RUNNING INSTRUCTION

Project Team: P2 Group 7

Table of Content

Data Scraping Process

Prerequisite

1. Python 3.7 installed

2. Python Library Needed:

```
pip install selenium
pip install httpx
pip install pandas
pip install parsel
pip install asyncio
pip install csv
```

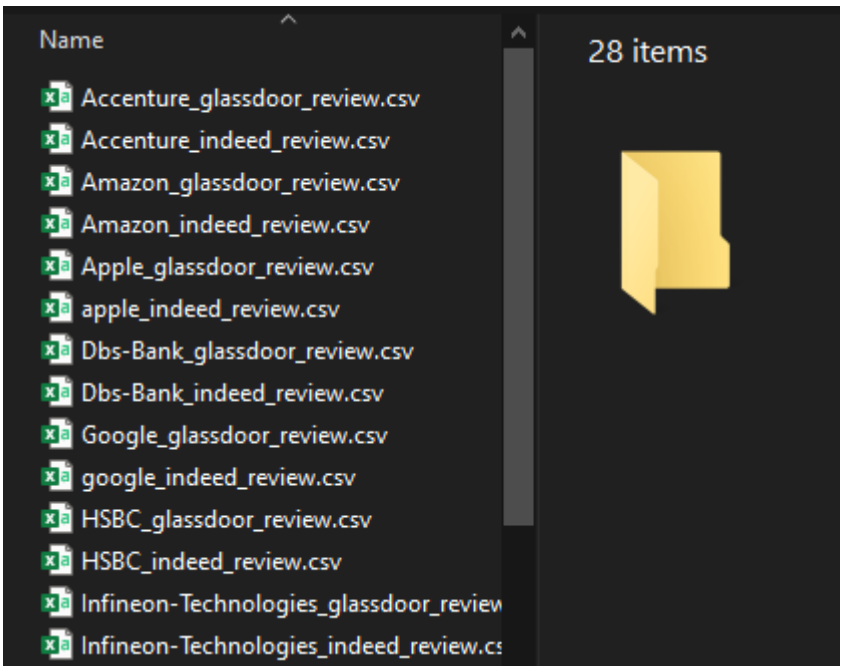
3. Chrome browser version 111 installed (to run chromedriver)

