingly dabbling with multiple storage methods for both structured and unstructured data such as data warehouses, data lakes, databases, etc.

- ETL, ELT, and Reverse ETL

ETL (Extract, Transform, Load) and ELT (Extract, Load, Transform) are two different types of data transformation systems. ETL is the traditional method and is favorable when processed data is a higher priority compared to preserving the raw data. It loads data to a temporary staging location, processes it, and then stores it in the target location. ELT is the more modern approach, ideal for time-optimization and high volumes of data. It loads data to the target location first and then processes it.

Reverse ETL is a more recent development and has just come under the spotlight. It connects data stores to customer-facing or action-based systems like CRMs and ERPs to enable shared real-time insights across applications, thus personalizing customer interactions at scale.

- Storage

Data storage has several factions, each with a different purpose. For instance, data lakes store unstructured data and aggregate all the available data in a flexible format.

The primary differences between the three unique types of data storage facilities are volume, interaction frequency, and structure. While simple databases store structured and filtered data and are ideal for frequent interaction, data warehouses are an advanced version of databases, optimized for analyzing and storing larger volumes of structured data across multiple touch points. Since warehouses depend

on the transformation and loading schedule, the updates are lagged as per the ETL/ELT frequency.

The latest storage technology is the Data Lakehouse, which emerged from the need to store a wide variety and a huge quantity of unstructured data that couldn't be processed instantly for either data warehouses or simple databases. As the data generated by digital systems and multiple customer touchpoints continued to grow, data lakehouses offered the solution to manage rich and high-quality data without the need to lose or process it. Being format-agnostic and cost-effective, Data Lakehouse is an ideal and fast way to store data for future analysis.

Data Processing

This is the process of converting raw data into useful data that can be consumed by the model. The raw inputs are converted to numbers, vectors, embeddings, etc. for model consumption.

- Data analysis

Data analysis (or exploratory data analysis) is one of the most time-intensive activities in the entire ML lifecycle. Feature management

After meeting the challenges of managing heavy volumes of raw data, their respective features, and feature versions, there is no surprise why Features stores are the talk of the town. Feature stores store, compute,

manage, and version features across machine learning solutions, making the entire feature pipeline much more reliable than manual management

Data versioning and lineage

Data versioning is as critical as code versioning for the same reasons. Given that data is dynamic and updated frequently, the same process will not produce the same results on the data unless it is carefully versioned.

Data Lineage on the other hand is the process of carefully mapping the journey of data across the entire ML pipeline. With data lineage, users can form a story out of the data, see how versions evolved over time, and make logical connections between every data touchpoint.

Data monitoring

Real-world data comes with a lot of loopholes due to input issues or manual errors. If erroneous data is allowed to pass into models, the model results could be misleading. However, maintaining the quality of large-scale data with, say, millions of data points, is both time and resource-intensive. Automated monitoring is the most immediate MLOps practice that can be established within a restricted budget and time.