Birla Institute of Technology & Science, Pilani
Work Integrated Learning Programmes Division
Second Semester 2020-2021

Assignment 1

Course No. : DSECL ZC556
Course Title : Stream Processing and Analytics
Nature of Exam : Open Book
Weightage : 10%
Duration : 10 days

No. of Pages = 2
No. of Questions = 4

Social Media Analytics with Streaming Data

Social media analytics solutions help organizations understand trending topics. Trending topics are nothing but subjects and attitudes that have a high volume of posts on social media. Sentiment analysis, also termed as opinion mining, makes utilization of social media analytics tools to determine attitudes toward a product or idea. Because of the hashtag subscription model, Real-time Twitter trend analysis is a great example of an analytics tool that enables organizations to listen to specific keywords (hashtags) and develop sentiment analysis of the feed.

"AdMagic" is a company that has a news media website is interested in gaining an advantage over its competitors by featuring site content that is immediately relevant to its readers. The company wants to explore the social media analytics tools on the topics that are relevant to readers by doing real-time sentiment analysis of Twitter data. To identify trending topics in real time on Twitter, the company needs real-time analytics about the tweet volume and sentiment for key topics.

You are appointed as a Streaming Analytics expert for this firm which is looking for utilizing the solutions / platforms available from the Streaming Analytics space. As the firm's maturity level in the social media data analytics space is at very nascent stage, you need to help them to understand how Streaming Analytics is helpful in their several use cases and also further on identifying the various options of tools and platforms those can be leveraged for this activity.

Microsoft Azure is leading player in the field of streaming analytics. Under the umbrella term "Streaming analytics", they have developed a several cloud services to handle various streaming analytics use cases in very simpler manner. One of the solution for streaming media analytics is described at this blog. You can refer to this blog or other documentation provided by Microsoft team while interacting with the client.

Q1. You need to introduce the client with other streaming analytics tools available for streaming analytics which are suitable for the use case of social media analytics. For that purpose, you need to formulate a comparison that describes the available tools / solutions along with their strength and weaknesses.

- Narration should have
  - brief description of the social media analytics use case scenario

- at least three different on-premise or cloud tools / solutions identified and reasoning for the same
- short explanation about each tool / solution - how it can be used for social media analytics
- justification about the comparison parameters and relevant detailing
- a recommendation of the platform / tool for the media company use case

[03 Marks]

Ans 1

Usecase – Social Media Analytics

In recent times due to cheap internet and smartphone usage, social media has become extremely popular and is used by millions of users. Social media has therefore emerged as powerful tool providing data insights which otherwise was exceedingly difficult to get using means of surveys, or program participations. Most important of social media includes Facebook, YouTube, LinkedIn, Twitter, and others. Among all social media platforms, Twitter is big source of information which can be publicly read. Twitter provides techniques to read these messages published by user and supports way to label the message using "Hashtag", which thereby allows to see similar messages grouped by "Hashtag". The popular "Hashtag" is known as Trending Topics. A tweet consists of 140chars, which forces users to summarize the information which they want to share. The tweet of trending topics can be harnessed to carry out sentimental analysis and process further.

As news media company, we can harness the trending topics which are relevant to readers thereby providing advantage over competitors. The trending topics can be harnessed using Twitter's Trending Topics and carry out sentiment analysis on the same to fetch the reaction of users to news.

Platform Options

With wide adoption of cloud platform, more and more cloud providers are adding analytics services to the platform which will make business to focus on the business activities while the infrastructure and other service configuration is abstracted. This implies faster go-to market and low seed cost for business to experiment and implement business solution. With scalability at hand, it becomes easier for business to scale in case of increase workloads without mush hassle and scale down when workloads are not required thereby saving total cost of ownership (TCO).

On-premises options will always be there which means increased security parameters, but at cost of time to market and high initial cost and scaling of infrastructure adds to the cost.

Summary of cloud & on-premises options for building streaming analytics platform.