To run

To Run Data Scraper

```
cmd Source_code/DataScraping
python main.py
```

Expected output



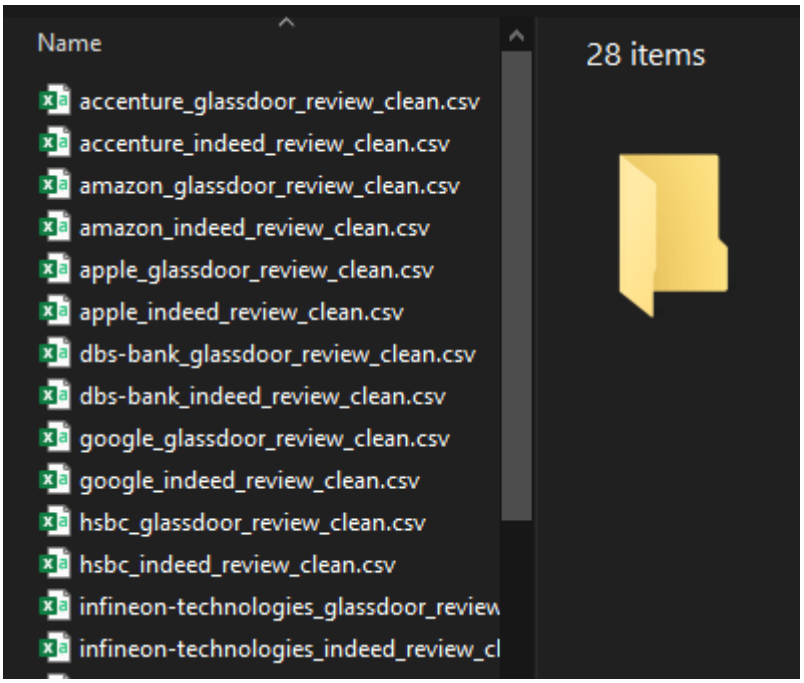
Data Cleaning

To run

To Run Data Cleaner

```
cmd Source_code
python dataCleaner.py
```

Expected output



Data Cleaning for LDA Topic Modelling using Mahout

Prerequisite

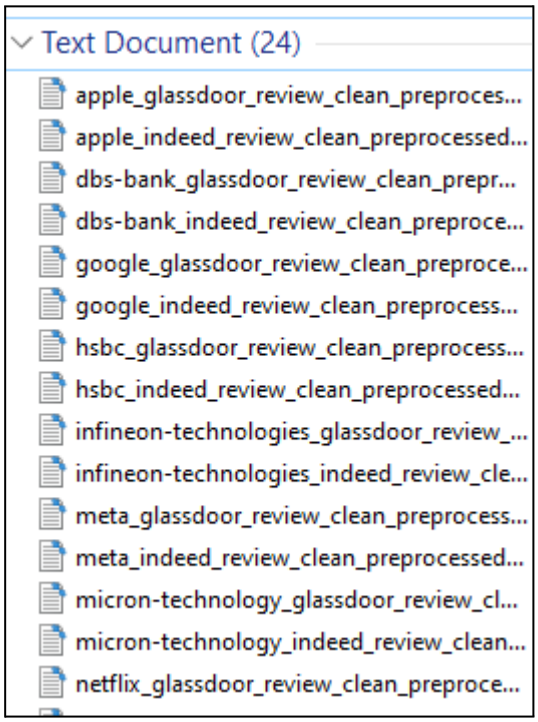
Ensure that csv files are in same location as the python file and NLTK is installed

To run

```
To Run pre-process on csv files
```

```
python preprocess_reviews.py
```

Expected Output



Hadoop MapReduce Analysis Guide:

Prerequisites

1. Download an IDE that can support Java and Maven: How to set up Java with Eclipse IDE
2. Install Java on your local computer: How to Install Java on Windows
3. Install Maven on your IDE: How to install Maven - javatpoint
4. Move all datasets into Dsail

Move files to DSAIL

1. Locate <datasets>.csv in folder: ".\Dataset_used\CleanDataset"
2. Locate stopwords.txt in folder: ".\Dataset_used\stopwords.txt"
3. Locate AFIN-111.txt in folder: ".\Dataset_used\AFIN-111.txt"
4. Locate company-industry.txt in folder: ".\Dataset_used\company-industry.txt"
4. Log into Dsail
ssh user@172.27.69.55
5. SCP all <datasets>.csv and .txt files into Dsail:
scp <all file paths delimited by space> user@172.27.69.55:
6. Perform ls command to check all <datasets> are available
7. Create a directory to store all <datasets>.csv
hadoop fs -mkdir input
8. Move <datasets>.csv into input folder
hadoop fs -put *.csv input
9. Move stopwords.txt, AFIN-111.txt, company-industry.txt into input folder

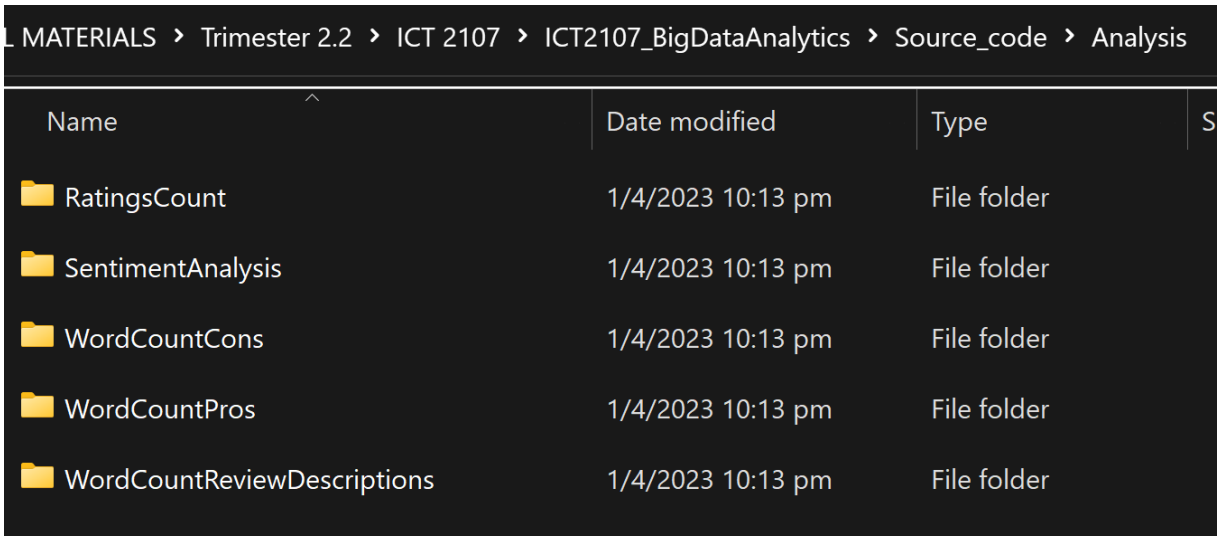
```
hadoop fs -put stopwords.txt input
hadoop fs -put AFIN-111.txt input
hadoop fs -put company-industry.txt input
```

