CS 5433: Bigdata Management
Programming Assignment 3
Task 4– Spark pipeline for the Task-1, Task-2, Task-3

Group 4

Task 4: Creating Spark pipeline for the Task-1, Task-2, Task-3

In this task, we have created a pipeline for Task1, Task2 and Task3. We have used pipeline to for each task to reduce the code complexity.

Description of Dataset:

For Task1, we are using the null values inserted dataset and for Task2 & Task3, we have used the output generated from Task1.

The input Data set for the Task 1

- Institute ID which is a "String" column
- Name – Name of the university/institute of type "String"
- City – Name of the city where university is located, which is of type "String"
- State – Name of the State where university is located, which is of type "String"
- PR Score – PR Score of the university which is of type "Double"
- PR Rank – PR Rank of the university which is of type "Integer"
- PR Score – PR Score of the university which is of type "Double"
- Score – Score of the university which is of type "Double"
- Year –Year (contains values 2017,2018,2019,2020 & 2021) is of type "Integer"
- Rank – Rank of the university which is of type "Integer"

Input file:

hdfs://hadoop-nn001.cs.okstate.edu:9000/user/sdarapu/Group4_DataSet/IndianUniversityRankingFrom2017to2021.csv

The input Data Set for the Task 2 & 3 is the Output from the Task 1

- Institute ID which is a "Double" column
- Name – Name of the university/institute of type "Double"
- City – Name of the city where university is located, which is of type "Double"
- State – Name of the State where university is located, which is of type "Double"
- PR Score – PR Score of the university which is of type "Double"
- PR Rank – PR Rank of the university which is of type "Double"
- PR Score – PR Score of the university which is of type "Double"
- Score – Score of the university which is of type "Double"

- Year –Year (contains values 2017,2018,2019,2020 & 2021) is of type “Double”
- Rank – Rank of the university which is of type “Double”

Input file:

hdfs://hadoop-nn001.cs.okstate.edu:9000/user/sdarapu/Assign3_Group4_Task1_Output_inpfor_Task2-4/part-00000-571e77d2-85ae-4579-92f8-dd4dc788ab7f-c000.csv"

Technical Approach and Formulae

Task-1:

Cosine Similarity:

Cosine Similarity is described as a type of similarity measure which is used to measure how similar the data frames are which is irrespective of their size. In the terms of mathematics, it describes the cosine of angle between the formed vectors which are projected in a multi-dimensional space. This similarity measure is very advantageous because if the angle between them is smaller then there is higher similarity. The formula for cosine similarity is described below:

$$Cos\theta = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \, \|\vec{b}\|} = \frac{\sum_1^n a_i b_i}{\sqrt{\sum_1^n a_i^2} \sqrt{\sum_1^n b_i^2}}$$

where, $\vec{a} \cdot \vec{b} = \sum_1^n a_i b_i = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$ is the dot product of the two vectors.

Cosine Similarity Formula

Fig 4,1: Cosine similarity formula

TASK - 2 & 3:

Linear Regression: Linear regression is a type of supervised learning of machine learning algorithm. It carries out a regression task. Based on independent variables, regression models a goal prediction value. It is mostly utilized in forecasting and determining the link between variables. Different regression models differ in terms of the type of relationship they evaluate between dependent and independent variables, as well as the number of independent variables they employ.

Random Forest Regression: Random Forest Regression is a supervised learning technique that solves classification or regression issues using an ensemble learning method. Ensemble learning is a machine learning technique that integrates predictions from numerous machine learning

algorithms to get a better prediction than a single algorithm. A random forest is an estimator technique that combines the results of several decision trees to produce the best possible result.

RMSE: It is also called as Root Mean Squared Error. The standard deviation of the errors that occur when making a prediction on a dataset is known as the RMSE. This is the same as MSE (Mean Squared Error), but the root of the number is taken into account when calculating the model's accuracy.

