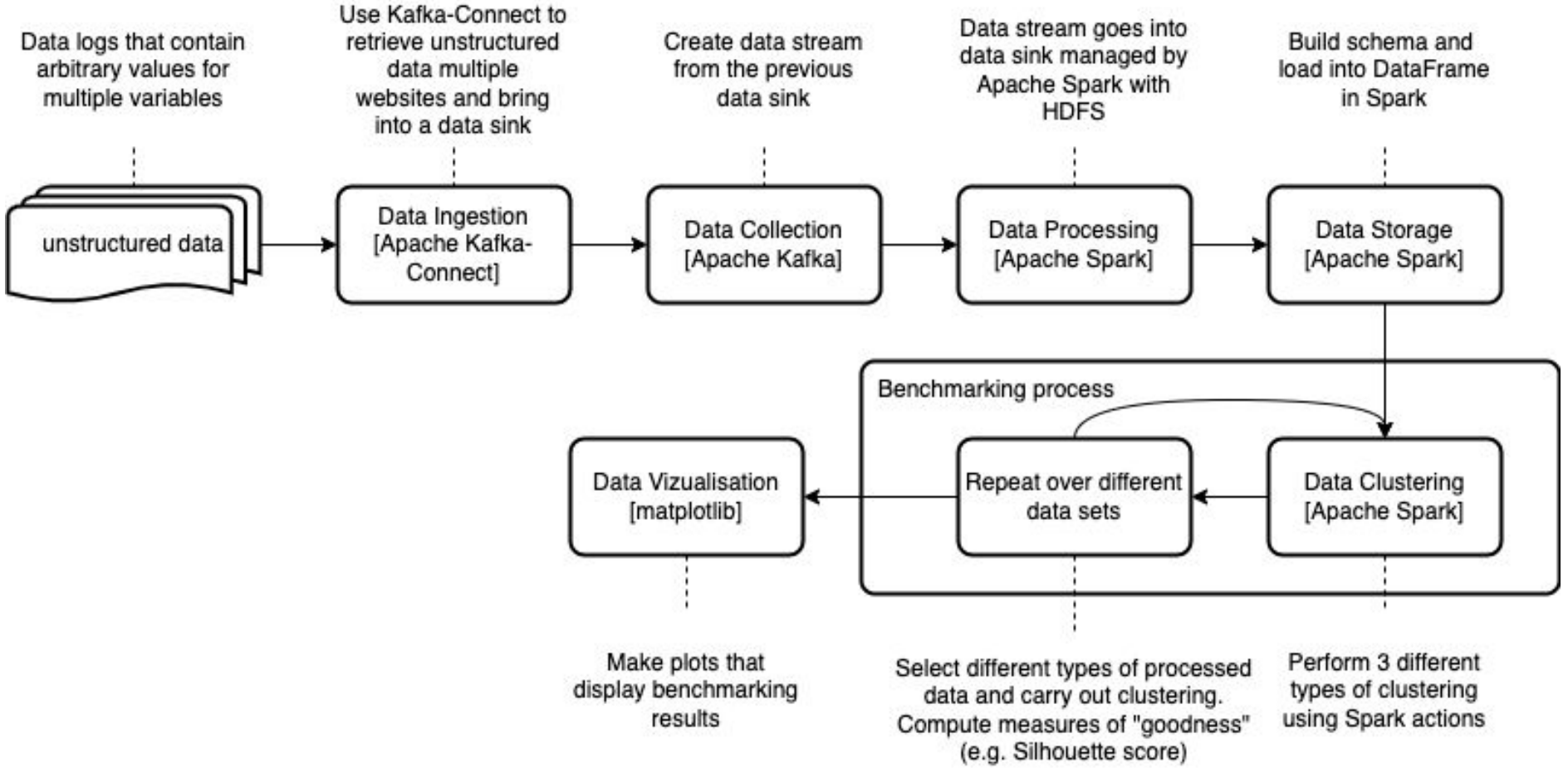
Basic Idea

- Create a clustering pipeline that handles structured/unstructured data
- Implement and compare different types of clustering algorithms, such as centroid-based (e.g. k-means) or connectivity-based (e.g. hierarchical clustering)
- Benchmark the algorithms using datasets with varying characteristics and determine their “scalable performance”
- Use a large-scale data processing engine like Apache Spark

Pipeline


Data logs that contain arbitrary values for multiple variables
Use Kafka-Connect to retrieve unstructured data multiple websites and bring into a data sink
Create data stream from the previous data sink
Data stream goes into data sink managed by Apache Spark with HDFS
Build schema and load into DataFrame in Spark
unstructured data
Data Ingestion [Apache Kafka-Connect]
Data Collection [Apache Kafka]
Data Processing [Apache Spark]
Data Storage [Apache Spark]
Benchmarking process
Data Vizualisation [matplotlib]
Repeat over different data sets
Data Clustering [Apache Spark]
Make plots that display benchmarking results
Select different types of processed data and carry out clustering. Compute measures of "goodness" (e.g. Silhouette score)
Perform 3 different types of clustering using Spark actions

Data Ingestion Frameworks

Apache Kafka

- Reliable stream data storage
- Scalability and message durability
- Kafka-Connect
- Limitation in message size
- Custom code required

Apache Flume

- Collect, aggregate, and transfer streaming events into Hadoop
- Many built-in sources
- Configuration based
- Limitation in message size
- Data loss scenarios

Apache NiFi

- Real-time control of data flow
- Dataflow management with visual control
- Arbitrary message size
- Configuration-based Web UI
- No data replication

Datasets

- Datasets with varying characteristics in order to test the applicability of different clustering techniques
- Data streams (e.g. real-time sensory data)
- Benchmark datasets of varying size (e.g. https://www.kaggle.com/currie32/crimes-in-chicago)
- Setup a data ingestion pipeline to collect streams of data(e.g. using Apache Kafka)

Algorithms

- Implement from scratch one or more clustering algorithms
- Centroid-based clustering (e.g. K-means)
- Density-based clustering (e.g. DBSSCAN)
- Distribution-based clustering (e.g. Gaussian mixture models)
- Connectivity-Based clustering (e.g. Hierarchical Clustering)