8. Create an output folder to store output of mapreduce

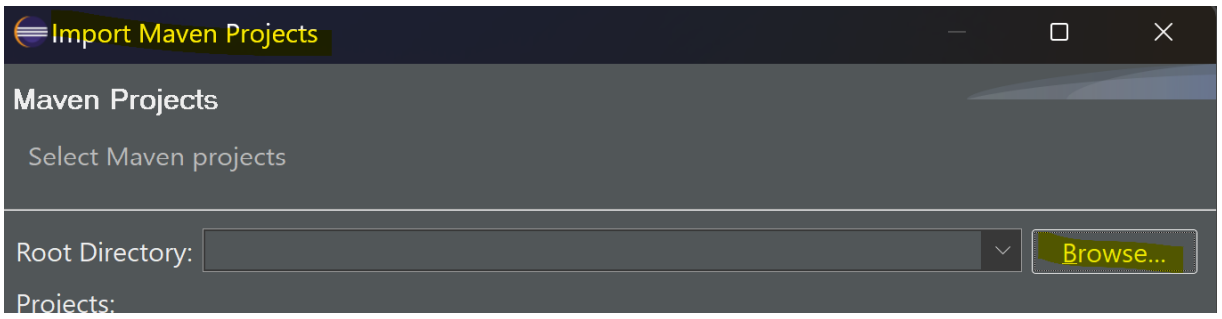
```
hadoop fs -mkdir output
```

Perform Analysis with Hadoop

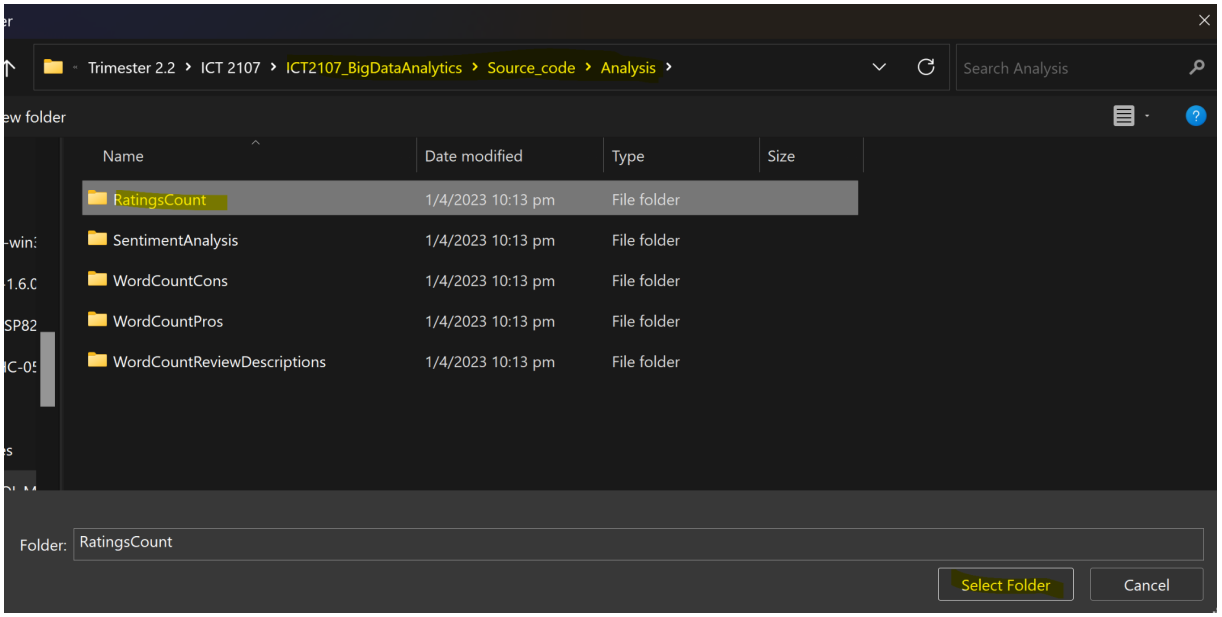
1. Select one analysis folder in:
".\Source_code\Analysis"



