Business domain:

Since our dataset contains a variety of sectors, it is critical to assess each one's performance using the change categories. We achieve this by grouping each sector's good, stable, and bad changes.

The output will be each sector with the change category occurrence, we can observe the sector performance.

Our approach:

First, we used already clean and preprocessed data, but to implement the MapReduce function, we select the change category and the sector columns, then we save. the save them on a new data frame to use them MapReduce function.

Then we built the MapReduce function, which as its name suggests, involves two phases: map and reduce. The reducer phase happens after the mapper phase is finished.

Installing MRJob and MRStep are required before starting the function,After that we created MRtarget class that takes the NRJob as a parameter, which contains two functions:

- Mapper phase

A mapper is a function that processes the input data. The mapper processes the data and creates several small chunks of data. The input to the mapper function is in the form of (key, value) pairs.

In our case, we employ the mapper function to generate a special key for the variable with contains the sector and change category, this key indicates the occurrence of the variable on the dataset.

The Mapper passes the output its output to the Reducer, which consists of lists of intermediate keys and their values, in sorted key order.

- Reducer phase

The Reducer takes the intermediate key-value pair output from the mapper and processes each one separately to produce the output.

In this step, we join all the tuples and use the sum aggregation to give us the total .

The final result, which is saved in HDFS, is the reducer's output.