Generalized Stream Analytics Arch. Tier	Azure Cloud Platform	AWS Cloud Platform	Open Source (On Premise)
Collection Tier *This tier collects **tweets** from Twitter*	**Azure Databricks** is a fully managed Apache Spark environment with the global scale and availability of Azure. Clusters are set up, configured and fine-tuned to ensure reliability and performance without the need for monitoring. The instances can be auto scaled and auto terminated to improve total cost of ownership (TCO). Databricks supports data pipeline for streaming real-time data / events into Kafka or Event Hub, and machine learning for model training, feature engineering.	**AWS Kinesis** enables to ingest, buffer, and process streaming data in real-time. Amazon Kinesis is fully managed and runs streaming applications without requiring managing any infrastructure. Amazon Kinesis can handle any amount of streaming data and process data from hundreds of thousands of sources with extremely low latencies. It consists of services for data stream and data firehose.	**Apache Flume** can be used to store the twitter data into event / database store for batch processing. Apache Flume is a tool/service/data ingestion mechanism for collecting aggregating and transporting large amounts of streaming data such as log files, events (etc...) from various sources to a centralized data store. We can use experimental twitter source provided by Apache Flume to stream the twitter data, using memory channel buffer these tweets and HDFS sink to push these tweets into the Hadoop File System (HDFS). From HDFS we can carry out Batch Stream Processing using Apache Hadoop and Apache Spark (map reduce tasks)
Dataflow Tier *This tier ingests tweets which will be then used by Analytics Tier. It decouples production of events stream from its consumption*	**Azure Event Hub** is a big data streaming platform and event ingestion service. It can receive and process millions of events per second. Event Hubs provides a distributed stream processing platform with low latency and seamless integration, with data and analytics services. Event Hubs represents the "front door" for an event pipeline, often called an event ingestor in solution architectures, thereby decouples the production of an event stream from the consumption of those events. Event Hub provides a time retention buffer, decoupling event producers from event consumers.	**AWS Managed Service for Kafka (MSK),** is a fully managed service that makes it easy to build and run applications that use Apache Kafka to process streaming data. Apache Kafka is an open-source platform for building real-time streaming data pipelines and applications. Amazon MSK uses native Apache Kafka APIs to populate data lakes, stream changes to and from databases, and power machine learning and analytics applications.	**Apache Kafka** is a community distributed event streaming platform capable of handling trillions of events a day. Initially conceived as a messaging queue, Kafka is based on an abstraction of a distributed commit log. Since being created and open sourced by LinkedIn in 2011, Kafka has quickly evolved from messaging queue to a full-fledged event streaming platform.

Generalized Stream Analytics Arch. Tier	Azure Cloud Platform	AWS Cloud Platform	Open Source (On Premise)
Analytics Tier *This tier runs required analytics on the data / tweets from data flow tier using SQL / ML jobs*	**Azure Stream Analytics** is a real-time analytics and complex event-processing engine that is designed to analyze and process high volumes of fast streaming data from multiple sources simultaneously. Patterns and relationships can be identified in information extracted from several input sources including devices, sensors, clickstreams, social media feeds, and applications. These patterns can be used to trigger actions and initiate workflows such as creating alerts, feeding information to a reporting tool, or storing transformed data for later use. An Azure Stream Analytics job consists of an input, query, and an output. Stream Analytics ingests data from Azure Event Hubs. The query, which is based on SQL query language, can be used to easily filter, sort, aggregate, and join streaming data over a period. event ordering options and duration of time windows can be configured while performing aggregation	**Amazon EMR** is the industry-leading cloud big data platform for processing vast amounts of data using open-source tools such as Apache Spark, Apache Hive, Apache HBase, Apache Flink, Apache Hudi, and Presto. Amazon EMR makes it easy to set up, operate, and scale big data environments by automating time-consuming tasks like provisioning capacity and tuning clusters. EMR can run petabyte-scale analysis at less than half of the cost of traditional on-premises solutions and over 3x faster than standard Apache Spark. Workloads can run on Amazon EC2 instances, on Amazon Elastic Kubernetes Service (EKS) clusters, or on-premises using EMR on AWS Outposts.	**Apache Storm**: is based on the idea of streaming computation, i.e., processing new data individually. It is distributed and it uses Apache Zookeeper for this task. Storm provides the primitives for transforming a stream into a new stream in a distributed and reliable way. The basic primitives Storm provides for doing stream transformations are "spouts" and "bolts". For example, a spout may connect to the Twitter API and emit a stream of tweets. A bolt consumes any number of inputs streams, does some processing, and possibly emits new streams. **Apache Spark Streaming** is an extension of the core Spark API that enables scalable, high-throughput, fault-tolerant stream processing of live data streams. Data can be ingested from many sources like Twitter and can be processed using complex algorithms expressed with high-level functions like map, reduce, join and window. Finally, processed data can be pushed out to filesystems, databases, and live dashboards. It's also possible to apply Spark's machine learning and graph processing algorithms on data streams. **Apache Flink** is a framework and distributed processing engine for stateful computations over unbounded and bounded data streams. Flink has been designed to run in all common cluster environments perform computations at in-memory speed and at any scale. Flink Streaming comes with a built-in TwitterSource class for establishing a connection to the Twitter Streaming API **Kafka Streams** is a lightweight library.