R2: The R2 score is a critical indicator for assessing the effectiveness of a regression-based machine learning model. It's also known as the coefficient of determination and is called as R squared. It operates by calculating the amount of variation in the dataset-explained predictions.

Pipeline: The end-to-end construct that orchestrates the flow of data into and output from a machine learning model is known as a machine learning pipeline (or set of multiple models). It contains raw data input, features, outputs, the machine learning model and model parameters, and prediction outputs, as well as the machine learning model and model parameters.

Approach:

Below are the steps we have followed to complete Task4 of this assignment,

1. At first, we have created a python file("Assign3_Group4_Task4.py") in the Hadoop cluster. Refer to "Group_4_Task_4_code.pdf".

2. **Code Explanation:**

   For Task1 Pipeline creation:

   All the libraries from Task 1 are imported to this program and we have imported one extra library which is shown below.

   ```
   from pyspark.ml import Pipeline
   ```

   And everything is same as Task 1 but here we have created a pipeline in which our data frame passes through. The pipeline code is shown as below,

   ```
   pipeline1=Pipeline(stages=[InsID_indexer,Name_indexer,State_indexer,City_ind
   exer])
   data=pipeline1.fit(data).transform(data)

   df=data.toPandas()
   ```

For Task 2 & 3 Pipeline creation for the model Linear Regression:

All the libraries from Task 2 & 3 for the model Linear Regression are imported to this program and we have imported one extra library which is shown below.

```
from pyspark.ml import Pipeline
```

And everything is same as Task 2 & 3 but here we have created a pipeline in which vector Assembler, normalizer and linear Regression model passes through where normalizer normalizes the values. The pipeline code is shown as below,

```
pipeline = Pipeline(stages=[vectorAssembler,normalizer, lr])
```

By using this pipeline we have fit the training data and then transformed the test data.

```
lr_model = pipeline.fit(trainingData)
lr_predictions = lr_model.transform(testData)
```

For Task 2 & 3 Pipeline creation for the model Random Forest:

All the libraries from Task 2 & 3 for the model Random Forest are imported to this program and we have imported one extra library which is shown below.

```
from pyspark.ml import Pipeline
```

And everything is same as Task 2 & 3 but here we have created a pipeline in which vector Assembler, normalizer and linear Regression model passes through where normalizer normalizes the values. The pipeline code is shown as below,

```
pipeline = Pipeline(stages=[vectorAssembler1,normalizer1, rf])
```

By using this pipeline we have fit the training data and then transformed the test data.

```
rf_model = pipeline.fit(trainingData1)
rf_predictions = rf_model.transform(testData1)
```

3. Steps to execute the code:

i. To run the code, we have executed below command as shown below.

```
sdarapu@hadoop-nn001:~$ spark-submit /home/sdarapu/Assign3_Group4_Task4.py
```

