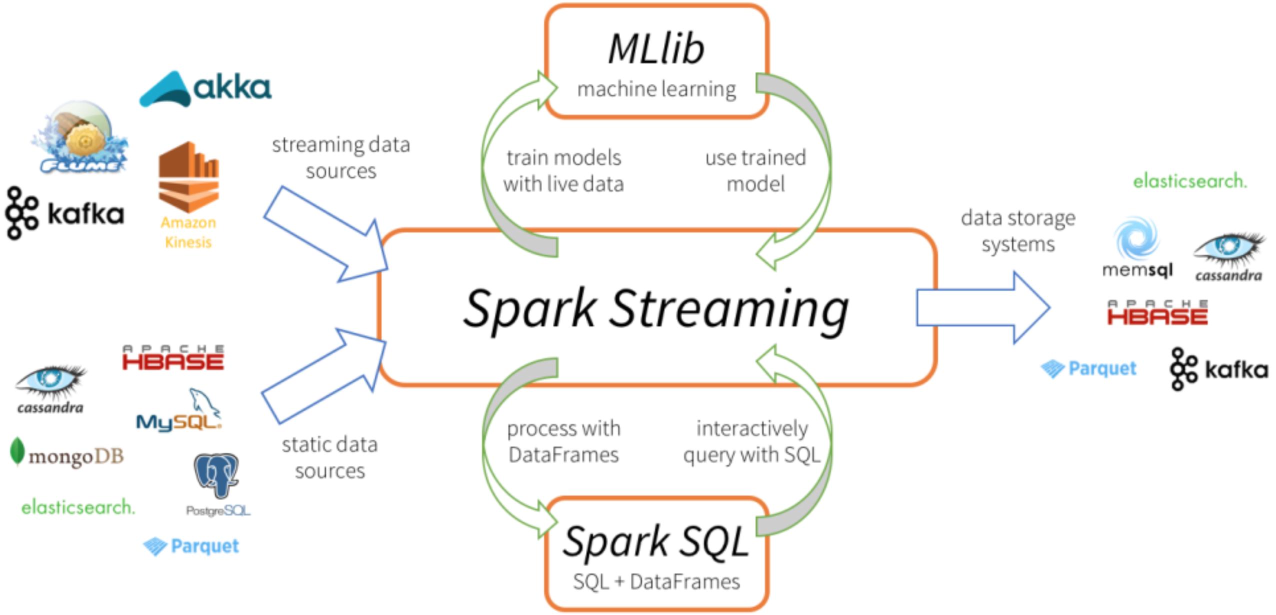
Processing of big data

spark

session-4



streaming data sources

static data sources

MLlib

machine learning

train models with live data

use trained model

Spark Streaming

process with DataFrames

interactively query with SQL

Spark SQL

SQL + DataFrames

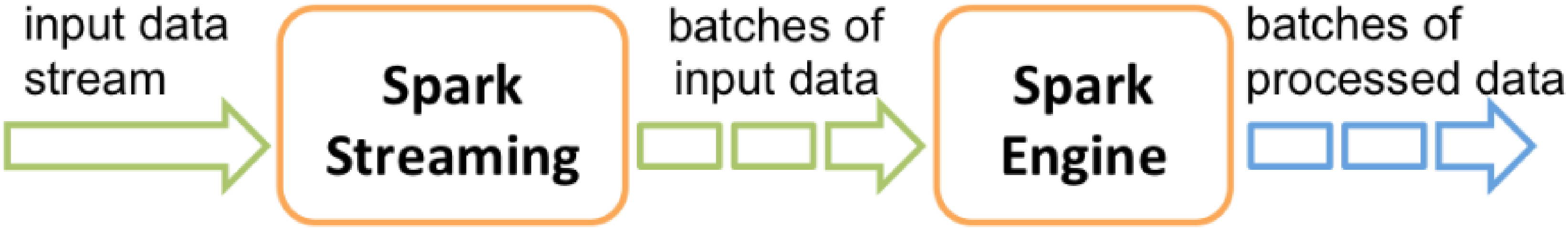
data storage systems

Agenda

- Unstructured Streaming
- Structured Streaming

Data Holder

- *Discretized stream* or ***DStream***



- Internally, a DStream is represented as a sequence of RDDs

- lines = ssc.socketTextStream(“localhost”, 9999)