Generalized Stream Analytics Arch. Tier	Azure Cloud Platform	AWS Cloud Platform	Open Source (On Premise)
			It is useful for streaming data from Kafka , doing transformation and then sending back to Kafka. We can understand it as a library like Java Executor Service Thread pool, but with inbuilt support for Kafka. It can be integrated well with any application and will work out of the box. One major advantage of Kafka Streams is that its processing is Exactly Once end to end. Tightly coupled with Kafka, cannot use without existing Kafka implementation
Deliver *This tier stores the data for consumption by the client application / user of stream analytics solution*	**Azure SQL Database** is fully managed SQL database automates updates, provisioning, and backups so we can focus on application development Flexible and responsive serverless compute and Hyperscale storage rapidly adapt to changing requirements. Layers of protection, built-in controls, and intelligent threat detection keep data secure. Built-in AI and built-in high availability maintain peak performance and durability with an SLA of up to 99.995 percent.	**Amazon Relational Database Service (Amazon RDS)** makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity while automating time-consuming administration tasks such as hardware provisioning, database setup, patching and backups. It frees one to focus on applications so one can give them the fast performance, high availability, security and compatibility they need. Amazon RDS is available on several database instance types - optimized for memory, performance or I/O - and provides with six familiar database engines to choose from, including Amazon Aurora, PostgreSQL, MySQL, MariaDB, Oracle Database, and SQL Server.	**MongoDB** is a general purpose, document-based, distributed database built for modern application developers and for the cloud era. **Apache Cassandra** is an open-source NoSQL distributed database trusted by thousands of companies for scalability and high availability without compromising performance. Linear scalability and proven fault-tolerance on commodity hardware or cloud infrastructure make it the perfect platform for mission-critical data.
Analytics	**Azure Machine Learning (Azure ML)** is a cloud-based	**Amazon SageMaker** helps data scientists and developers	Write Custom code using Python or equivalent language to implement basic

Generalized Stream Analytics Arch. Tier	Azure Cloud Platform	AWS Cloud Platform	Open Source (On Premise)
Tier	service for creating and managing machine learning solutions. It's designed to help data scientists and machine learning engineers leverage their existing data processing and model development skills & frameworks.	to prepare, build, train, and deploy high-quality machine learning (ML) models quickly by bringing together a broad set of capabilities purpose-built for ML.	model

Comparison Parameters for Stream Analytics Tool

Delivery Guarantees : It can be either Atleast-once (will be processed atleast one time even in case of failures) , Atmost-once (may not be processed in case of failures) or Exactly-once (will be processed one and exactly one time even in case of failures) . Exactly-once is desirable but is hard to achieve in distributed systems and comes in tradeoffs with performance.

Fault Tolerance : In case of failures like node failures, network failures, framework should be able to recover and should start processing again from the point where it left. This is achieved through checkpointing the state of streaming to some persistent storage from time to time. e.g., checkpointing kafka offsets to zookeeper after getting record from Kafka and processing it.

State Management : In case of stateful processing requirements where we need to maintain some state (e.g., counts of each distinct word seen in records), framework should be able to provide some mechanism to preserve and update state information.

Performance : This includes latency(how soon a record can be processed), throughput (records processed/second) and scalability. Latency should be as minimum as possible while throughput should be as much as possible. It is difficult to get both at same time.

Other advance features include Maturity / proven framework, event time processing, watermarks, windowing.

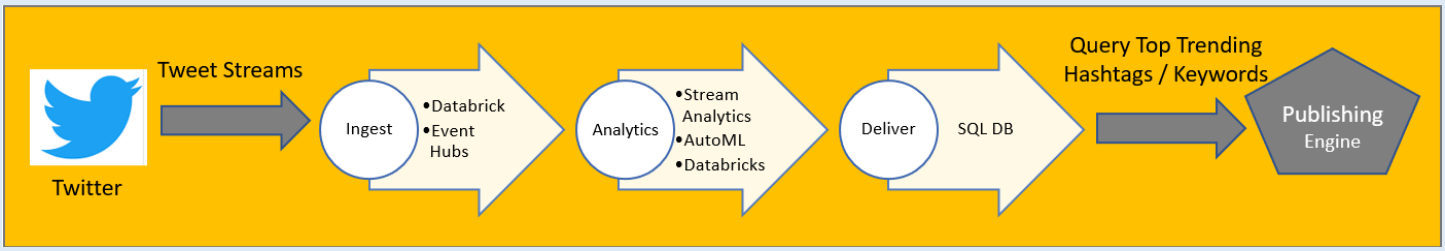
Processing Type

Naïve processing of events as soon as it arrives, without waiting for others. This is desirable when we need to process events without latency but makes fault tolerance a bit difficult. Example: Storm, Flink, Kafka Streams, Samza

Micro-batching or Fast batching wait for few seconds for incoming events and process them as single mini batch. In this implementation fault tolerance is easily achieved, but at cost of few secs' latency. Example: Spark streaming, Storm (Trident)

Stream Analytics Platform Recommendation for "AdMagic"

It is prudent that we want to explore the social media analytics, and see its benefit for business, and we want to do it as soon as possible. Our recommendation is to go for **Azure / Cloud based offering**. High level solution will be as follows:



Why?

