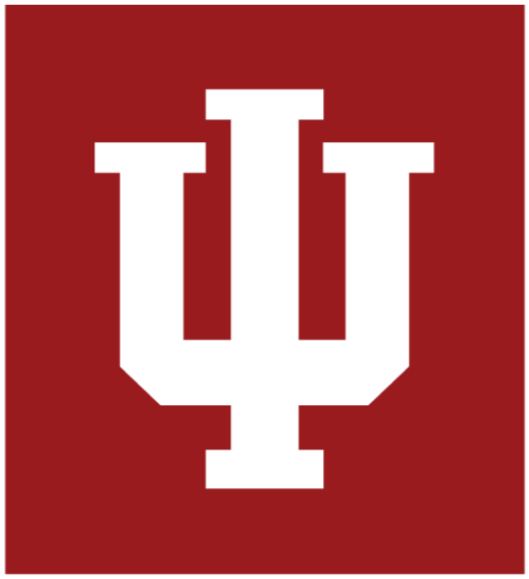
MID-TERM PROJECT REPORT



Unveiling Trends: A Cloud-Driven Data Engineering Project on AWS

ENGR - 516: Engineering Cloud Computing
Prof. Dingwen Tao

Unveiling Trends: A Cloud-Driven Data Engineering Project on AWS

Prachi Jethava
pjethava@iu.edu

Nisarg Shah
ns26@iu.edu

Pavan Pandya
pnpandya@iu.edu

Project Goals:

The primary objective of our project is to develop a cloud-based data engineering pipeline using Amazon Web Services (AWS) to extract valuable insights from YouTube data. By leveraging modern data processing tools and scalable infrastructure provided by AWS, we aim to empower stakeholders with actionable intelligence to optimize their online video strategies. Specifically, we aim to:

- Analyze specific trends within YouTube data, such as emerging video categories and audience demographics for trending videos.
- Implement a scalable and cost-effective architecture on AWS to handle large volumes of YouTube data efficiently.
- Provide stakeholders with interactive dashboards and visualizations to explore and analyze the extracted insights.

AWS Services Used:

1. **Amazon S3 (Simple Storage Service):**

Amazon S3 is an object storage service that offers industry-leading scalability, data availability, security, and performance. It allows users to store and retrieve any amount of data from anywhere on the web. S3 provides highly durable and available storage infrastructure, making it suitable for a wide range of use cases, including data backup and recovery, data archiving, content distribution, and data lakes.

2. **AWS IAM (Identity and Access Management):**

AWS IAM is a web service that helps you securely control access to AWS services and resources. IAM enables you to create and manage AWS users and groups, assign granular permissions, and

define roles to control access to resources. By implementing IAM, you can ensure that only authorized users and applications have access to your AWS resources, helping to maintain security and compliance in your environment.

3. **AWS Glue:**

AWS Glue is a fully managed extract, transform, and load (ETL) service that makes it easy to prepare and load data for analytics. Glue automates the process of discovering, cataloging, and transforming data, allowing you to build scalable and efficient data pipelines without managing infrastructure. With Glue, you can easily extract data from various sources, transform it using built-in or custom transformations, and load it into data lakes, data warehouses, or analytics platforms.

4. **AWS Lambda:**

AWS Lambda is a serverless compute service that lets you run code in response to events without provisioning or managing servers. Lambda functions are stateless, event-driven functions that can be triggered by various AWS services, such as S3, DynamoDB, and SNS, as well as custom events. With Lambda, you can execute code in response to changes in data, user actions, or system events, allowing you to build highly scalable and cost-effective applications and workflows.

5. **Amazon Athena:**

Amazon Athena is an interactive query service that makes it easy to analyze data in Amazon S3 using standard SQL. Athena allows you to run ad-hoc queries on data stored in S3 without the need to set up or manage infrastructure. You can query data directly from S3 using SQL syntax, and Athena automatically scales to handle any amount of data, making it ideal for analyzing large datasets and performing complex analytics tasks.

Project Workflow:

1. **AWS IAM User:**

- Following the project requirements, an IAM user has been created with appropriate permissions to access AWS services. Initially, by logging into the AWS Management Console and navigating to the IAM dashboard, a new IAM user was created "**ECC"**.

- Subsequently, policies were attached to this user, granting the necessary permissions to access the required AWS services, including Amazon S3 and AWS Glue. Additionally, IAM roles were established to ensure secure access to AWS Glue and AWS Lambda services.
- Specifically, an IAM role was configured with the requisite permissions for the AWS Glue service to access data stored in S3. Similarly, another IAM role was created to grant AWS Lambda the necessary permissions to access both S3 and the Glue service.
- This meticulous setup ensures that the IAM user and associated roles have precisely defined permissions, facilitating secure and efficient interaction with AWS services within the data engineering pipeline.