Fig 4,2: Command to execute

ii. The above command executes as follows.

```
sdarapu@hadoop-nn001:~$ spark-submit /home/sdarapu/Assign3_Group4_Task4.py
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.spark.unsafe.Platform (file:/usr/local/spark-3.0.1-bin-hadoop3.2/jars/spark-unsafe_2.12-3.0.
DirectByteBuffer(long,int)
WARNING: Please consider reporting this to the maintainers of org.apache.spark.unsafe.Platform
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
2022-04-30 10:55:07,222 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes wher
2022-04-30 10:55:09,067 INFO spark.SparkContext: Running Spark version 3.0.1
2022-04-30 10:55:09,127 INFO resource.ResourceUtils: ==============================================================
2022-04-30 10:55:09,129 INFO resource.ResourceUtils: Resources for spark.driver:

2022-04-30 10:55:09,129 INFO resource.ResourceUtils: ==============================================================
2022-04-30 10:55:09,130 INFO spark.SparkContext: Submitted application: Assignment3_Group4_Task4
2022-04-30 10:55:09,202 INFO spark.SecurityManager: Changing view acls to: sdarapu
2022-04-30 10:55:09,203 INFO spark.SecurityManager: Changing modify acls to: sdarapu
2022-04-30 10:55:09,203 INFO spark.SecurityManager: Changing view acls groups to:
2022-04-30 10:55:09,203 INFO spark.SecurityManager: Changing modify acls groups to:
2022-04-30 10:55:09,203 INFO spark.SecurityManager: SecurityManager: authentication disabled; ui acls disabled; users  with view permissions:
 permissions: Set(); users  with modify permissions: Set(sdarapu); groups with modify permissions: Set()
2022-04-30 10:55:09,509 INFO util.Utils: Successfully started service 'sparkDriver' on port 37573.
2022-04-30 10:55:09,543 INFO spark.SparkEnv: Registering MapOutputTracker
2022-04-30 10:55:09,578 INFO spark.SparkEnv: Registering BlockManagerMaster
2022-04-30 10:55:09,601 INFO storage.BlockManagerMasterEndpoint: Using org.apache.spark.storage.DefaultTopologyMapper for getting topology in
2022-04-30 10:55:09,601 INFO storage.BlockManagerMasterEndpoint: BlockManagerMasterEndpoint up
2022-04-30 10:55:09,644 INFO spark.SparkEnv: Registering BlockManagerMasterHeartbeat
2022-04-30 10:55:09,658 INFO storage.DiskBlockManager: Created local directory at /tmp/blockmgr-5f0f3218-74e8-445a-a733-f640c1d34605
2022-04-30 10:55:09,685 INFO memory.MemoryStore: MemoryStore started with capacity 434.4 MiB
2022-04-30 10:55:09,732 INFO spark.SparkEnv: Registering OutputCommitCoordinator
```

Fig 4,3: Execution process

```
2022-04-30 10:55:11,243 WARN yarn.Client: Neither spark.yarn.jars nor spark.yarn.archive is set, falling back to uploading libraries
2022-04-30 10:55:14,338 INFO yarn.Client: Uploading resource file:/tmp/spark-6a69e261-ea35-49d6-97af-86381b87ccf7/__spark_libs__7816
01.cs.okstate.edu:9000/user/sdarapu/.sparkStaging/application_1647031195237_1589/__spark_libs__7816335684312437030.zip
2022-04-30 10:55:16,813 INFO yarn.Client: Uploading resource file:/usr/local/spark/python/lib/pyspark.zip -> hdfs://hadoop-nn001.cs.
ing/application_1647031195237_1589/pyspark.zip
2022-04-30 10:55:16,876 INFO yarn.Client: Uploading resource file:/usr/local/spark/python/lib/py4j-0.10.9-src.zip -> hdfs://hadoop-n
parkStaging/application_1647031195237_1589/py4j-0.10.9-src.zip
2022-04-30 10:55:17,105 INFO yarn.Client: Uploading resource file:/tmp/spark-6a69e261-ea35-49d6-97af-86381b87ccf7/__spark_conf__7050
01.cs.okstate.edu:9000/user/sdarapu/.sparkStaging/application_1647031195237_1589/__spark_conf__.zip
2022-04-30 10:55:17,163 INFO spark.SecurityManager: Changing view acls to: sdarapu
2022-04-30 10:55:17,163 INFO spark.SecurityManager: Changing modify acls to: sdarapu
2022-04-30 10:55:17,163 INFO spark.SecurityManager: Changing view acls groups to:
2022-04-30 10:55:17,163 INFO spark.SecurityManager: Changing modify acls groups to:
2022-04-30 10:55:17,164 INFO spark.SecurityManager: SecurityManager: authentication disabled; ui acls disabled; users  with view per
 permissions: Set(); users  with modify permissions: Set(sdarapu); groups with modify permissions: Set()
2022-04-30 10:55:17,188 INFO yarn.Client: Submitting application application_1647031195237_1589 to ResourceManager
2022-04-30 10:55:17,227 INFO impl.YarnClientImpl: Submitted application application_1647031195237_1589
2022-04-30 10:55:18,232 INFO yarn.Client: Application report for application_1647031195237_1589 (state: ACCEPTED)
2022-04-30 10:55:18,237 INFO yarn.Client:
         client token: N/A
         diagnostics: AM container is launched, waiting for AM container to Register with RM
         ApplicationMaster host: N/A
         ApplicationMaster RPC port: -1
         queue: default
         start time: 1651334117206
         final status: UNDEFINED
         tracking URL: http://hadoop-nn001.cs.okstate.edu:8088/proxy/application_1647031195237_1589/
         user: sdarapu
2022-04-30 10:55:19,240 INFO yarn.Client: Application report for application_1647031195237_1589 (state: ACCEPTED)
2022-04-30 10:55:20,242 INFO yarn.Client: Application report for application_1647031195237_1589 (state: ACCEPTED)
2022-04-30 10:55:21,195 INFO cluster.YarnClientSchedulerBackend: Add WebUI Filter. org.apache.hadoop.yarn.server.webproxy.amfilter.A
01, PROXY_URI_BASES -> http://hadoop-nn001:8088/proxy/application_1647031195237_1589), /proxy/application_1647031195237_1589
2022-04-30 10:55:21,245 INFO yarn.Client: Application report for application_1647031195237_1589 (state: RUNNING)
2022-04-30 10:55:21,245 INFO yarn.Client:
```