1. **Fully managed**, Azure Event Hub & Stream Analytics are fully managed Platform-as-a-Service (PaaS) with little configuration or management overhead, so we can focus on business solutions.
2. **Programmer productivity**, with developer productivity tools to complement the services it makes implementing the business requirements very easy.
3. **Low total cost of ownership**, as cloud service, Event Hub and Stream Analytics are optimized for cost. There are no upfront costs involved - we only pay for the streaming units we consume. There is no commitment or cluster provisioning required, and we can scale the job up or down based on business needs.
4. **Mission-critical ready**, the services are available across multiple regions worldwide and is designed to run mission-critical workloads by ***supporting reliability, security***, and ***compliance requirements***

Note: While TCO is low, we need to evaluate the running cost over business benefit over time.

Q2. You are in a meeting with the firm's management who are little bit concerned about the capabilities associated with social media analytics tools discussed in question 1. The client it bit hesitant to rely on the tools for these analytics. In order to assist the client

- Briefly narrate the at least five key capabilities of the tool / solution that you have recommended.
- Address how each of this key capability can be leveraged for the use case identified in part 1

[1.5 Marks]

Ans 2

Key capabilities of using Azure (or *any* Cloud) Stream Analytics Platform

Azure Stream Analytics services are designed to be easy to use, flexible, reliable, and scalable to any job size. It is available across multiple Azure regions.

Ease of getting started, Azure Stream services are easy to start. It only takes a few clicks to connect to multiple sources and sinks, creating an end-to-end pipeline. Stream Analytics can connect to Azure Event Hubs for streaming data ingestion. Stream Analytics can route job output to many storage systems including Azure Data Lake Store.

Programmer productivity, Azure Stream Analytics uses a SQL query language that has been augmented with powerful temporal constraints to analyze data in motion. The jobs can be created using developer tools like Azure PowerShell, Azure CLI, Stream Analytics Visual Studio tools, the Stream Analytics Visual Studio Code

extension, or Azure Resource Manager templates. Using developer tools allows to develop transformation queries offline and use the CI/CD pipeline to submit jobs to Azure. The Stream Analytics query language supports simple data manipulation, aggregation and analytics functions, geospatial functions, pattern matching and anomaly detection.

Fully managed, Azure services are fully managed (PaaS) offering on Azure. There is no need to provision any hardware or infrastructure, update OS or software. Azure Stream Analytics fully manages the job, so we can focus on business logic and not on the infrastructure.

Low total cost of ownership, as a cloud service, services are optimized for cost. There are no upfront costs involved - we only pay for the streaming units consumed. There is no commitment or cluster provisioning required, and we can scale the job up or down based on business needs.

Mission-critical ready, Azure services are available across multiple regions worldwide and is designed to run mission-critical workloads by supporting reliability, security, and compliance requirements.

- ***Reliability***, Azure Stream Analytics guarantees exactly once event processing and at-least-once delivery of events, so events are never lost. Azure services have built-in recovery capabilities in case the delivery of an event fails. Stream Analytics also provides built-in checkpoints to maintain the state of job and provides repeatable results. As a managed service, Stream Analytics guarantees event processing with a 99.9% availability at a minute level of granularity.

- ***Security***, Azure Stream Analytics encrypts all incoming and outgoing communications and supports TLS 1.2. Built-in checkpoints are also encrypted. Stream Analytics doesn't store the incoming data since all processing is done in-memory.

- ***Compliance***, Azure Stream Analytics follows multiple compliance certifications

High Level Approach

1. Tweet streams will be collected / ingested using Databricks and tweets will be ingested to Azure Event Hub
2. From Event Hub the tweets will be read by the Azure Streams Analytics via Jobs
3. Azure streams analytics supports a simple, declarative query model. Using this we will query the incoming tweet streams using different window - tumbling window / hopping window / sliding window / session window i.e., get count of tweets per every N secs.
4. The tweet (data) transformation (sentimental analysis) will be done using AutoML or Databricks services. We will prefer using databricks as we are already using this service to collect tweets thereby saving cost on using addition AutoML service.
5. The final results of positive / influence tweets / topics will be stored into database.
6. The publishing engine can query the database and prioritize the news feeds accordingly.

