

Project: Kafka + Spark Streaming + PySpark

CS570: Big-data Processing and Analytics

SFBU
Navya Kandimalla

Table Of Contents

- Introduction
- Design
- Implementation
- Test
- Enhancement Ideas
- Conclusion
- References

Introduction

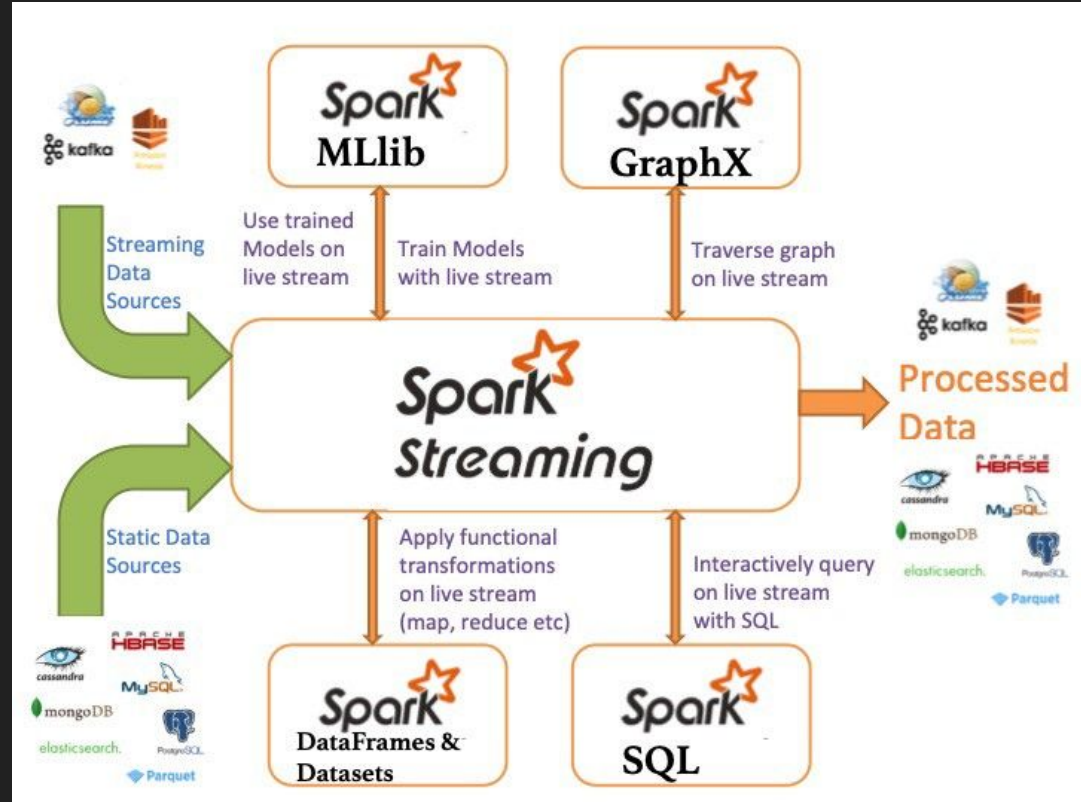
- Kafka is a distributed streaming platform that allows applications to publish and subscribe to streams of data.
- It is designed to be highly scalable and fault tolerant. Spark Streaming is an extension of the Apache Spark platform that enables real-time processing of streaming data.
- PySpark is the Python API for Apache Spark, which allows developers to write Python code to interact with the Spark platform.
- PySpark enables developers to create applications that can process streaming data in real-time, as well as batch data.

Design

- The design of a Kafka + Spark Streaming + PySpark system involves setting up a Kafka cluster to ingest streaming data, setting up a Spark cluster to process the data, and then using PySpark to write applications that can interact with the Spark cluster.
- The Kafka cluster can be configured to ingest data from various sources, such as web servers, databases, and message queues. The Spark cluster can then process the data in real-time, and the PySpark applications can be used to interact with

Design

The following diagram illustrates the design of a Kafka + Spark Streaming + PySpark system:



Implementation

Kafka QuickStar—Apache Kafka + Kafka-Python

1.The latest version of Kafka binary distribution is available at

<https://kafka.apache.org/downloads>.

Test

2. Starting Zookeeper. Unzip it, get into the folders cd into it

[GET STARTED](#)[DOCS](#)[POWERED BY](#)[COMMUNITY](#)[APACHE](#)[DOWNLOAD KAFKA](#)

Otherwise any version should work (2.13 is recommended).

Kafka 3.2.1 fixes 13 issues since the 3.2.0 release. For more information, please read the detailed [Release Notes](#).

3.2.0

- Released May 17, 2022
- [Release Notes](#)
- Source download: [kafka-3.2.0-src.tgz](#) ([asc](#), [sha512](#))
- Binary downloads:
 - Scala 2.12 - [kafka_2.12-3.2.0.tgz](#) ([asc](#), [sha512](#))
 - Scala 2.13 - [kafka_2.13-3.2.0.tgz](#) ([asc](#), [sha512](#))

We build for multiple versions of Scala. This only matters if you are using Scala and you want a version built for the same Scala version you use.

