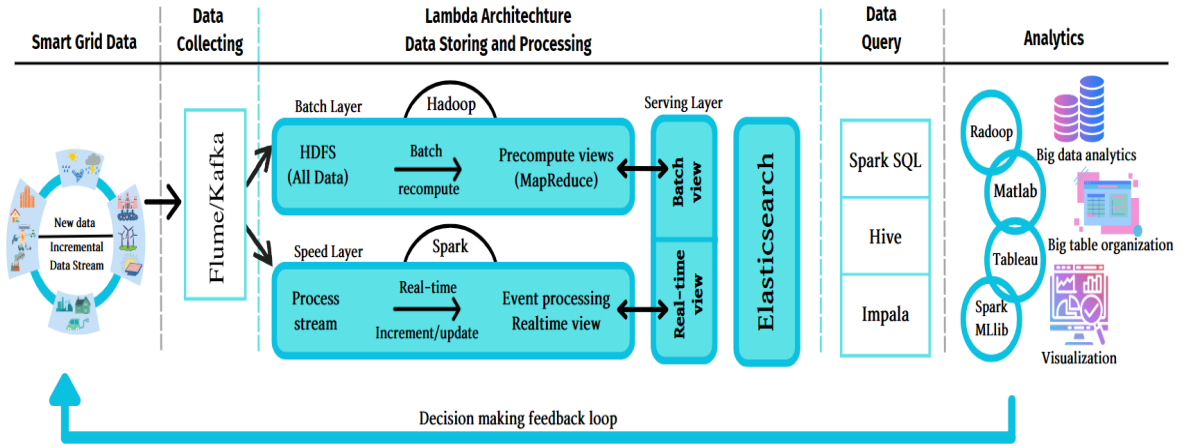
Capstone Project

1. Introduction

Requirement: Establishing a comprehensive big data processing workflow, encompassing data crawling, storage, analysis, and visualization components.



System architecture

All student groups are required to follow the below infrastructure provided by Centic company (the resources will be provided later by the next announcements):

- Data Collecting: Kafka
- Database: Google bucket and Elasticsearch
- Data query and process: Spark
- Analytics: Kibana

Note: you are advised using Airflow to manage and schedule data engineering pipelines.

Checkpoints

Checkpoint 1 (9^{th} week): Data Crawling

- You have the option of 500MB for text data or 3GB for image data.
- The data has been processed to eliminate duplicates.

Checkpoint 2 (11^{th} week): Data Processing and Analysis

- Data is cleaned and analyzed to extract insights.
- Both raw and processed data should be stored in a Google bucket.

Checkpoint 3 (14^{th} week): Architecture Deployment and Data Visualization

- The pipelines will be deployed on the Centic's infrastructure and should run flawlessly.
- You will also present visualizations or demonstrations.

Topic 1 - Crawl, process, and analyze web3 social data

Social media and messaging platforms are increasingly becoming a pivotal component of applications within the Web3 landscape. These platforms are host to substantial user communities, particularly including the technologically adept and those with investment expertise, and they serve as repositories for a massive volume of information. This serves to:

- Projects: Disseminate updated information to users.
- Key Opinion Leaders (KoLs): Assess and endorse project-related information.
- Professors: Contribute knowledge and informed opinions.
- End Users: Seek and obtain project-related information.

Numerous projects have been executed to harness these data sources, which encompass:

- Quality assessment of social channels for Web3 projects (e.g., Lunar Crush).
- Provision of marketing strategy recommendations for Web3 projects, involving the identification of suitable Key Opinion Leaders (KoLs) and communities (e.g., Addressable).
- Etc.

Nonetheless, this constitutes a vast data domain, and a multitude of challenges remain associated with this data source, including:

- The prevalence of numerous bots leads to a high frequency and substantial volume of low-quality data.
- A continuous and rapid surge in data volume necessitates the development of corresponding mechanisms for collection, clearning, and processing.
- The absence of a perfect mechanism for mapping social data and blockchain data.

