# Configuration Manual

Rishab Rao

## 1 Introduction

This configuration manual will illustrate the environment setup and the configurations for the same to implement ”Prediction and Classification of Electrocardiogram-Signals using Machine Learning using Apache Spark”.

## 2 Environment Specification and Configuration

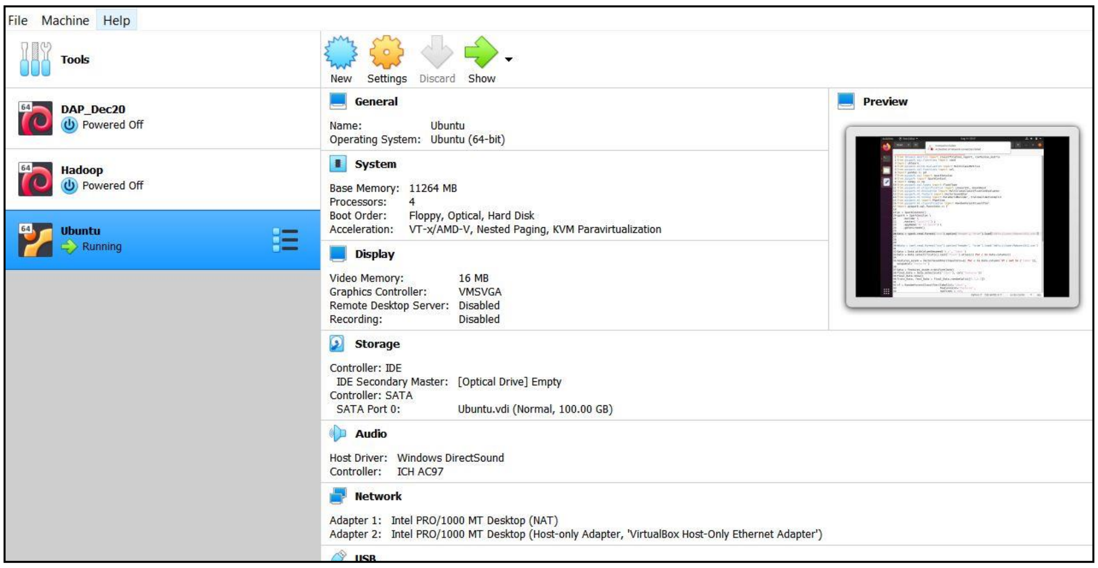
### 2.1 Software Specifications and Requirements

1. To download the Apache ecosystem on a Virtual Machine, VirtualBox application was installed with Ubuntu Operating System.

The download link for VirtualBox is as follows –
https://download.virtualbox.org/virtualbox/6.1.26/VirtualBox-6.1.26-145957-Win.exe

The download link for the Ubuntu OS execution file is as follows –
https://ubuntu.com/download/desktop/thank-you?version=20.04.2.0&architecture=amd64

2. After downloading both the files, the following steps will provide a detailed description to setup VirtualBox and Ubuntu on the VirtualBox. After downloading the execution file for VirtualBox, run the execution file and then open it.



DAP_Dec20 Powered Off
Hadoop Powered Off
Ubuntu Running

**General**
Name: Ubuntu
Operating System: Ubuntu (64-bit)

**System**
Base Memory: 11264 MB
Processors: 4
Boot Order: Floppy, Optical, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging, KVM Paravirtualization

**Display**
Video Memory: 16 MB
Graphics Controller: VMSVGA
Remote Desktop Server: Disabled
Recording: Disabled

**Storage**
Controller: IDE
IDE Secondary Master: [Optical Drive] Empty
Controller: SATA
SATA Port 0: Ubuntu.vdi (Normal, 100.00 GB)

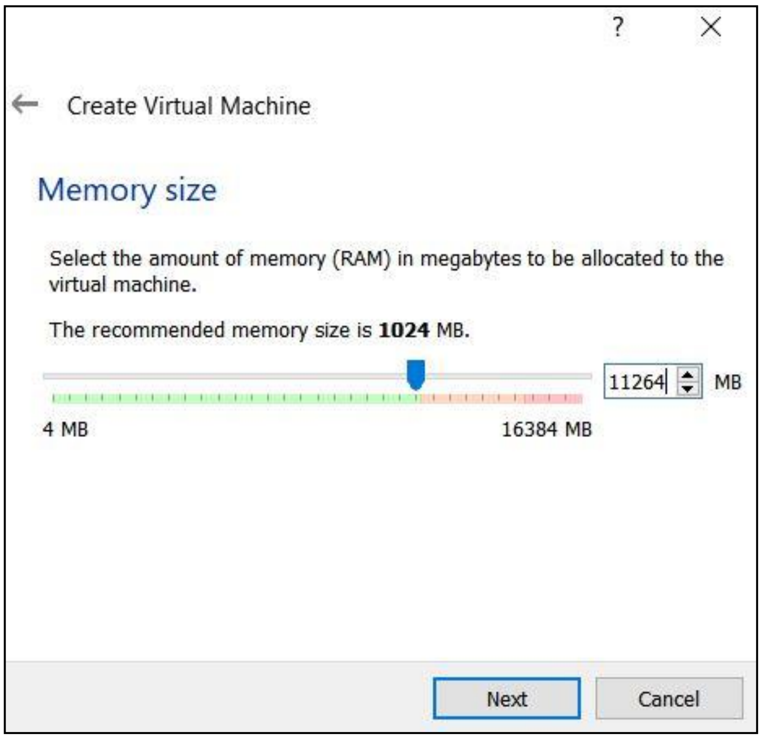
**Audio**
Host Driver: Windows DirectSound
Controller: ICH AC97