DStream API

- map
- flatMap
- filter
- repartition
- union
- count
- countByValue
- reduceByKey
- join
- **updateStateByKey**
- **transform**

UpdateStateByKey

- Refer to the following code for better understanding:

https://github.com/apache/spark/blob/v2.4.5/examples/src/main/python/streaming/stateful_network_wordcount.py

Transform

- Allows arbitrary RDD-to-RDD functions to be applied on a DStream
- Used to apply any RDD operation that is not exposed in the DStream API
- E.g.: Join an RDD with DStream

Transform: Cont…

```
lines = ssc.socketTextStream(“localhost”, 9999)

#Generate key value pairs DStream
keyVal = lines.map(lambda x: (x.split(",")[0],x))

#Read product RDD in memory
productRdd = sc.textFile("product.csv")
product = productRdd.map(lambda x: (x.split(",")[0],x))

joined = keyVal.transform(lambda rdd: rdd.join(product))
```

Checkpoint

- Metadata: Saving of the information defining the streaming computation to fault-tolerant storage like HDFS
- Stores*: Configuration, DStream operations, Incomplete batches*

- Data: Saving of the generated RDDs to reliable storage
- Stores: intermediate RDDs of stateful transformations

Windowing operation

```
from pyspark.sql.functions import window



df.groupBy(window(df.timestamp,
          "2 minutes", "1 minutes"),
          product_id, name)
          .agg(_sum("qty"), _sum("amt"))
```

Unstructured Streaming

- Source
  - Socket: exercise 1
  - File: exercise 2
  - Kafka: exercise 3

- Sink
  - File: exercise 4
  - Database: exercise 5

Unstructured Streaming: Cont..

- ETL on streaming data
  - Transform operation: exercise 6
- Checkpoint for restartability
  - Metadata checkpoint: exercise 7
  - Data checkpoint: self study
- Social media feed processing
  - Twitter feed analysis: self study

Structured Streaming

- Source
    - Socket: exercise 8
    - File: exercise 9
    - Kafka: exercise 10

- Sink
    - File: exercise 11
    - Kafka: exercise 12, exercise 13 (self study)

Structured Streaming: Cont..

- ETL
  - Apply all possible transformation as you do on normal dataframe: exercise 14

- Window aggregation
  - Analyze aggregated data over a period of time:

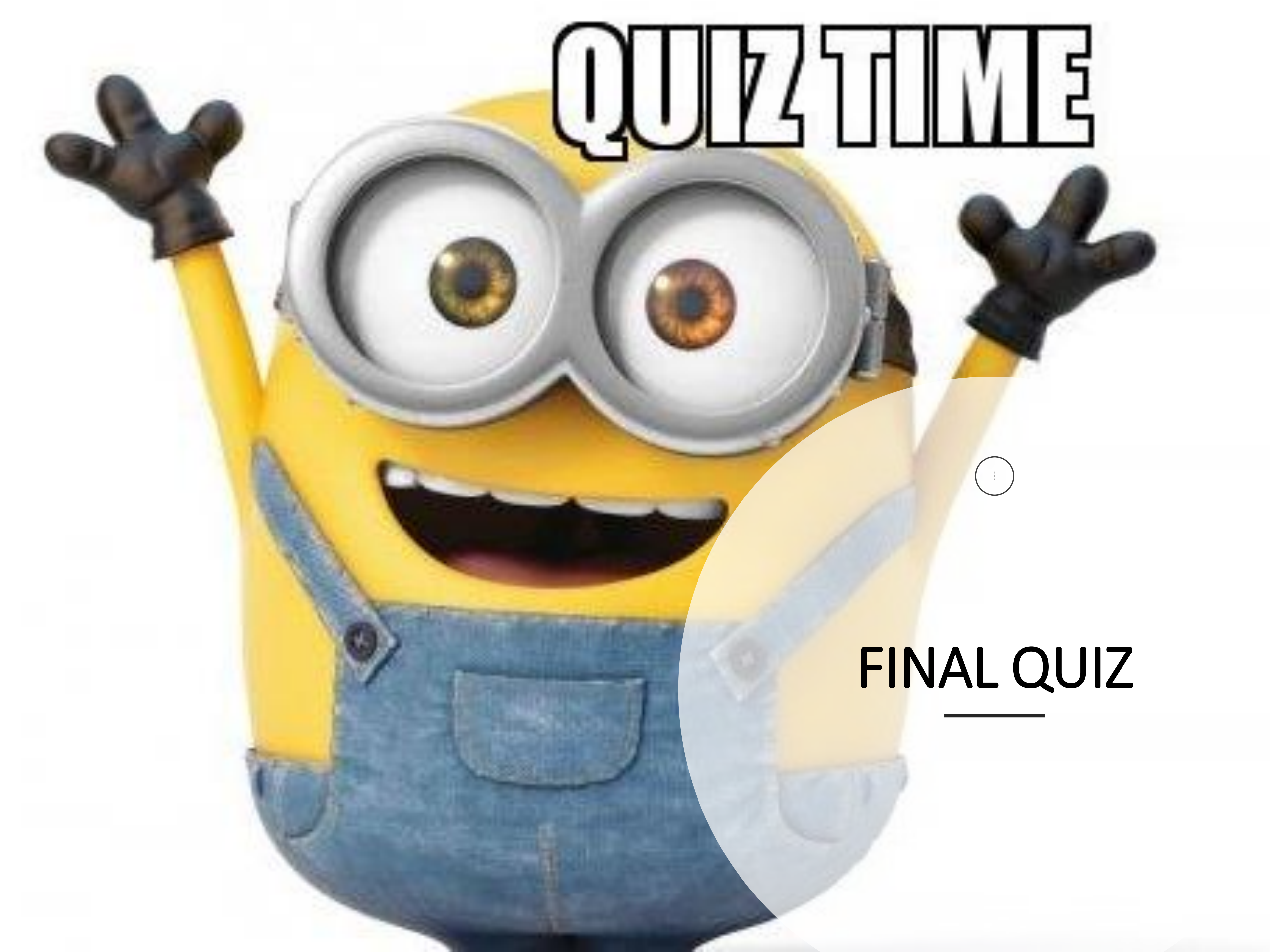
    exercise 15

Reference

- https://spark.apache.org/docs/latest/streaming-programming-guide.html

- https://spark.apache.org/docs/2.4.0/structured-streaming-kafka-integration.html

- https://spark.apache.org/docs/latest/structured-streaming-programming-guide.html



FINAL QUIZ

Q1

Three different ways to set driver memory

- Command line
- Command line conf
- Configuration file

Q2

Display formatted result of a dataframe on console without truncating any column value

- df.show(truncate=false)

Q3

Two major categories of machine learning algorithms. Also state an example of for each category.

- Supervised: Classification algo
- Unsupervised: Clustering algo

Q4

Method to identify relation among features

- Correlation Matrix

Q5

State two Components of ML pipeline

- Transformer: returns a dataframe
- Estimator: returns model

Q6

What does independence hypothesis indicate?

- Helps to identify if feature and class are independent

Q7

What are the different ways to pass external libraries with spark job?

- --package
- jars

Q8

What is the difference between MlLib and ML packages?

- MlLib: RDD based. In maintenance mode
- ML: Dataframe based. Future

Q9

What driver we had used to export data from spark to RDBMS?

- JDBC

Q10

Explain the command to join two dataframes

- df1.join(df2, “column name”)

Q11

Explain the command to join two RDD

- rdd1.join(rdd2)

Q12

By default save method of dataframe saves data in which format?

- parquet

Q13

Difference between JIT, JVM, JRE and JDK

Thank you!