Otherwise any version should work (2.13 is recommended).

```
karthikchatla@Karthiks-MacBook-Pro: ~/Downloads
```

```
karthikchatla@Karthiks-MacBook-Pro: ~/Downloads
```

```
karthikchatla@Karthiks-MacBook-Pro: ~/Downloads
```

```
karthikchatla@Karthiks-MacBook-Pro: ~/Downloads
```

```
LICENSE      bin      libs      logs
```

```
NOTICE       config   licenses  site-docs
```

```
karthikchatla@Karthiks-MacBook-Pro: ~/Downloads
```

```
[2022-12-06 20:22:12,774] INFO Reading configuration from: config/zookeeper.properties (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
```

```
[2022-12-06 20:22:12,775] WARN config/zookeeper.properties is relative. Prepend ./ to indicate that you're sure! (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
```

```
[2022-12-06 20:22:12,779] INFO clientPortAddress is 0.0.0.0:2181 (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
```

```
[2022-12-06 20:22:12,779] INFO secureClientPort is not set (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
```

```
[2022-12-06 20:22:12,779] INFO observerMasterPort is not set (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
```

```
[2022-12-06 20:22:12,779] INFO metricsProvider.className is org.apache.zookeeper.metrics.impl.DefaultMetricsProvider (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
```

```
[2022-12-06 20:22:12,780] INFO autopurge.snapRetainCount set to 3 (org.apache.zookeeper.server.DataDirCleanManager)
```

```
[2022-12-06 20:22:12,781] INFO autopurge.purgeInterval set to 0 (org.apache.zookeeper.server.DataDirCleanManager)
```

```
[2022-12-06 20:22:12,781] INFO Purge task is not scheduled. (org.apache.zookeeper.server.DataDirCleanManager)
```

```
[2022-12-06 20:22:12,781] WARN Either no config or no quorum defined in config, running in standalone mode (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
```

```
[2022-12-06 20:22:12,782] INFO Load1.2.imx support not found; imx disabled (org.apache.zookeeper.metrics.impl.DefaultMetricsProvider)
```

3. Starting Kafka Brokers

Create another terminal, do not close zookeeper

```
sses/cloud_computing/kafka/hw/q6/2022_fall/KafkaQuickStart.pdf
kafka_2.12-3.3.1 — java -Xmx512M -Xms512M -server -XX:+UseG1GC -XX:MaxGCPauseMillis=20 -XX:InitiatingHeapOccupancyPercent=35 -XX:+ExplicitGCInvokesC
...downloads/kafka_2.12-3.3.1/bin/./libs/connect-mirror-3.3.1.jar;/Users/karthikchatla/downloads/kafka_2.12-3.3.1/bin/./libs/connect-mirror-clien org.apache.zookeeper.server.quorum.QuorumPeerMain

Last login: Wed Dec 7 23:31:15 on ttys000
[karthikchatla@Karthiks-MBP ~ % cd downloads
[karthikchatla@Karthiks-MBP downloads % cd kafka_2.12-3.3.1
[karthikchatla@Karthiks-MBP kafka_2.12-3.3.1 % ls
[karthikchatla@Karthiks-MBP kafka_2.12-3.3.1 % bin/zookeeper-server-start.sh config/zookeeper.properties
[2022-12-07 23:32:48,645] INFO Reading configuration from: config/zookeeper.properties (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2022-12-07 23:32:48,646] WARN config/zookeeper.properties is relative. Prepend ./ to indicate that you're sure! (org.apache.zookeeper.server.quorum.QuorumPeerC
[2022-12-07 23:32:48,651] INFO clientPortAddress is 0.0.0.0:2181 (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2022-12-07 23:32:48,651] INFO secureClientPort is not set (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2022-12-07 23:32:48,651] INFO observerMasterPort is not set (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2022-12-07 23:32:48,651] INFO metricsProvider.className is org.apache.zookeeper.metrics.impl.DefaultMetricsProvider (org.apache.zookeeper.server.quorum.QuorumP
[2022-12-07 23:32:48,652] INFO autopurge.snapRetainCount set to 3 (org.apache.zookeeper.server.DatadirCleanupManager)
[2022-12-07 23:32:48,652] INFO autopurge.purgeInterval set to 0 (org.apache.zookeeper.server.DatadirCleanupManager)
[2022-12-07 23:32:48,652] INFO Purge task is not scheduled. (org.apache.zookeeper.server.DatadirCleanupManager)
[2022-12-07 23:32:48,653] WARN Either no config or no quorum defined in config, running in standalone mode (org.apache.zookeeper.server.quorum.QuorumPeerMain)
[2022-12-07 23:32:48,654] INFO Log4j 1.2 jmx support not found; jmx disabled. (org.apache.zookeeper.jmx.ManagedUtil)
[2022-12-07 23:32:48,654] INFO Reading configuration from: config/zookeeper.properties (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2022-12-07 23:32:48,654] WARN config/zookeeper.properties is relative. Prepend ./ to indicate that you're sure! (org.apache.zookeeper.server.quorum.QuorumPeerC
[2022-12-07 23:32:48,655] INFO clientPortAddress is 0.0.0.0:2181 (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2022-12-07 23:32:48,655] INFO secureClientPort is not set (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2022-12-07 23:32:48,655] INFO observerMasterPort is not set (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2022-12-07 23:32:48,655] INFO metricsProvider.className is org.apache.zookeeper.metrics.impl.DefaultMetricsProvider (org.apache.zookeeper.server.quorum.QuorumP
[2022-12-07 23:32:48,655] INFO Starting server (org.apache.zookeeper.server.ZooKeeperServerMain)
[2022-12-07 23:32:48,663] INFO ServerMetrics initialized with provider org.apache.zookeeper.metrics.impl.DefaultMetricsProvider@41ee392b (org.apache.zookeeper.s
```