**Network**
Adapter 1: Intel PRO/1000 MT Desktop (NAT)
Adapter 2: Intel PRO/1000 MT Desktop (Host-only Adapter, 'VirtualBox Host-Only Ethernet Adapter')

**USB**

3. VirtualBox Setup –

Click the "**New**" button. This is used to create a virtual machine as shown. Then press the "**Create**" button. Assign the Memory Size. In this case 11GB Ram was used.



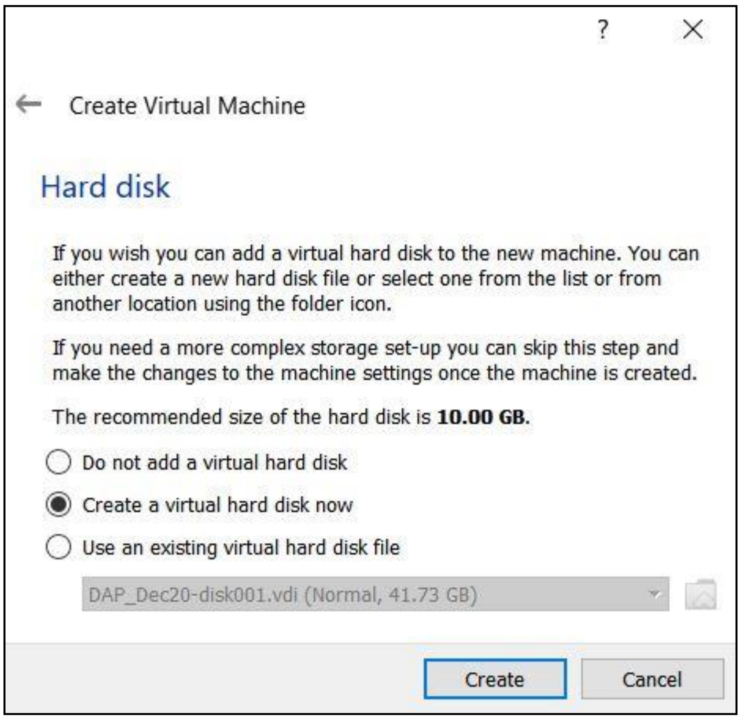
Memory size

Select the amount of memory (RAM) in megabytes to be allocated to the virtual machine.

The recommended memory size is **1024** MB.

11264 MB

The next step is to create the hard disk. Select the option as shown. Then press "**Next**".



Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select one from the list or from another location using the folder icon.

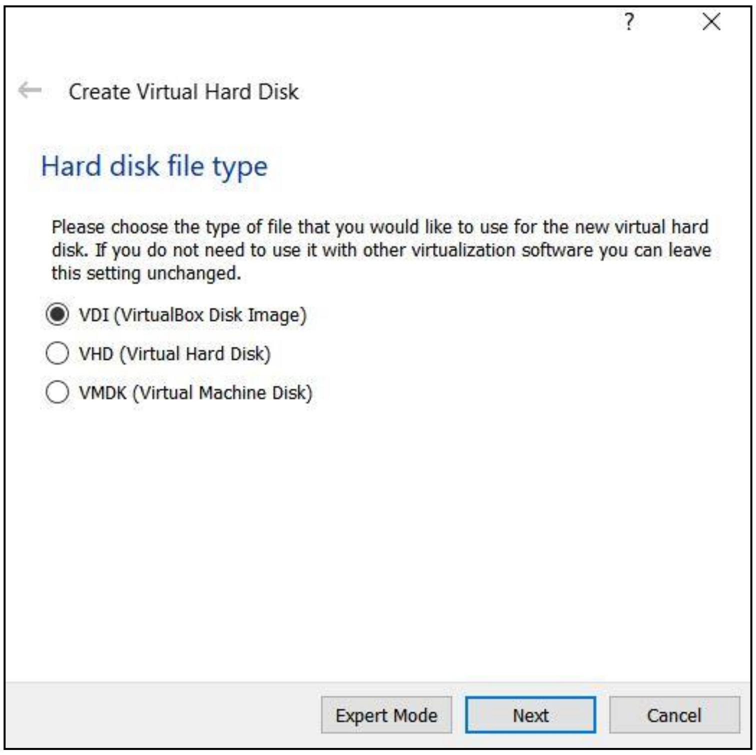
If you need a more complex storage set-up you can skip this step and make the changes to the machine settings once the machine is created.

The recommended size of the hard disk is **10.00 GB**.

