Group Member: CHONG XIAN JUN MCS211047

Lambda architecture is a generic term that refers to a scalable and fault-tolerant data processing architecture designed to handle batch and streaming data processing in big data analytic. Lambda architecture has been adapted in the cloud services.

*Discuss **an application of lambda architecture** and **choose the appropriate tool** or **software that can be used in the cloud services to support the data processing in the application***

Application of lambda architecture

One of the main applications of lambda architecture is for the processing and analytics of massive data sets that includes both real-time data and historical data.

Traditionally, data processing was usually conducted through relatively simple methods. Two of which that are the most popular are batch processing or stream processing that come with their respective pros and cons:

- Batch Processing: allows for complex analysis on large, complex data sets. However, it features an inevitable delay between data collection and processing output due to its nature.
- Stream Processing: allows for real-time processing. But as the data is computed as it is received, it can only be used for simple tasks and has a much lower throughput (data process per second) than batch processing [1]

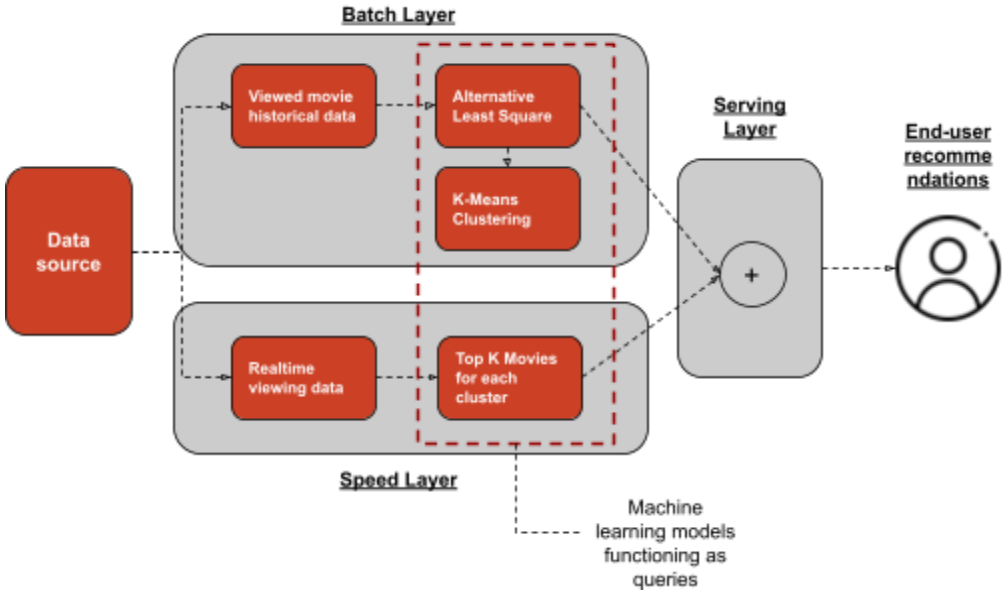
In the age of big data 1.0, the analysis of large, complex data was dealt with by batch processing, as it is versatile, effective, fault-tolerant and scalable. It was functioning fairly effectively until the arrival of age big data 2.0, where there is a growing need in the field of real-time big data analysis. Since then, Lambda architecture has been deployed widely to solve the limitations in the simpler processing methods.

A common application of Lambda architecture in real-time big data analysis is in recommendation systems. In this particular context, one of the main challenges is to recommend trending items to customers with massively streaming data on top of their preferences based on historical data. [2]

For instance, content recommendations by Netflix:
Lambda architecture used here processes data with three major layers; batch layer, speed layer, and serving layer.

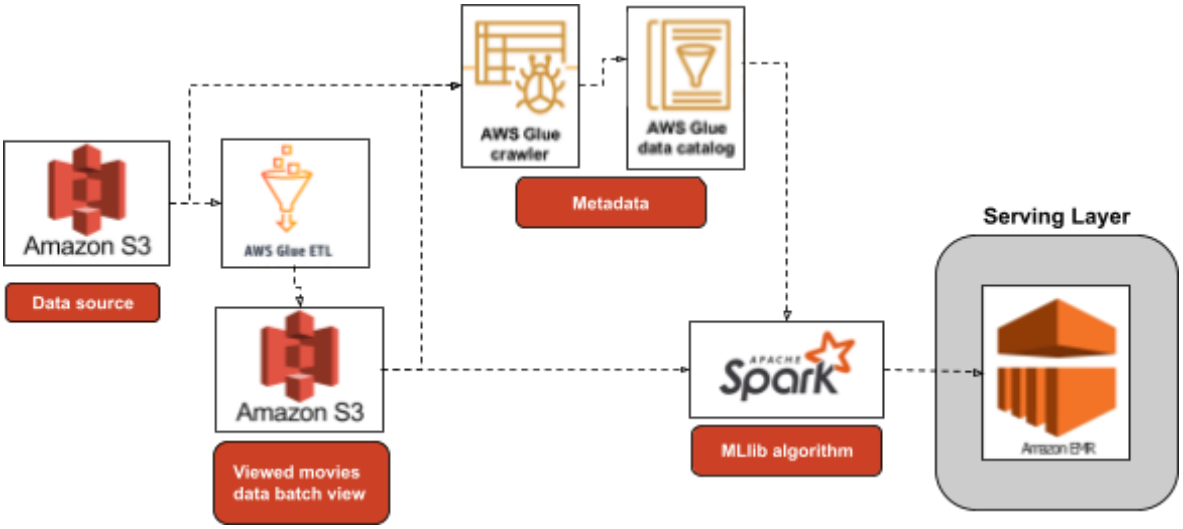
- **Batch layer**: retrieve raw datasets, pre-compute batch views from these datasets and send them to the serving layer.
- **Speed layer**: process real-time data and send to serving layer
- **Serving layer**: Merge and index views from batch layer and speed layer so that they can be queried.