Fig 4,4: Execution process

```
2022-04-30 10:55:35,091 INFO scheduler.DAGScheduler: Final stage: ResultStage 3 (showString at NativeMethodAccessorImpl.java:0)
2022-04-30 10:55:35,091 INFO scheduler.DAGScheduler: Parents of final stage: List(ShuffleMapStage 2)
2022-04-30 10:55:35,092 INFO scheduler.DAGScheduler: Missing parents: List(ShuffleMapStage 2)
2022-04-30 10:55:35,094 INFO scheduler.DAGScheduler: Submitting ShuffleMapStage 2 (MapPartitionsRDD[13] at showString at NativeMethodAccessorIm
ing parents
2022-04-30 10:55:35,115 INFO memory.MemoryStore: Block broadcast_5 stored as values in memory (estimated size 26.6 KiB, free 433.7 MiB)
2022-04-30 10:55:35,122 INFO memory.MemoryStore: Block broadcast_5_piece0 stored as bytes in memory (estimated size 11.1 KiB, free 433.7 MiB)
2022-04-30 10:55:35,124 INFO storage.BlockManagerInfo: Added broadcast_5_piece0 in memory on hadoop-nn001:38337 (size: 11.1 KiB, free: 434.3 Mi
2022-04-30 10:55:35,124 INFO spark.SparkContext: Created broadcast 5 from broadcast at DAGScheduler.scala:1223
2022-04-30 10:55:35,127 INFO scheduler.DAGScheduler: Submitting 1 missing tasks from ShuffleMapStage 2 (MapPartitionsRDD[13] at showString at N
0) (first 15 tasks are for partitions Vector(0))
2022-04-30 10:55:35,127 INFO cluster.YarnScheduler: Adding task set 2.0 with 1 tasks
2022-04-30 10:55:35,131 INFO scheduler.TaskSetManager: Starting task 0.0 in stage 2.0 (TID 2, hadoop-dn006.cs.okstate.edu, executor 2, partitio
2022-04-30 10:55:35,161 INFO storage.BlockManagerInfo: Added broadcast_5_piece0 in memory on hadoop-dn006.cs.okstate.edu:40875 (size: 11.1 KiB,
2022-04-30 10:55:35,317 INFO storage.BlockManagerInfo: Added broadcast_4_piece0 in memory on hadoop-dn006.cs.okstate.edu:40875 (size: 28.5 KiB,
2022-04-30 10:55:35,410 INFO scheduler.TaskSetManager: Finished task 0.0 in stage 2.0 (TID 2) in 281 ms on hadoop-dn006.cs.okstate.edu (executo
2022-04-30 10:55:35,410 INFO cluster.YarnScheduler: Removed TaskSet 2.0, whose tasks have all completed, from pool
2022-04-30 10:55:35,414 INFO scheduler.DAGScheduler: ShuffleMapStage 2 (showString at NativeMethodAccessorImpl.java:0) finished in 0.315 s
2022-04-30 10:55:35,415 INFO scheduler.DAGScheduler: looking for newly runnable stages
2022-04-30 10:55:35,416 INFO scheduler.DAGScheduler: running: Set()
2022-04-30 10:55:35,417 INFO scheduler.DAGScheduler: waiting: Set(ResultStage 3)
2022-04-30 10:55:35,418 INFO scheduler.DAGScheduler: failed: Set()
2022-04-30 10:55:35,424 INFO scheduler.DAGScheduler: Submitting ResultStage 3 (MapPartitionsRDD[16] at showString at NativeMethodAccessorImpl.j
parents
2022-04-30 10:55:35,441 INFO memory.MemoryStore: Block broadcast_6 stored as values in memory (estimated size 15.7 KiB, free 433.7 MiB)
2022-04-30 10:55:35,451 INFO memory.MemoryStore: Block broadcast_6_piece0 stored as bytes in memory (estimated size 6.4 KiB, free 433.7 MiB)
```

