Yandex

Cluster mode

Two unresolved issues

- How to make a standalone application
- How to run an application on a cluster

```
$ pyspark
>>> from collections import namedtuple
>>> from datetime import datetime, timedelta
>>> Record = namedtuple("Record", ["date", "open", "high", "low", "close",
                                   "adj_close", "volume"])
>>> def parse_record(s):
...   fields = s.split(",")
...   return Record(fields[0], *map(float, fields[1:6]), int(fields[6]))
...
>>> def get_next_date(s):
...   fmt = "%Y-%m-%d"
...   return (datetime.strptime(s, fmt) + timedelta(days=1)).strftime(fmt)
...
>>> parsed_data = sc.textFile("nasdaq.csv").map(parse_record).cache()
>>> date_and_close_price = parsed_data.map(lambda r: (r.date, r.close))
>>> date_and_prev_close_price = parsed_data.map(lambda r: (get_next_date(r.date), r.close))
>>> joined = date_and_close_price.join(date_and_prev_close_price)
>>> returns = joined.mapValues(lambda p: (p[0] / p[1] - 1.0) * 100.0)
>>>
```

```
$ cat myapp.py
from collections import namedtuple
from datetime import datetime, timedelta
Record = namedtuple("Record", ["date", "open", "high", "low", "close",
                               "adj_close", "volume"])
def parse_record(s):
  fields = s.split(",")
  return Record(fields[0], *map(float, fields[1:6]), int(fields[6]))

def get_next_date(s):
  fmt = "%Y-%m-%d"
  return (datetime.strptime(s, fmt) + timedelta(days=1)).strftime(fmt)

from pyspark import SparkConf, SparkContext
sc = SparkContext(conf=SparkConf().setAppName("MyApp").setMaster("local"))

parsed_data = sc.textFile("nasdaq.csv").map(parse_record).cache()
date_and_close_price = parsed_data.map(lambda r: (r.date, r.close))
date_and_prev_close_price = parsed_data.map(lambda r: (get_next_date(r.date), r.close))
joined = date_and_close_price.join(date_and_prev_close_price)
returns = joined.mapValues(lambda p: (p[0] / p[1] - 1.0) * 100.0)

print(returns.collect())
```

```
$ cat myapp.py
from collections import namedtuple
from datetime import datetime, timedelta
Record = namedtuple("Record", ["date", "open", "high", "low", "close",
                               "adj_close", "volume"])
def parse_record(s):
  fields = s.split(",")
  return Record(fields[0], *map(float, fields[1:6]), int(fields[6]))

def get_next_d
  fmt = "%Y-%m
  return (dateti

from pyspark i
sc = SparkCon

parsed_data =
date_and_close_price = parsed_data.map(lambda r: (r.date, r.close))
date_and_prev_close_price = parsed_data.map(lambda r: (get_next_date(r.date), r.close))
joined = date_and_close_price.join(date_and_prev_close_price)
returns = joined.mapValues(lambda p: (p[0] / p[1] - 1.0) * 100.0)

print(returns.collect())
```

Possible values for the master URL:

local[K] — local mode with K threads
spark://HOST:PORT — standalone Spark cluster
mesos://HOST:PORT — Mesos cluster
yarn — YARN cluster

```
$ cat myapp.py
... lots of Python code ...
```

```
$ cat myapp.py
... lots of Python code ...

$ spark-submit myapp.py
... lots of Spark messages ...
```

```
$ spark-submit ./myapp.py
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
17/07/23 23:08:51 INFO SparkContext: Running Spark version 2.2.0
17/07/23 23:08:52 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
17/07/23 23:08:52 INFO SparkContext: Submitted application: MyApp
17/07/23 23:08:52 INFO SecurityManager: Changing view acls to: sandello
17/07/23 23:08:52 INFO SecurityManager: Changing modify acls to: sandello
17/07/23 23:08:52 INFO SecurityManager: Changing view acls groups to:
17/07/23 23:08:52 INFO SecurityManager: Changing modify acls groups to:
17/07/23 23:08:52 INFO SecurityManager: SecurityManager: authentication disabled; ui acls disabled; users  with view permissions: Set(sandello); groups with view permissions: Set(); users  with modify permissions: Set(sandello); groups wi
th modify permissions: Set()
17/07/23 23:08:52 INFO Utils: Successfully started service 'sparkDriver' on port 54222.
17/07/23 23:08:52 INFO SparkEnv: Registering MapOutputTracker
17/07/23 23:08:53 INFO SparkEnv: Registering BlockManagerMaster
17/07/23 23:08:53 INFO BlockManagerMasterEndpoint: Using org.apache.spark.storage.DefaultTopologyMapper for getting topology information
17/07/23 23:08:53 INFO BlockManagerMasterEndpoint: BlockManagerMasterEndpoint up
17/07/23 23:08:53 INFO DiskBlockManager: Created local directory at /private/var/folders/_6/cf5sbvgs45d8rft5r085zss58sjrq5/T/blockmgr-cfbaadb3-3f95-49b1-b8be-a66ceff959de
17/07/23 23:08:53 INFO MemoryStore: MemoryStore started with capacity 366.3 MB
17/07/23 23:08:53 INFO SparkEnv: Registering OutputCommitCoordinator
17/07/23 23:08:53 INFO Utils: Successfully started service 'SparkUI' on port 4040.
17/07/23 23:08:53 INFO SparkUI: Bound SparkUI to 0.0.0.0, and started at http://77.88.19.2:4040
17/07/23 23:08:53 INFO SparkContext: Added file file:/Users/sandello/1/./myapp.py at file:/Users/sandello/1/./myapp.py with timestamp 1500840533997
17/07/23 23:08:54 INFO Utils: Copying /Users/sandello/1/myapp.py to /private/var/folders/_6/cf5sbvgs45d8rft5r085zss58sjrq5/T/spark-cffcc59d-9aa3-4158-b6f5-a24194b061a7/userFiles-16db97e0-e0c2-4edf-a0c7-cda707546c66/myapp.py
17/07/23 23:08:54 INFO Executor: Starting executor ID driver on host localhost
17/07/23 23:08:54 INFO Utils: Successfully started service 'org.apache.spark.network.netty.NettyBlockTransferService' on port 54223.
17/07/23 23:08:54 INFO NettyBlockTransferService: Server created on 77.88.19.2:54223
17/07/23 23:08:54 INFO BlockManager: Using org.apache.spark.storage.RandomBlockReplicationPolicy for block replication policy
17/07/23 23:08:54 INFO BlockManagerMaster: Registering BlockManager BlockManagerId(driver, 77.88.19.2, 54223, None)
17/07/23 23:08:54 INFO BlockManagerMasterEndpoint: Registering block manager 77.88.19.2:54223 with 366.3 MB RAM, BlockManagerId(driver, 77.88.19.2, 54223, None)
17/07/23 23:08:54 INFO BlockManagerMaster: Registered BlockManager BlockManagerId(driver, 77.88.19.2, 54223, None)
17/07/23 23:08:54 INFO BlockManager: Initialized BlockManager: BlockManagerId(driver, 77.88.19.2, 54223, None)
17/07/23 23:08:55 INFO MemoryStore: Block broadcast_0 stored as values in memory (estimated size 236.5 KB, free 366.1 MB)