On each layer, machine learning algorithms such as ALS and K-means algorithms are applied on the dataset to analyze these contents to be recommended to the end users. By combining batch processing and stream processing through Lambda architecture, Netflix is able to make full use of both large, complex historical data and latest, real-time data to provide the most relevant recommendations to its users. Process refers to the diagram below.



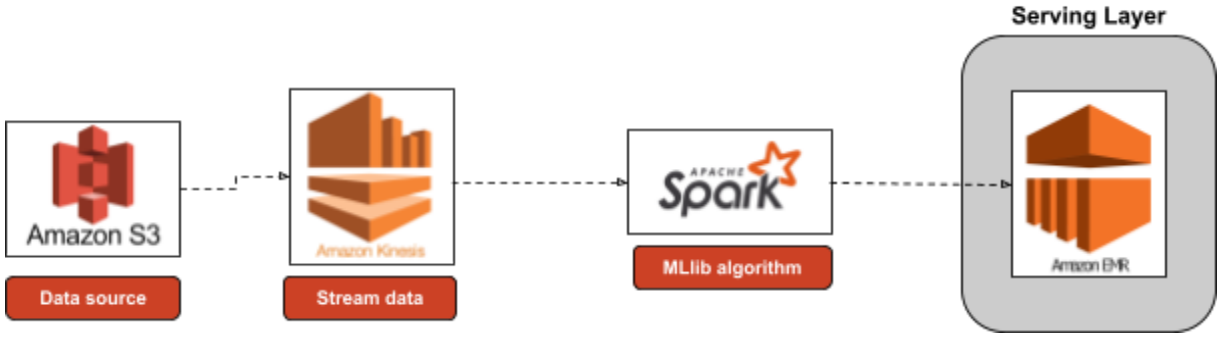
Choose appropriate cloud services software to support data processing in application mentioned above

While there is no open documentation on Netflix's implementation of lambda architecture to the best of my knowledge, we can design a very simplified version of the recommendation system that describes the cloud service softwares that might be used for the said recommendation system.



Cloud services software supporting **batch layer** in Lambda architecture of recommendation system.

Batch layer: A streaming platform like Netflix consists of incredibly large and complex data from a variety of sources. These big, complex unstructured data that can be stored with Amazon S3 data lakes. This raw data is then converted by AWS Glue ETL into a batch view to be fed into the machine learning algorithm with Apache Spark. Other than this, metadata such as title, copyright information, duration of the video, or creation date, are also great indicators of whether the users will like a certain recommended content. Thus it can be extracted through AWS Glue Crawler and AWS Glue Data Catalog and be fed into the recommendation system's machine learning models too. The output is transferred to the serving layer to merge with that from the batch layer. [3]



Cloud services software supporting **speed layer** in Lambda architecture of recommendation system.

Speed Layer: The speed layer can be built by using Amazon Kinesis. Under the framework of Amazon Kinesis, one of the simplest options to process the real-time data that passes through its Kinesis Data Stream or Kinesis Data Firehose is Kinesis Data Analytics as it provides the easiest way to process the data through SQL. This enables users to gain actionable insight in near real-time. This data is then used to support machine learning algorithms in Apache Spark to recommend content for its users. This output is then fed into the serving layer to merge with that from the batch layer. [3]

Serving Layer: Finally, the serving layer can be implemented with Spark SQL on Amazon EMR (Amazon Elastic MapReduce) to process the large amount of data [4] from both batch layer and stream layer. The merged data set can be written to Amazon S3 for further visualization.

References:

1. Learn | Hevo. “Batch Processing vs Stream Processing: 9 Critical Differences.” Accessed May 8, 2022. https://hevodata.com/learn/batch-processing-vs-stream-processing/.
2. Numnonda, Thanisa. “A Real-Time Recommendation Engine Using Lambda Architecture.” *Artificial Life and Robotics* 23 (December 29, 2017). https://doi.org/10.1007/s10015-017-0424-8.
3. Rajeev, Srinivasan, Ujjwal Ratan. “Lambda Architecture for Batch and Stream Processing” Amazon Web Services (October 2018). https://d1.awsstatic.com/whitepapers/lambda-architecure-on-for-batch-aws.pdf
4. “What Is Amazon EMR? - Amazon EMR.” Accessed May 8, 2022. https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-what-is-emr.html.