Fig 4, 5: Execution process

Output displayed on the Console:

For Task1:

Before performing data correction using cosine similarity using pipeline

```
+------------+----+----+-----+--------+-------+-----+----+----+
|Institute ID|Name|City|State|PR Score|PR Rank|Score|Year|Rank|
+------------+----+----+-----+--------+-------+-----+----+----+
|           0|   0|   0|    0|       0|    113|    0|   0|   0|
+------------+----+----+-----+--------+-------+-----+----+----+
```

Fig 4, 6: 113 Null values are present in PR Rank column

No null values present after performing data correction using cosine similarity

```
+------------+----+----+-----+--------+-------+-----+----+----+
|INSTITUTE ID|NAME|CITY|STATE|PR Score|PR Rank|Score|Year|Rank|
+------------+----+----+-----+--------+-------+-----+----+----+
|           0|   0|   0|    0|       0|      0|    0|   0|   0|
+------------+----+----+-----+--------+-------+-----+----+----+
```

Fig 4, 7: No Null values are present in PR Rank column

For Task2 & Task3 Linear Regression:

RMSE value for Linear Regression after using pipeline

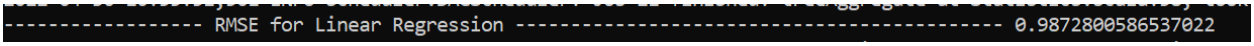

Fig 4, 8: RMSE value

R2 Value for Linear Regression after using pipeline

Fig 4, 9: R2 value

For Task2 & Task3 Random Forest:

RMSE value for Random Forest after using pipeline

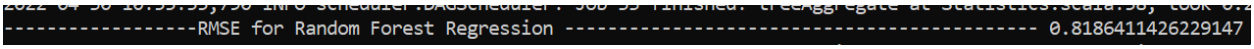

Fig 4, 10: RMSE value

R2 value for Random Forest after using pipeline

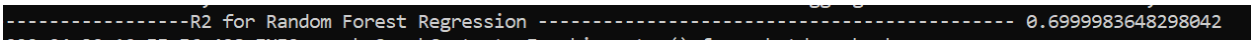

Fig 4, 11: R2 value

Discussion Of Results

In this Task, we have created pipeline for Task1, Task2 and Task3. We have found out that the RMSE value is more (i.e., more accuracy) for both models when using pipeline. More details are explained below:

We calculated RMSE value for both linear regression, Random Forest in Task 3 without using pipeline.

Without using Pipeline:

RMSE value for Linear Regression – 0.868

RMSE value for Random Forest – 0.713

With using Pipeline:

RMSE value for Linear Regression – 0.987

RMSE value for Random Forest - 0.818