17/07/23 23:08:55 INFO MemoryStore: Block broadcast_0_piece0 stored as bytes in memory (estimated size 22.9 KB, free 366.0 MB)
17/07/23 23:08:55 INFO BlockManagerInfo: Added broadcast_0_piece0 in memory on 77.88.19.2:54223 (size: 22.9 KB, free: 366.3 MB)
17/07/23 23:08:55 INFO SparkContext: Created broadcast 0 from textFile at NativeMethodAccessorImpl.java:0
17/07/23 23:08:55 INFO FileInputFormat: Total input paths to process : 1
17/07/23 23:08:55 INFO SparkContext: Starting job: collect at /Users/sandello/1/./myapp.py:22
17/07/23 23:08:55 INFO DAGScheduler: Registering RDD 7 (join at /Users/sandello/1/./myapp.py:19)
17/07/23 23:08:55 INFO DAGScheduler: Got job 0 (collect at /Users/sandello/1/./myapp.py:22) with 2 output partitions
17/07/23 23:08:55 INFO DAGScheduler: Final stage: ResultStage 1 (collect at /Users/sandello/1/./myapp.py:22)
17/07/23 23:08:55 INFO DAGScheduler: Parents of final stage: List(ShuffleMapStage 0)
17/07/23 23:08:55 INFO DAGScheduler: Missing parents: List(ShuffleMapStage 0)
17/07/23 23:08:55 INFO DAGScheduler: Submitting ShuffleMapStage 0 (PairwiseRDD[7] at join at /Users/sandello/1/./myapp.py:19), which has no missing parents
17/07/23 23:08:56 INFO MemoryStore: Block broadcast_1 stored as values in memory (estimated size 13.8 KB, free 366.0 MB)
17/07/23 23:08:56 INFO MemoryStore: Block broadcast_1_piece0 stored as bytes in memory (estimated size 7.2 KB, free 366.0 MB)
17/07/23 23:08:56 INFO BlockManagerInfo: Added broadcast_1_piece0 in memory on 77.88.19.2:54223 (size: 7.2 KB, free: 366.3 MB)
17/07/23 23:08:56 INFO SparkContext: Created broadcast 1 from broadcast at DAGScheduler.scala:1006
17/07/23 23:08:56 INFO DAGScheduler: Submitting 2 missing tasks from ShuffleMapStage 0 (PairwiseRDD[7] at join at /Users/sandello/1/./myapp.py:19) (first 15 tasks are for partitions Vector(0, 1))
17/07/23 23:08:56 INFO TaskSchedulerImpl: Adding task set 0.0 with 2 tasks
17/07/23 23:08:56 INFO TaskSetManager: Starting task 0.0 in stage 0.0 (TID 0, localhost, executor driver, partition 0, PROCESS_LOCAL, 4950 bytes)
17/07/23 23:08:56 INFO Executor: Running task 0.0 in stage 0.0 (TID 0)
17/07/23 23:08:56 INFO Executor: Fetching file:/Users/sandello/1/./myapp.py with timestamp 1500840533997
17/07/23 23:08:56 INFO Utils: /Users/sandello/1/./myapp.py has been previously copied to /private/var/folders/_6/cf5sbvgs45d8rft5r085zss58sjrq5/T/spark-cffcc59d-9aa3-4158-b6f5-a24194b061a7/userFiles-16db97e0-e0c2-4edf-a0c7-cda707546c66/my
app.py
17/07/23 23:08:56 INFO HadoopRDD: Input split: file:/Users/sandello/1/nasdaq.csv:0+11398
17/07/23 23:08:57 INFO PythonRunner: Times: total = 493, boot = 466, init = 24, finish = 3
17/07/23 23:08:57 INFO MemoryStore: Block rdd_2_0 stored as bytes in memory (estimated size 5.7 KB, free 366.0 MB)
17/07/23 23:08:57 INFO BlockManagerInfo: Added rdd_2_0 in memory on 77.88.19.2:54223 (size: 5.7 KB, free: 366.3 MB)
17/07/23 23:08:57 INFO PythonRunner: Times: total = 4, boot = -207, init = 209, finish = 2
/usr/local/Cellar/apache-spark/2.2.0/libexec/python/lib/pyspark.zip/pyspark/shuffle.py:58: UserWarning: Please install psutil to have better support with spilling
17/07/23 23:08:57 INFO PythonRunner: Times: total = 50, boot = 3, init = 3, finish = 44
17/07/23 23:08:57 INFO Executor: Finished task 0.0 in stage 0.0 (TID 0). 2006 bytes result sent to driver
```

Summary

- You have learned how to:
  - create the SparkContext object
  - correctly set the master URL
  - launch an application with the 'spark-submit' command

BigDATAteam