2. Open IDE to import maven project selected analysis folder



3. Find analysis folder and import selected analysis folder
Folder name: ".\Source_code\Analysis"



4.Locate the driver class and edit the hdfsPath: "hdfs://hadoop-master:9000/user/<userid>/input/"

```
// edit the hdfs path here
String hdfsPath = "hdfs://hadoop-master:9000/user/ict2101702/glassDoor/input/";
```

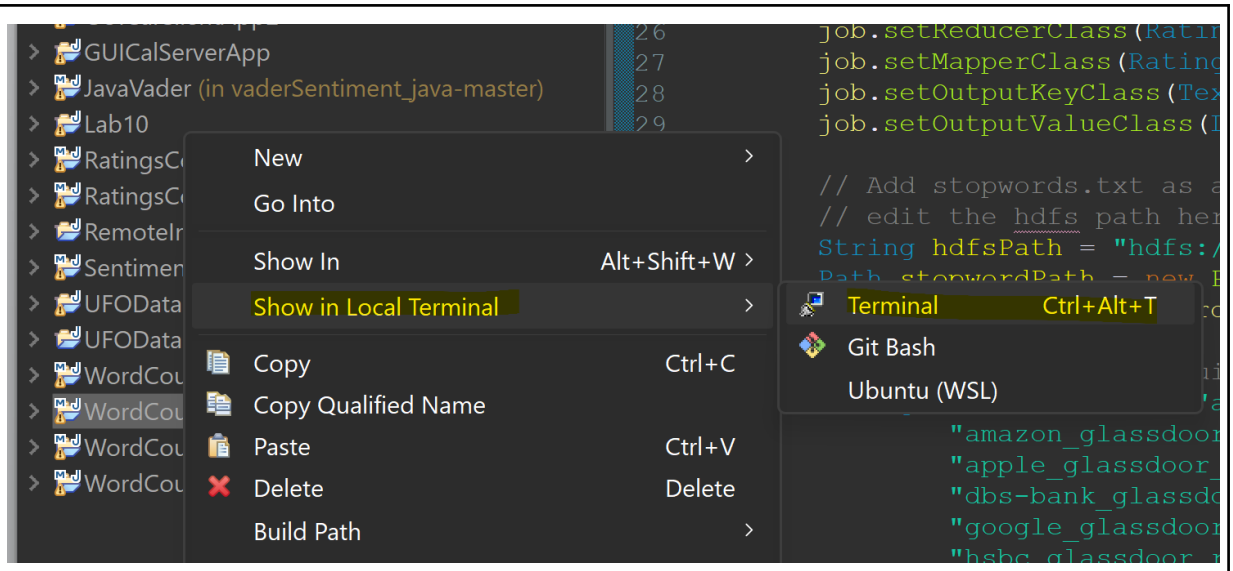
5. In the driver class edit the outputPath:"hdfs://hadoop-master:9000/user/<userid>/output/"

```
// Set the output path
Path outputPath = new Path(
        "hdfs://hadoop-master:9000/user/ict2101702/ratingCount/" + new Date().getTime());
FileOutputFormat.setOutputPath(job, outputPath);
```

6. In the driver class edit the cache file paths
URI("hdfs://hadoop-master:9000/user/<userid>/input/AFINN-111.txt")
URI("hdfs://hadoop-master:9000/user/<userid>/input/stopwords.txt")

```
job.addCacheFile(new URI("hdfs://hadoop-master:9000/user/ict2100868/project/input/AFINN-111.txt"));
job.addCacheFile(new URI("hdfs://hadoop-master:9000/user/ict2100868/project/input/stopwords.txt"));
```

6. Open a local terminal in IDE: Right Click Project/ Ctrl + Alt + T



7. Enter mvn package in local terminal opened

```
C:\Users\mager\eclipse-workspace\RatingsCountAverage>mvn package
[INFO] Scanning for projects...
[INFO]
[INFO] ------------------< com.example:ratingscountaverage >-------------------
[INFO] Building ratingscountaverage 0.0.1-SNAPSHOT
[INFO]   from pom.xml
[INFO] --------------------------------[ jar ]---------------------------------
```