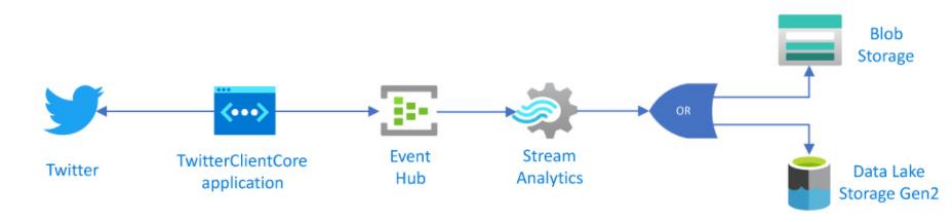
Q3. The blog discusses use case which the media company has addressed in space of social media analytics. But the solution is described in terms of various cloud services offered by Microsoft Azure. The client does not have the knowledge about the cloud computing and Azure. In fact, all the use cases can be very well addressed with a general architecture used in the big data analytics and streaming analytics. You need to work upon helping client to understand those common architectures.

- Identify the architecture that can be fitted well for capturing the use cases.
- Prepare an architecture diagram based upon your answer to earlier question.
- Take care that use cases should be vividly coming out of the architecture diagram, if required add brief description about each flow

[02 Marks]

Ans 3

Reference architecture described in Microsoft blog is as follows:



A generic architecture for streams processing includes following components,

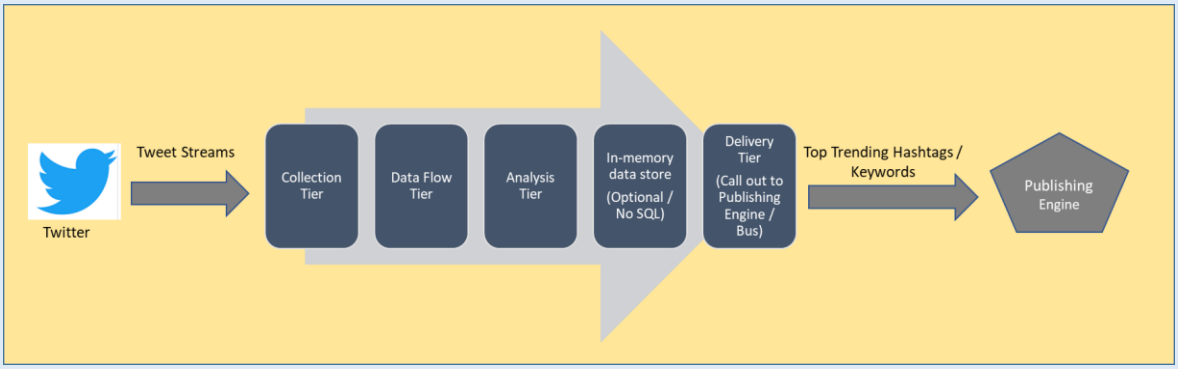
Collection Tier : This is responsible for connecting with the external services / streams and get the data into tool for real-time processing (*this is represented as TwitterClientCoreApplicaiton in MS Blog*)

Data Flow : Moves data from collection tier and channels it to Analytics Tool (*this is represented as EvenHub in MS Blog)*

Analytics Tier : This tier processes the data stream and perform various tasks like filtering, cleansing, machine learning, grouping, statistical etc. (*this is represented as Stream Analytics in MS Blog*)

In-memory data store : [**optional**], and someitmes this is used to cache the output before pushing this to persistent tier

Delivery Tier : This pushes the data to required output format which can be persistent database OR event bus or API callout. This will differ use-case to use-case. (*this is represented as "Data Lake OR Blob" in MS Blog*)



High Level Architecture of Proposed Solution (mapping proposed solution to Generic SPA Architecture)

The Azure platform services will be used to build the real-time stream analytics solution. From twitter the tweets will be collected / stream via databricks data stream services, in the same databricks we will carry out sentimental analyis. The

output along with topic / hashtag will be moved to Azure Event Hub. From Event Hub the data will be picked by Azue Stream Analytics Job, and it will run Query using Sliding Window to get count of tweet topics / hashtag along with its sentiments every N secs, this data will be stored into Azue SQL database for further query by Publishing Engine. We can also publish the data directly to pusblishing engnie without storing into database.