2. **AWS CLI Setup:**

- To streamline interaction with AWS services programmatically, the AWS Command Line Interface (CLI) has been installed and configured. This process involved installing the AWS CLI on the local machine or codespace, employing the appropriate method tailored to the operating system in use.
- Subsequently, the AWS CLI was configured with the credentials of the IAM user created earlier, ensuring seamless access to AWS resources. Additionally, default settings for the AWS CLI were established, including setting the default AWS region and output format, to optimize the command-line interface for subsequent interactions with AWS services.

3. **Data Ingestion with AWS S3:**

- In the context of data ingestion, data collected from YouTube Data API or Kaggle is efficiently managed and stored within AWS S3 buckets. Leveraging the AWS CLI, data files are uploaded to designated S3 buckets, facilitating the transfer of YouTube data to the AWS cloud environment.
- Furthermore, to ensure systematic organization and accessibility, data files are organized within S3 buckets according to a specified naming convention. This structured approach to data ingestion enables the seamless integration of YouTube data into the AWS ecosystem, laying the foundation for subsequent processing and analysis within the data engineering pipeline.

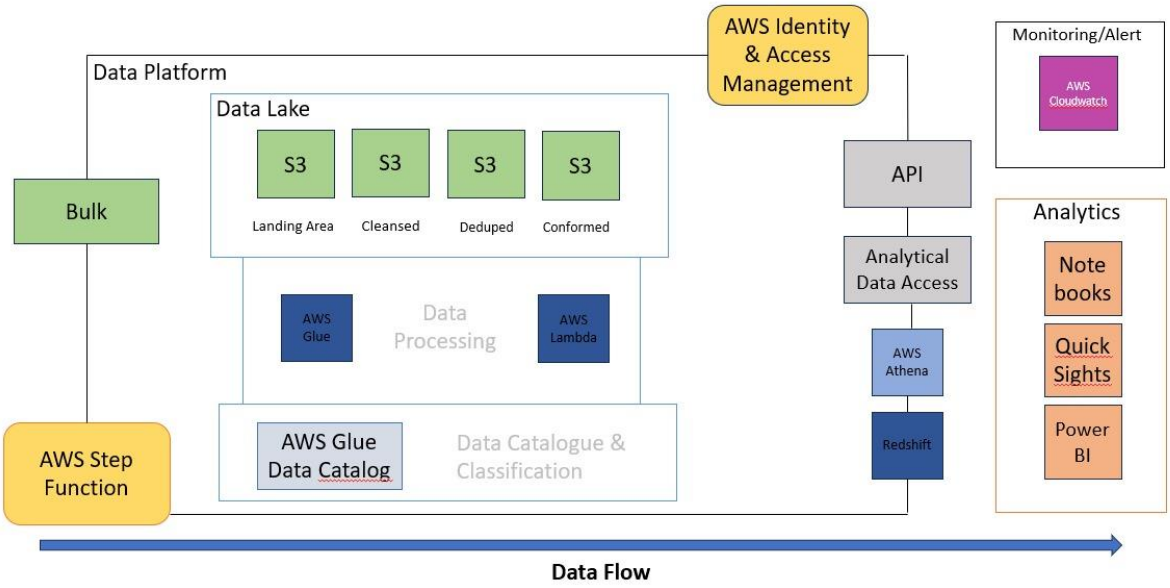
4. **AWS Glue Crawlers:**

- To gain a comprehensive understanding of the raw data stored in the AWS S3 buckets, AWS Glue crawlers have been employed. This process involves configuring the Glue crawler to traverse the designated S3 bucket paths containing both JSON and CSV files.
- Firstly, the Glue crawler is configured to crawl the S3 bucket path housing the raw data reference JSON files. Subsequently, upon execution of the Glue crawler, the schema of the JSON files is automatically inferred, facilitating the creation of a database with corresponding tables based on the discovered schema.
- Similarly, a second AWS Glue crawler is configured to traverse the S3 bucket path containing the raw data CSV files. Once again, the Glue crawler is executed to ascertain the schema of the CSV files, consequently generating a database with tables reflecting the identified schema.
- Notably, the partition key of the region is automatically generated as part of the schema discovery process, given the structured organization of the data files within the S3 bucket directory tree format.
- This meticulous schema exploration and database creation process via AWS Glue enables a nuanced understanding of the data structure and paves the way for streamlined data processing and analysis within the project's data engineering pipeline.