8. Wait for this message

```
[INFO] Building jar: C:\Users\mager\eclipse-workspace\RatingsCountAverage\target\ratingscountaverage-0.0.1-SNAPSHOT-jar-with-dependencies.jar
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
```

9. Use the terminal Build jar: <filepath> to scp into Dsail

```
scp <filepath> user@172.27.69.55:
```

Mahout for LDA Topic Modelling Guide:

Prerequisite

1. Python 3.7 installed
2. Java Version 8 or later
3. Maven

[Text in yellow changed according to your directory]

Perform Topic Modelling on preprocessed text

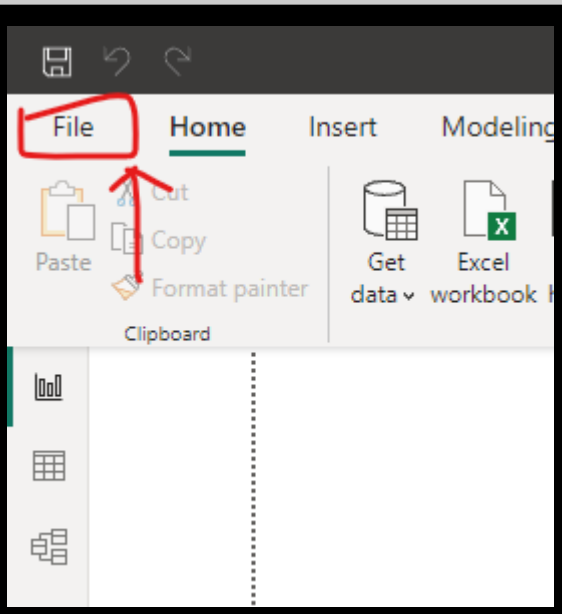
1. Run Mahout's seqdirectory command to convert text file from here
mahout seqdirectory -i <input-text> -o <output-directory>
2. Convert the SequenceFile to a sparse vector format
mahout seq2sparse -i <output-directory-used-above> -o <new-output-directory> -nv -wt tf -seq
3. Run mahout cvb to run LDA on the sparse vectors
mahout cvb -i <new-output-directory-used-above>/tfidf-vectors -o <Topic-Word-distribution-directory> -k <Number-of-Topics> -x <number of iterations> -dict <new-output-directory-used-above>/dictionary.file-0 -dt <lda-topic-document-output-directory>
4. Get output files from hadoop
hadoop fs -get <Topic-Word-distribution-directory> <Local-Path> hadoop fs -get <lda-topic-document-output-directory> <Local-Path>