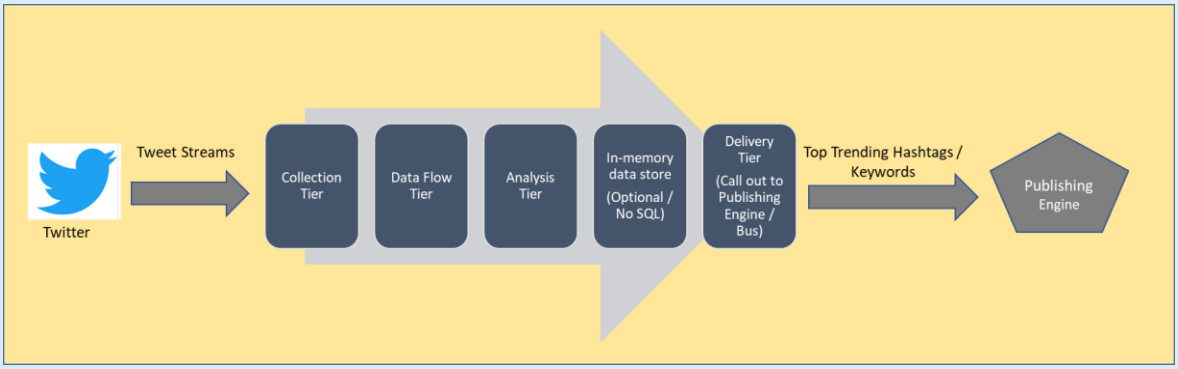
Q4. The client is now impressed with the capabilities of the Microsoft Azure and how it's streamlining the application development and deployment. But they also want to discover more on the open-source tools / platforms that can be leveraged. As a result, you need to work upon identifying the open-source tools for the use case.

- Identify the tools / platforms that can be used to solve it.
- Draw a solution diagram using the tools identified in earlier question the flow should come out clearly from the solution diagram.

[3.5 Marks]

Ans 4

Open-Source Tools which can be utilized for building real-time streaming solution.



Collection + Dataflow Tier

Apache Flume can be used to store the twitter data into event / database store for batch processing. Apache Flume is a tool/service/data ingestion mechanism for collecting aggregating and transporting large amounts of streaming data such as log files, events (etc...) from various sources to a centralized data store. We can use experimental twitter source provided by Apache Flume to stream the twitter data, using memory channel buffer these tweets and HDFS sink to push these tweets into the Hadoop File System (HDFS). From HDFS we can carry out Batch Stream Processing using Apache Hadoop and Apache Spark (map reduce tasks)

Apache Kafka is a community distributed event streaming platform capable of handling trillions of events a day. Initially conceived as a messaging queue, Kafka is based on an abstraction of a distributed commit log. Since being created and open sourced by LinkedIn in 2011, Kafka has quickly evolved from messaging queue to a full-fledged event streaming platform.

Analytics Tier

Apache Storm: is based on the idea of streaming computation, i.e., processing new data individually. It provides support for micro-batching operations, collecting new data in small data set and then executing operations on the whole dataset with the Trident API. It's distributed and it uses Apache Zookeeper for this task. The core abstraction in Storm is the "stream". A stream is an unbounded sequence of tuples. Storm provides the primitives for transforming a stream into a new stream in a distributed and reliable way. For example, transform a stream of tweets into a stream of trending topics. The basic primitives Storm provides for doing stream transformations are "spouts" and "bolts". Spouts and bolts have interfaces that implement to run application-specific logic. A spout is a source of streams. For example, a spout may connect to the Twitter API and emit a stream of tweets. A bolt consumes any number of inputs streams, does some processing, and possibly emits new streams. Complex stream transformations, like computing a stream of trending topics from a stream of tweets, require multiple steps and thus multiple bolts. Bolts can do anything from run functions, filter tuples, do streaming aggregations, do streaming joins, talk to databases, and more.

Apache Spark Streaming is an extension of the core Spark API that enables scalable, high-throughput, fault-tolerant stream processing of live data streams. Data can be ingested from many sources like Twitter and can be processed using complex algorithms expressed with high-level functions like map, reduce, join and window. Finally, processed data can be pushed out to filesystems, databases, and live dashboards. It's also possible to apply Spark's machine learning and graph processing algorithms on data streams.

Apache Flink is a framework and distributed processing engine for stateful computations over unbounded and bounded data streams. Flink has been designed to run in all common cluster environments perform computations at in-memory speed and at any scale. Flink Streaming comes with a built-in TwitterSource class for establishing a connection to the Twitter Streaming API.