5. **Amazon Lambda:**

- To resolve JSON format errors incompatible with AWS services, an AWS Lambda solution is implemented. This lightweight ETL job converts JSON files to the Parquet format, overcoming compatibility issues.
- Essential parameters such as S3 path, Glue catalog details, and data write operation are configured as environment variables. Additionally, the Lambda function is equipped with the "AWSSDKPandas-Python38" layer for necessary dependencies. Configuration adjustments optimize performance, including timeout time and memory requirements.
- The Lambda function performs data transformations, extracting required columns from JSON files. Validation is ensured through test events, and triggers are set on the S3 bucket path to initiate Lambda processing upon new JSON file detection. This comprehensive approach streamlines data processing, ensuring compatibility with AWS services.

Project Architecture:



Related Work Gap Analysis:

1. **Scalability and Efficiency:**

Existing data engineering solutions often struggle to efficiently handle large volumes of data, leading to scalability issues and increased processing times. Our project leverages the scalable infrastructure provided by AWS, including services like Amazon S3, AWS Glue, and AWS Lambda, to design a robust architecture capable of handling massive datasets efficiently. By utilizing serverless computing and parallelized ETL workloads, we aim to improve scalability and optimize processing times, thus addressing the scalability and efficiency gap highlighted in the research paper.

2**. Data Quality and Reliability:**

Data quality and reliability are critical factors in deriving accurate insights from data. Traditional approaches may lack mechanisms for rigorous data quality checks and validation, leading to potential inaccuracies in analysis outcomes. Our project emphasizes data quality throughout the pipeline, implementing validation mechanisms at various stages, including data ingestion,

transformation, and analysis. By conducting thorough data quality checks and ensuring adherence to predefined schema during processing, we strive to enhance the reliability of insights extracted from YouTube data, thereby addressing the data quality and reliability gap identified in the research paper.

3. **Cost Optimization:**

Cost optimization is a significant concern in cloud-based data engineering projects, where inefficient resource utilization can lead to unnecessary expenses. Our project prioritizes cost efficiency by leveraging cost-effective AWS services such as S3 for storage and Lambda for serverless computing. By optimizing data storage and processing on AWS and adopting a pay-as-you-go model, we aim to minimize costs while ensuring scalability and performance. Through continuous monitoring and optimization of resource usage, we endeavor to address the cost optimization gap highlighted in the research paper.

4**. Interactive Analysis and Visualization:**

Traditional data engineering pipelines may lack robust mechanisms for interactive analysis and visualization, limiting stakeholders' ability to explore and derive insights from data effectively. Our project integrates Amazon Athena and Amazon QuickSight to provide stakeholders with interactive dashboards and visualizations, enabling intuitive exploration and analysis of YouTube data. By facilitating ad-hoc querying and visualization of data stored in S3, we aim to empower stakeholders with actionable intelligence for optimizing their online video strategies, thus bridging the gap in interactive analysis and visualization identified in the current work.

Proposed Tasks:

Our proposed methodology for this project entails a focused data selection and analysis approach followed by the design of a scalable and cost-effective architecture on AWS. To begin, we will meticulously explore Kaggle to identify a dataset pertinent to our analysis objectives, encompassing essential attributes such as video views, likes, comments, and viewer demographics. This targeted analysis will delve into areas such as trending video categories, audience engagement, and demographic targeting, providing valuable insights for content creators and

marketers. Leveraging this data, we aim to refine content strategies and optimize audience targeting for increased engagement and effectiveness.

Subsequently, we will design a robust cloud-based architecture on AWS, incorporating key components to facilitate efficient data processing and analysis. Amazon S3 will serve as the primary storage solution due to its scalability and cost-effectiveness, complemented by AWS Glue for ETL tasks and data cataloging. Furthermore, serverless computing with AWS Lambda and integration with Apache Spark via AWS Glue will enable parallelized ETL workloads and complex data transformations for improved performance, especially with large datasets. Upon processing, data will be stored in Amazon Redshift, with Amazon Redshift Spectrum facilitating efficient querying of data stored in S3. Stakeholders will access insights through Amazon QuickSight, ensuring user-friendly exploration and analysis of the data.

Key Success Metrics:

The success of our project will be evaluated based on several key performance indicators (KPIs). We will monitor the time to insights, ensuring prompt delivery of actionable insights to stakeholders by optimizing pipeline performance. Cost efficiency will be prioritized through the optimization of data storage and processing on AWS, utilizing services like S3 and Lambda to minimize costs. Additionally, the accuracy of insights extracted from YouTube data will be rigorously validated through data quality checks, ensuring reliability and usefulness for informed decision-making.