Data Visualization

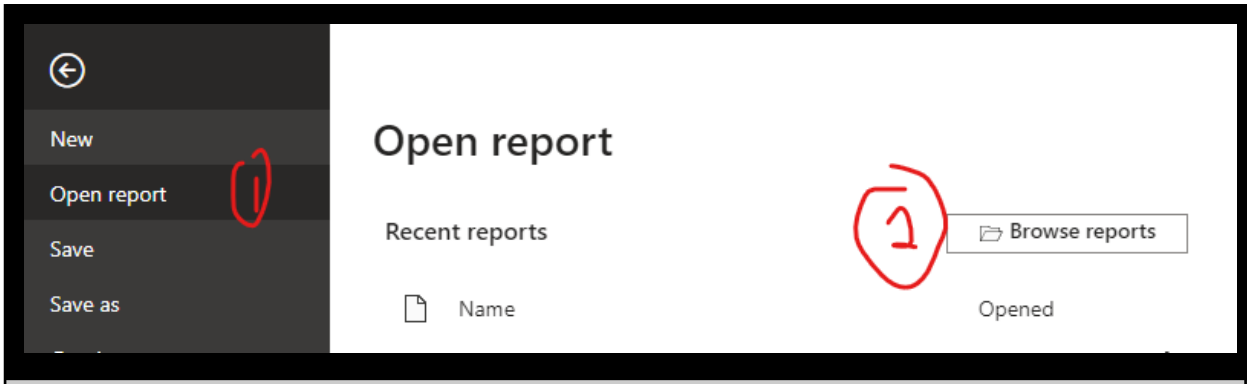
To visualise

1. Install and open Microsoft Power BI

2. Click on File



3. Open report and browser reports

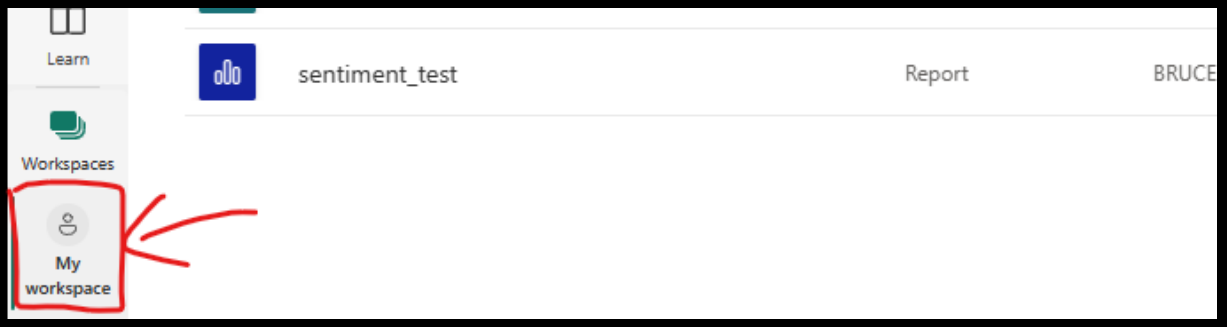


4. Select the "job_analysis_report.pbix" file under "Source_code\DataVisualization"

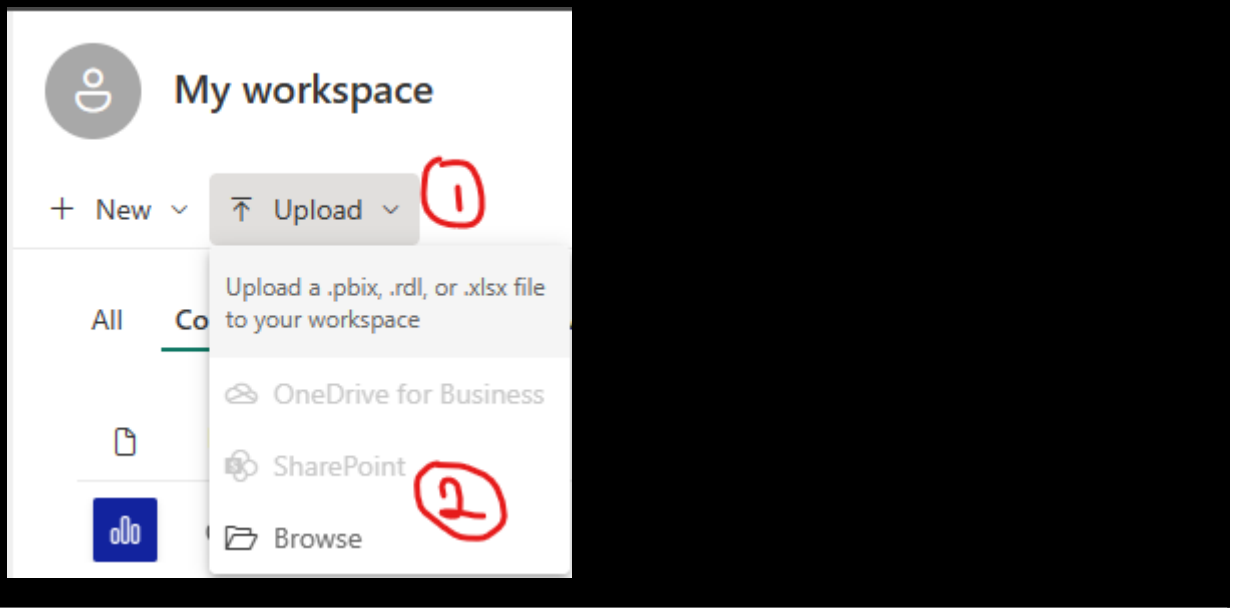
Alternative

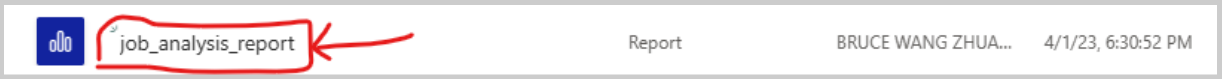
1. Sign in to Power BI on the Web

2. Go to My Workspace



3. Click on upload and then browse



4. Select the "job_analysis_report.pbix" file under "Source_code\DataVisualization"
5. Click on the "job_analysis_report.pbix" in Power BI


Expected output