Problem 1: Evaluate users by geographical area.

Functional requirements:

- Categorize users by geographical area
- Identify users relating to Web3
- Classify customers according to projects they are interested in, and online interaction frequency, ...
- Data visualization systems for these problems.

Implementation:

- Collect data on social media like Twitter
- Store data on the NoSQL database
- Conduct user classification
- Visualize data and classification results.

Problem 2: Combined with blockchain data, build a social information suggestion system for users in the portfolio management application.

Portfolio management is an application designed to empower users in effectively overseeing their investment portfolios. Within this application, a feature is integrated to provide valuable project recommendations and timely alerts regarding critical updates.

For example, Wallet A holds AAVE tokens - a token of a lending protocol project. In portfolio management, wallet A should have some information below:

- Real-time updates from the AAVE team via Twitter.
- Exclusive insights into lending trends.
- …

However, because a wallet usually holds a lot of tokens and joins many different projects, there are a varied number of possible suggestions. Consequently, the application necessitates a robust ranking mechanism capable of offering the most pertinent suggestions to users.

Functional requirements:

- Integrate social data from various projects with their respective applications.
- Establish connections between application users and relevant social data sources.
- Implement a robust system for categorizing, assessing, and prioritizing information based on users' portfolio investments, social network trends, significant project updates, and more.

Implementation:

- Associate social media accounts with their respective projects.
- Categorize projects based on type, quality, and other relevant factors.
- Continuously gather real-time data from social media posts.
- Apply classification and evaluation algorithms to determine the significance of project-related posts.
- Present the top three most relevant pieces of information within the application.
- Conduct an initial assessment of the outcomes.

Problem 3: System for labeling and evaluating the quality of accounts on Twitter.

Collected social data need to be monitored to evaluate the quantity and quality which enable us to do these things below:

- Establish an optimized storage mechanism to augment server performance.
- Log and promptly alert concerning any irregular or noteworthy events.

Functional requirements:

- Identify unusual and low-quality accounts.
- Visualize social data in terms of quantity and quality

Implementation:

- Gather data from Twitter, utilizing either the Twitter API or web scraping techniques.
- Evaluate project quality and Key Opinion Leaders (KoLs) based on key social network metrics, including posting frequency, impression counts, and engagement levels.
- Create visual representations of the analyzed data.

Topic 2 - Analyze on-chain data

On-chain data represents a fusion of data derived from a multitude of distinct blockchain networks, including Ethereum, BNB Chain, Polygon, and others, along with applications within those ecosystems such as lending platforms, decentralized exchanges (Dexes), and GameFi projects. It's noteworthy that a project can be deployed across various blockchain networks, and a user may concurrently possess multiple wallets, actively engaging in numerous projects.

Centic diligently monitors data emanating from seven blockchain networks, encompassing approximately 200 million unique wallet addresses.

Problem 4: Profiler

A wallet address is a lifeless sequence of characters that functions as a unique identifier for a user within blockchain networks. The profiler application is designed to present comprehensive information that delineates the user's personality, traits, and profile.

Functional requirements:

- Automatically categorize wallet addresses based on the associated project and the type of project in which they are involved. Furthermore, distinctive wallets,

such as those belonging to foundations (projects, exchanges, teams, etc.), should also be appropriately labeled.
- Identify associated wallets or those that impact a wallet address, including highly interactive wallets, whale wallets, and others.
- Link relevant social media accounts.

Problem 5: Alerts

When a wallet engages in the Web3 space, it holds assets or maintains positions in the applications it invests in. Consequently, they need to continuously and proactively update with crucial information. For instance, if the lending project in which they are involved experiences a security break or any significant developments, they should be promptly informed.

Functional Requirements:

- Detect high-risk or distinctive transactions.
- Issue alerts to pertinent users through the application or via a Telegram bot.