4. Creating Kafka Topics.

Create another terminal, do not close zookeeper and kafka brokers

```
bin/kafka-topics.sh --create --topic input_recommend_product --zookeeper localhost:2181 --partitions 3 --replication-factor 1
```

```
[2022-12-09 00:27:51,172] INFO App info kafka.server for 0 unregistered (org.apache.kafka.common.utils.AppInfoParser)
[2022-12-09 00:27:51,173] INFO shut down completed (kafka.server.KafkaServer)
[2022-12-09 00:27:51,173] ERROR Exiting Kafka due to fatal exception during startup. (kafka.Kafka$)
kafka.zookeeper.ZooKeeperClientTimeoutException: Timed out waiting for connection while in state: CONNECTING
    at kafka.zookeeper.ZooKeeperClient.$anonfun$waitUntilConnected$3(ZooKeeperClient.scala:254)
    at kafka.zookeeper.ZooKeeperClient.waitUntilConnected(ZooKeeperClient.scala:250)
    at kafka.zookeeper.ZooKeeperClient.<init>(ZooKeeperClient.scala:108)
    at kafka.zk.KafkaZkClient$.apply(KafkaZkClient.scala:1980)
    at kafka.server.KafkaServer.initZkClient(KafkaServer.scala:503)
    at kafka.server.KafkaServer.startup(KafkaServer.scala:203)
    at kafka.Kafka$.main(Kafka.scala:109)
    at kafka.Kafka.main(Kafka.scala)
[2022-12-09 00:27:51,173] INFO shutting down (kafka.server.KafkaServer)
karthikchatla@Karthiks-MBP kafka_2.12-3.3.1 % bin/kafka-topics.sh --create --topic input_recommend_product --zookeeper localhost:2181 --partition
--replication-factor 1
Exception in thread "main" joptsimple.UnrecognizedOptionException: zookeeper is not a recognized option
    at joptsimple.OptionException.unrecognizedOption(OptionException.java:108)
    at joptsimple.OptionParser.handleLongOptionToken(OptionParser.java:510)
    at joptsimple.OptionParserState$2.handleArgument(OptionParserState.java:56)
    at joptsimple.OptionParser.parse(OptionParser.java:396)
    at kafka.admin.TopicCommand$TopicCommandOptions.<init>(TopicCommand.scala:567)
    at kafka.admin.TopicCommand$.main(TopicCommand.scala:47)
    at kafka.admin.TopicCommand.main(TopicCommand.scala)
karthikchatla@Karthiks-MBP kafka_2.12-3.3.1 %
```

5 Creating Producer and Consumer using Kafka-python

5.1 Create producer.py

```
from kafka import KafkaProducer
producer =
KafkaProducer(bootstrap_servers='localhost:9092')
producer.send('input_recommend_product', b'(1,
Main Menu), (2, Phone) ,(3, Smart Phone), (4,
iPhone)')
producer.close()
```

5.2 Create consumer.py

```
from kafka import KafkaConsumer
consumer = KafkaConsumer('input_recommend_product',
bootstrap_servers=['localhost:9092'])
for msg in consumer:
    print(msg)
```

5.3 Run consumer.py first (you can run it in your IDE)

5.4 Create another terminal, run the producer.py

5.5 Go to the consumer terminal, you can see the result

Enhancement Ideas

- Kafka/Spark data pipelines are custom code based on some very stripped down system software, with all the dials available to you so that you can build exactly what you need.
- Building the system that matches the various impedances is not done automatically, but is something that has to be designed into your software.

Conclusion

- Kafka + Spark Streaming + PySpark is a powerful combination for real-time data processing.
- It allows for the ingestion of data from Kafka topics, the processing of data using Spark Streaming and PySpark, and the output of the processed data to other systems.
- This combination of technologies can be used to build powerful real-time data pipelines and applications.