[**Methodology and Experiments**](https://courses.torontomu.ca/d2l/lms/dropbox/user/folder_submit_files.d2l?db=320471&grpid=0&isprv=0&bp=0&ou=762967)

**Milestone Phase 1**

**Data Storage and Cleaning**

**In this phase, the focus is on storing the datasets and performing data cleaning tasks using Hadoop and PySpark.**

For managing and processing our large datasets, we utilize Apache Hadoop, an open-source software framework that allows for distributed processing of large data sets across clusters of computers. Given the scale of data from Amazon and Flipkart, Hadoop's ability to scale from a single server to thousands of machines, each offering local computation and storage, is crucial.

To assist in the cleaning and pre-processing of our data, we use PySpark, the Python library for Apache Spark. PySpark is a powerful tool for handling large-scale data in a distributed fashion and provides a variety of functionalities for data transformation, making it a suitable tool for preparing our datasets for further analysis.

A close-up of a logo

Description automatically generated with low confidence

A screenshot of a computer

Description automatically generated with medium confidence

Run the virtual box first. VirtualBox allows you to create isolated, reproducible environments for Hadoop development and testing, and simulate multi-node clusters on a single machine.

A screenshot of a computer

Description automatically generated with medium confidence

Start the services on Ambari using the browser link<http://sandbox-hdp.hortonworks.com:8080/#/login>

A screenshot of a computer

Description automatically generated

Services are in red alert, that means not started yet.

A screenshot of a computer

Description automatically generated with medium confidence

Services on Ambari started successfully.

A screenshot of a computer

Description automatically generated

Login to HDP sandbox using the browser link [**http://sandbox-hdp.hortonworks.com:4200/**](http://sandbox-hdp.hortonworks.com:4200/). I checked the list of all the directories/data that is present at the location /user/root. Then, I made the directory using ‘mkdir’ on local system and ‘Hadoop fs -mkdir’ for making directory in hdfs.

A screenshot of a computer program

Description automatically generated with medium confidence

Directory created successfully with the name /mrp\_lab

A screenshot of a computer

Description automatically generated

Login to FileZilla and check if the directory created already. I now transferred/drag and dropped the desired files (Dataset1\_flipkart\_product.csv, Dataset2\_Amazon\_Mobiles.csv) into the remote site directory ‘mrp\_lab’

*Host: sftp://127.0.0.1*

*Username: root*

*Password: \*\*\*\*\*\*\**

*Port: 2222*

A picture containing text, font, line, screenshot

Description automatically generated

Now in this step, I changed the directory so that I can go to mrp\_lab folder. Once I enter to that directory/folder. I then copied the files from local system to hdfs using put command while the get command performs the visa-versa operation on the files.

A screenshot of a computer program

Description automatically generated with low confidence

Here in the above snapshot, we can see that by typing the pyspark, it landed me to the spark session where I can start typing my pyspark commands to get the desired outputs.

A screenshot of a computer code

Description automatically generated with low confidence

A screenshot of a computer

Description automatically generated with low confidence

A screenshot of a phone number

Description automatically generated with low confidence

A picture containing text, font, screenshot, document

Description automatically generated

A screenshot of a computer

Description automatically generated with low confidence



A screenshot of a computer

Description automatically generated with low confidence

A screenshot of a computer

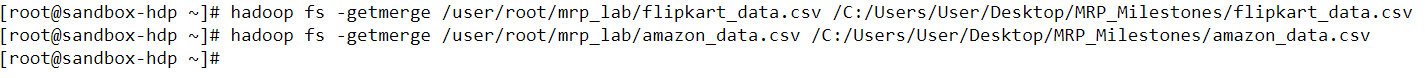
Description automatically generated with low confidence



Wrote DataFrame to HDFS

A picture containing text, screenshot, font

Description automatically generated



Then, used the command hadoop fs -getmerge to merge and download the .csv file to my local machine





Downloaded the files to the local file system on the server. The paths are now Unix-style and point to the server's local file system.

A screenshot of a computer

Description automatically generated

Now I have the files on my server's local file system (/tmp/flipkart\_data.csv and /tmp/amazon\_data.csv). The next step is to get these files to my local Windows machine. For this, I need to use a different tool, because Hadoop's get merge command can't transfer files from the server to my local Windows machine. Hence Used FileZilla tool to transfer the cleaned data files.

**Note: 60% to 70% data has been cleaned while the remaining data will be cleaned and preprocessed in the next phase using python**

**Milestone Phase 2**

**Experimentation with Models**

**In this phase, the focus is on pre-processing and cleaning the remaining uncleaned data. After which the focus is on experimentation part with 3 models CNN, RNN and Transformers**

Python, known for its simplicity, readability, and the availability of powerful data manipulation and machine learning libraries, is our language of choice for model experimentation and prediction tasks.

To accomplish sentiment analysis, we utilize Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs), both types of deep learning models. RNNs' strength lies in their ability to analyze sequential data, like sentences, making them ideal for our project's sentiment analysis. While CNNs are traditionally used for image processing, their application to text data allows us to capture local semantic features, thus enhancing the performance of our sentiment analysis.

In addition, we leverage Transformer models, like BERT, which have revolutionized natural language processing. Transformer models can capture more complex patterns in text data compared to traditional methods, leading to improved performance in tasks such as sentiment analysis.

GitHub Link for all the models experiment and its code:

[*https://github.com/BislaSonal/Sentiment-Analysis-of-Products-Review*](https://github.com/BislaSonal/Sentiment-Analysis-of-Products-Review)