Kafka Streams is a lightweight library. It is useful for streaming data from Kafka , doing transformation and then sending back to Kafka. We can understand it as a library like Java Executor Service Thread pool, but with inbuilt support for Kafka. It can be integrated well with any application and will work out of the box. One major advantage of Kafka Streams is that its processing is Exactly Once end to end. Tightly coupled with Kafka, cannot use without Kafka in picture.

Persistence Tier

Redis / Memcached are in-memory data store tier. They are simple key-value store, they provide for high order data structures, and useful for streaming analysis application. Although they are not natively distributed, as it focuses on high performance.

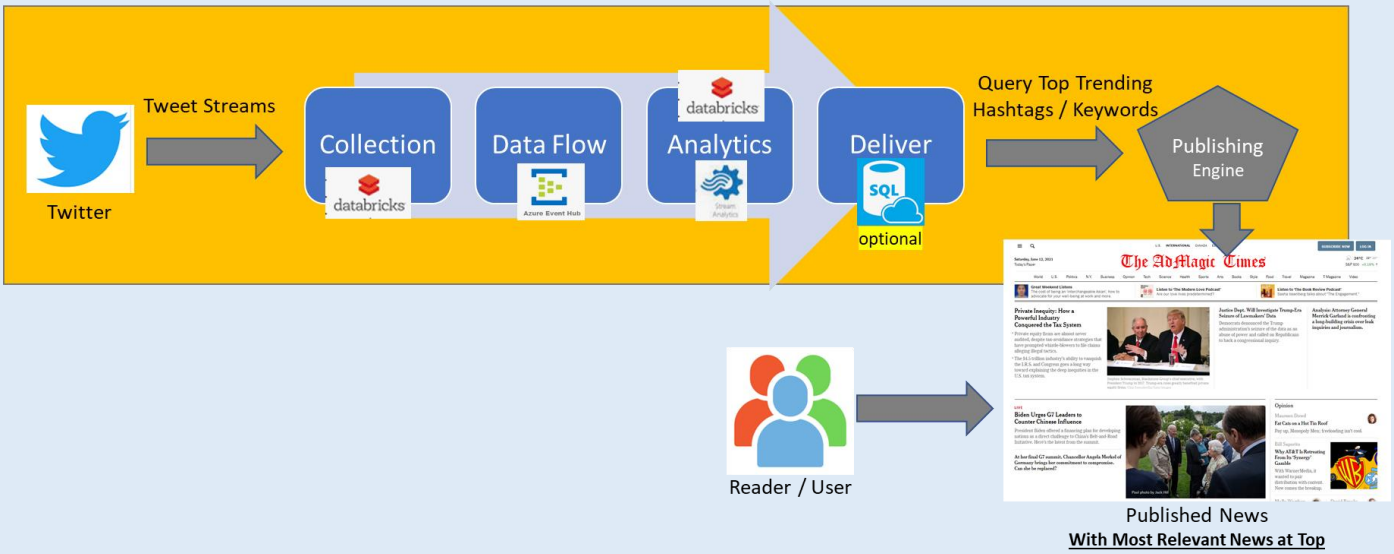
Mongo DB is no-sql document store. It is schema-less and has proven popular for applications that maintain rich profiles or data that can be naturally ordered into documents. It supports master-slave replication as well as

sharding (partitioning). MongoDB is a general purpose, document-based, distributed database built for modern application developers and for the cloud era.

Apache Cassandra is an open-source NoSQL distributed database trusted by thousands of companies for scalability and high availability without compromising performance. Linear scalability and proven fault-tolerance on commodity hardware or cloud infrastructure make it the perfect platform for mission-critical data. Cassandra is a decentralized database system that takes features from both key-value stores and tabular databases using elements from both Amazon's DynamoDB as well as Google's BigTable.

Recommended solution using Azure (cloud) services.

This will help "AdMagic" to explore the business benefit much fast with minimum initial cost. If "AdMagic" decides to continue with the solution, it can be scaled up or down due to cloud architecture benefit.