Progress on Proposed Tasks:

Significant progress has been made across various stages of the proposed tasks outlined for the project. Firstly, the foundational setup of AWS IAM users and roles has been meticulously executed, ensuring secure access to AWS services. The creation of an IAM user "ECC" and the establishment of corresponding roles with precise permissions lay the groundwork for subsequent interactions within the AWS environment. Furthermore, the AWS CLI has been successfully configured, enabling streamlined interaction with AWS services programmatically. By installing and configuring the AWS CLI with the necessary credentials and default settings, the project team has facilitated efficient management and manipulation of AWS resources.

In parallel, substantial advancements have been achieved in the realm of data ingestion and processing. Leveraging AWS S3, YouTube data collected from the Data API or Kaggle has been efficiently managed and stored within designated S3 buckets. The systematic organization of data files within S3 buckets, coupled with the utilization of AWS Glue crawlers, has enabled the comprehensive exploration of raw data schemas. Through meticulous schema discovery and database creation processes, facilitated by AWS Glue crawlers, a nuanced understanding of the data structure has been attained. Additionally, the implementation of AWS Lambda solutions has addressed JSON format errors, ensuring compatibility with AWS services and streamlining data processing workflows. These accomplishments underscore the project's progress towards establishing a robust data engineering pipeline on AWS, poised to deliver actionable insights from YouTube data for stakeholders.

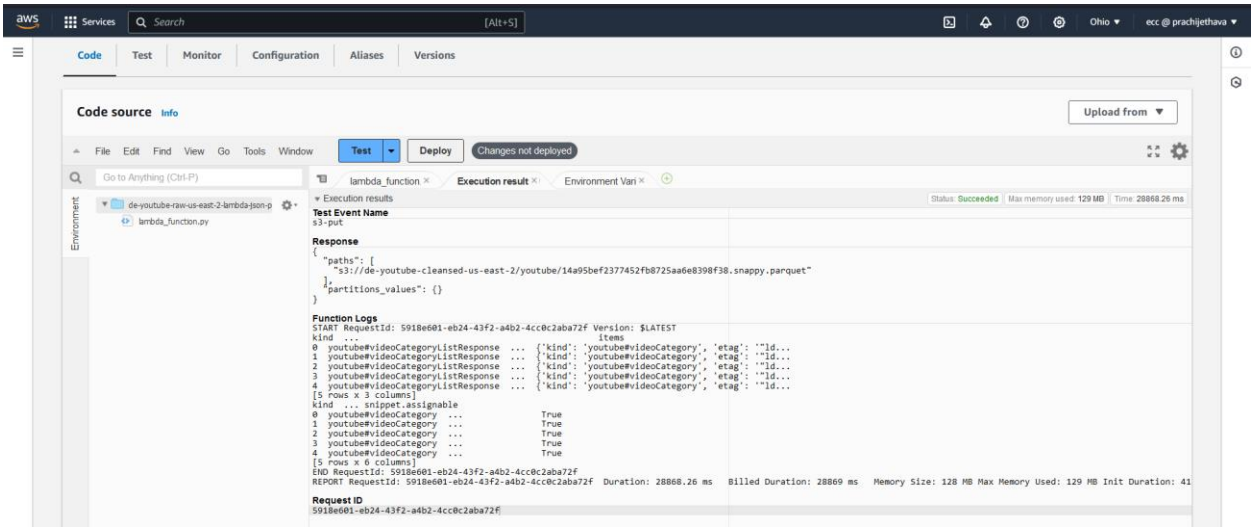
Initial Benchmark and Profiling:

Data Testing on a Small Scale:

The project team conducted initial testing of the data on a small scale to validate the functionality and integrity of the data engineering pipeline. This testing phase involved ingesting a subset of YouTube data into the AWS environment and performing basic data processing tasks to ensure that the pipeline functions as expected. By conducting testing on a small scale, the team identified and addressed any potential issues or anomalies in the data ingestion and processing workflows, laying the foundation for future scalability and performance optimizations.

Data Normalization:

As part of the data preprocessing phase, the project team normalized the raw data, which was initially in JSON format. Normalization involves structuring and organizing the data into a consistent format, making it suitable for analysis and storage in relational databases or data warehouses. By normalizing the data, the team ensures consistency and uniformity across different data sources, enabling efficient querying and analysis downstream. This normalization process enhances data quality and accessibility, facilitating more accurate and insightful analytics.



Screenshot showing the output of lambda function executed on a sample of data

Preliminary Observations:

1. **Importance of IAM User Management**: Establishing an IAM user before commencing AWS usage proved foundational in bolstering security measures. This practice not only safeguards the root user credentials but also enables granular control over access permissions, mitigating the risk of unauthorized access or data breaches.

2. **Role and Permission Granularity**: A meticulous approach towards defining roles and permissions for IAM users is imperative within cloud-based projects. Accurate delineation of roles ensures that users possess precisely the requisite permissions for executing designated tasks, thereby fostering a secure and efficient operational environment.

3. **Investment in AWS Services Understanding**: Significant time investment was dedicated to comprehending the functionalities and nuances of various AWS services. This exploration phase was indispensable in selecting appropriate services aligning with project requirements, optimizing resource utilization, and designing an efficient architecture for the data engineering pipeline.

4. **Data Exploration and EDA Significance**: Extensive efforts were directed towards understanding the intricacies of the YouTube dataset and performing exploratory data analysis

(EDA). This phase facilitated the identification of key relationships between dataset features, laying the groundwork for informed decision-making during subsequent stages of data processing and analysis.

5. **JSON Data Compatibility Considerations**: Working with JSON files necessitated careful consideration of compatibility with AWS services, particularly Amazon Athena. Efforts were dedicated to transforming data into a format compatible with AWS Athena's requirements, ensuring seamless integration and query execution.

6. **Architecture Design Iteration**: Iterative refinement of the project architecture was essential to accommodate evolving requirements and optimize the efficiency of data pipelines. Deliberate design decisions were made to leverage AWS services synergistically, balancing scalability, performance, and cost considerations.

Timeline Status and Next Steps:

As per the initial project proposal, significant progress has been achieved in various phases of the project within the stipulated timeframe. The project commenced with meticulous planning and research, focusing on defining project objectives and exploring AWS services suitable for data engineering. Subsequently, substantial advancements were made in data exploration and preparation, where the team meticulously cleaned and understood the dataset for analysis.

The setup and configuration phase on AWS followed, with the team successfully establishing AWS accounts, IAM permissions, and S3 bucket for data storage. Moving forward, considerable efforts were dedicated to building the data pipeline using AWS Glue, facilitating data cataloging and transformation for analysis. The subsequent phases including querying and analysis using Amazon Athena, serverless computing with AWS Lambda development, and optimization for performance tuning (yet in progress).

As of the mid-term project report, the project is progressing in alignment with the proposed timeline with a little delay in the timeline that includes querying using AWS Athena, with key tasks accomplished within the designated timeframes. The foundational setup, exploratory data

analysis, data ingestion, schema exploration, and preliminary data processing have laid a robust groundwork for the subsequent phases of the project. The team's adherence to the proposed timeline reflects a disciplined approach to project management and execution, ensuring timely progress towards the project objectives.

Next Steps:

Moving forward, the project will prioritize completing the remaining pipeline flow outlined in the architecture. This entails refining data processing tasks, optimizing data storage and retrieval mechanisms, and fine-tuning the integration between AWS services to ensure seamless data flow. By focusing on completing the pipeline flow, the project aims to streamline data processing workflows and enhance the efficiency and scalability of the data engineering pipeline.

Once the pipeline flow is fully established and operational, the team will shift its focus towards creating a dashboard for visualization. Leveraging tools such as Amazon QuickSight or custom-built visualization solutions, the team will develop interactive dashboards that enable stakeholders to explore and analyze insights derived from the processed data. The dashboard will provide intuitive visualization of key metrics, trends, and patterns extracted from the YouTube data, empowering stakeholders to make informed decisions based on actionable intelligence.

Following the deployment of the visualization dashboard, the project will embark on integrating real-time streaming data processing into the existing data engineering pipeline. This involves exploring AWS services such as Amazon Kinesis and AWS Lambda for stream processing and event-driven architectures. By incorporating real-time streaming data processing, the project aims to capture and analyze dynamic trends and patterns as they unfold, enhancing the relevance and timeliness of the insights provided to stakeholders.

In summary, the next steps for the project involve completing the remaining pipeline flow, developing a visualization dashboard, and integrating real-time streaming data processing. By sequentially addressing these objectives, the project aims to further enhance its capabilities in

deriving valuable insights from YouTube data and delivering actionable intelligence to stakeholders.

References:

1. https://docs.aws.amazon.com/
2. https://aws.amazon.com/getting-started/hands-on/
3. https://arxiv.org/ftp/arxiv/papers/1409/1409.7733.pdf
4. https://www.youtube.com/watch?v=e0mKkTgxZ_Y
5. https://www.youtube.com/watch?v=kSppyMr6sXA
6. https://www.youtube.com/watch?v=SslOHC-qji4