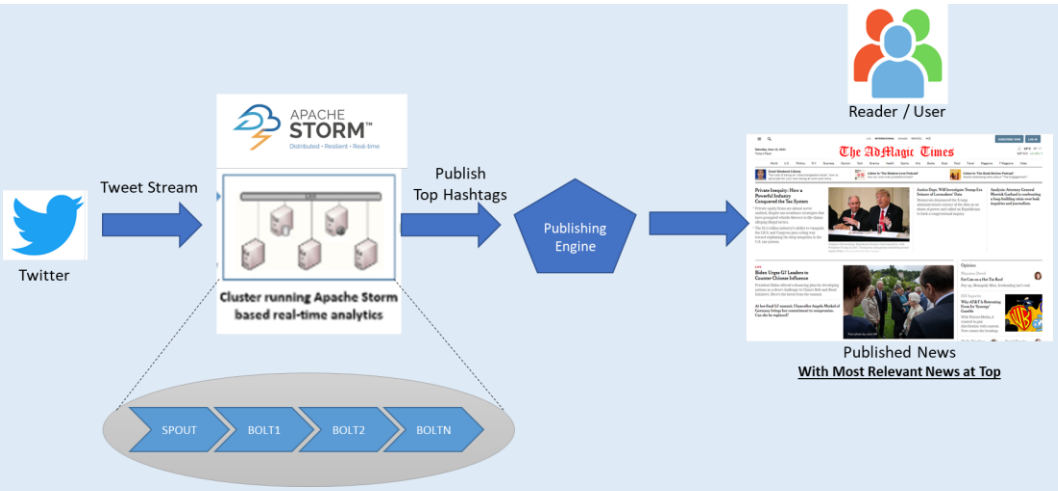


The Azure platform services will be used to build the real-time stream analytics solution. From twitter the tweets will be collected / stream via databricks data stream services, in the same databricks we will carry out sentimental analyis. The output along with topic / hashtag will be moved to Azure Event Hub. From Event Hub the data will be picked by Azue Stream Analytics Job, and it will run Query using Sliding Window to get count of tweet topics / hashtag along with its sentiments every N secs, this data will be stored into Azue SQL database for further query by Publishing Engine. We can also publish the data directly to pusblishing engnie without storing into database.

Appendix

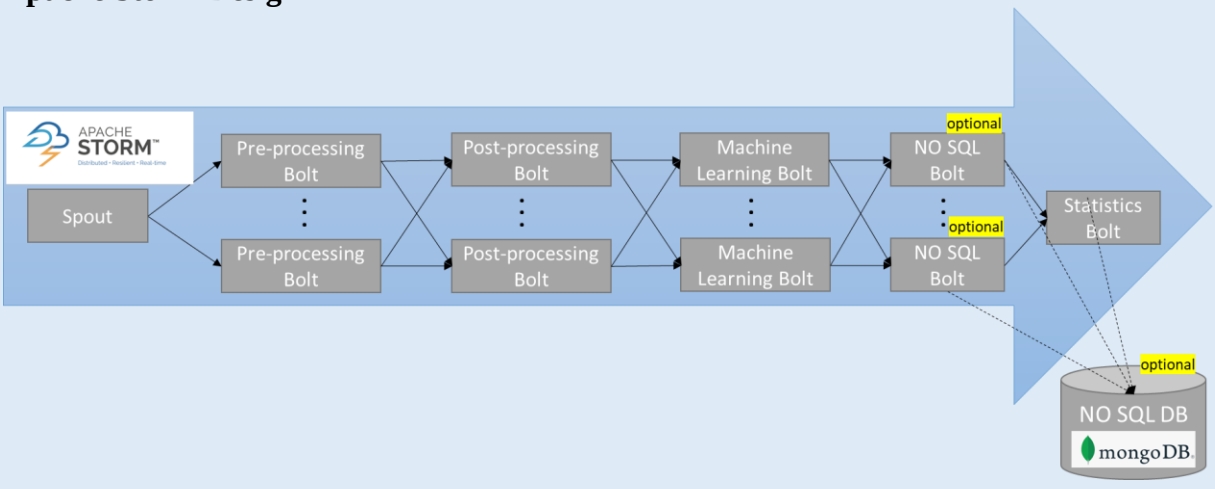
Solution Architecture (using opensource tools)

We explored opensource tools option, but we are not recommending the same for "AdMagic" as this will need more time, resources and funding. But this will help understand how the real-time streaming application can be built using opensource tools. This will help incase we want to have on-premises solution in case of any eventuality, and this can be built over time.



1.Apache Storm will subscribe to the twitter real-time stream, these tweets will be streamed using Spout and Bolt configuration
2. Using configuration of Bolt Jobs the tweets will cleansed, filtered based on required topics / keywords, run machine learning / classification, and finally compute statistics to arrive at top N keywords or topics or hashtags
3. [optional] If required we can store the data into no-sql / sql DB. This may serve utility for batch processing of these tweets to arrive at some further deep analysis (future)
4. Finally the output of the pipeline which is top N keywords / hashtags will be published to the publishing engine via API call or Event Bus, which will finally publish the relevant news article to the news site for readers / users reading.

Apache Storm Design



****

Group 142
SURESH BABASAHEB NIMBALKAR (2019HC04104)
WAVHAL HEMANT SUDHIR (2019HC04093)
SURAJ KUMAR (2019HC04912)