- ○ Do not add a virtual hard disk
- ● Create a virtual hard disk now
- ○ Use an existing virtual hard disk file

DAP_Dec20-disk001.vdi (Normal, 41.73 GB)

Select VirtualBox Disk Image and press “Next”

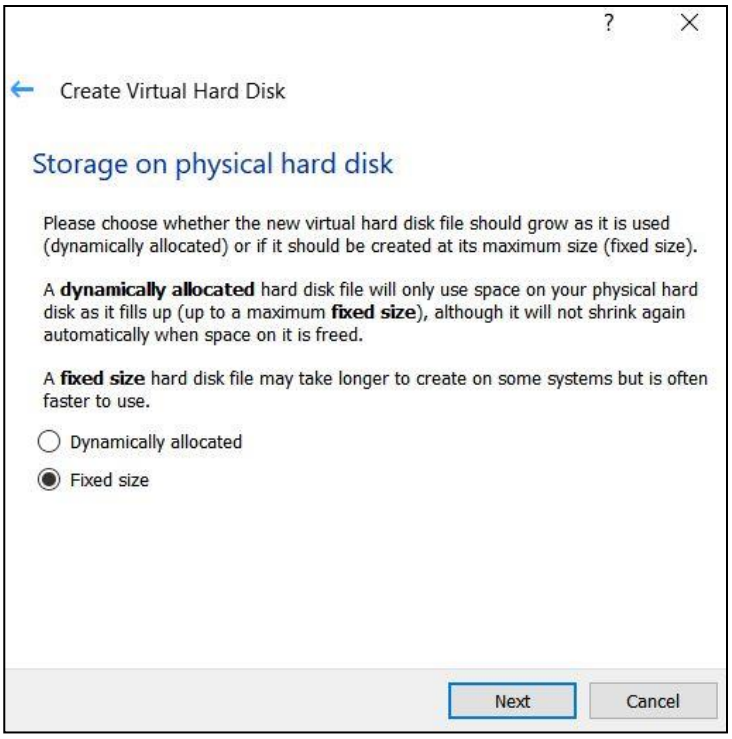

Create Virtual Hard Disk

Hard disk file type

Please choose the type of file that you would like to use for the new virtual hard disk. If you do not need to use it with other virtualization software you can leave this setting unchanged.

[✓ VDI (VirtualBox Disk Image)]
VHD (Virtual Hard Disk)
VMDK (Virtual Machine Disk)

Choose the storage type. If the host device has less storage, then choose “dynamically allocated” option, else, choose “fixed size”


Create Virtual Hard Disk

Storage on physical hard disk

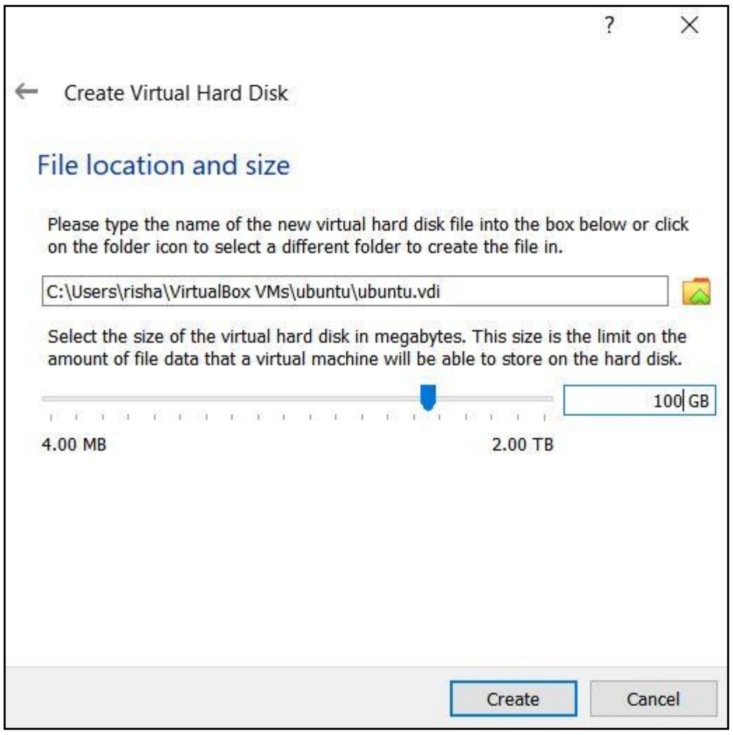
Please choose whether the new virtual hard disk file should grow as it is used (dynamically allocated) or if it should be created at its maximum size (fixed size).

A **dynamically allocated** hard disk file will only use space on your physical hard disk as it fills up (up to a maximum **fixed size**), although it will not shrink again automatically when space on it is freed.

A **fixed size** hard disk file may take longer to create on some systems but is often faster to use.

Dynamically allocated
[✓ Fixed size]

Select the file path for the Virtual Machine to be stored and storage size



Create Virtual Hard Disk

File location and size

Please type the name of the new virtual hard disk file into the box below or click on the folder icon to select a different folder to create the file in.

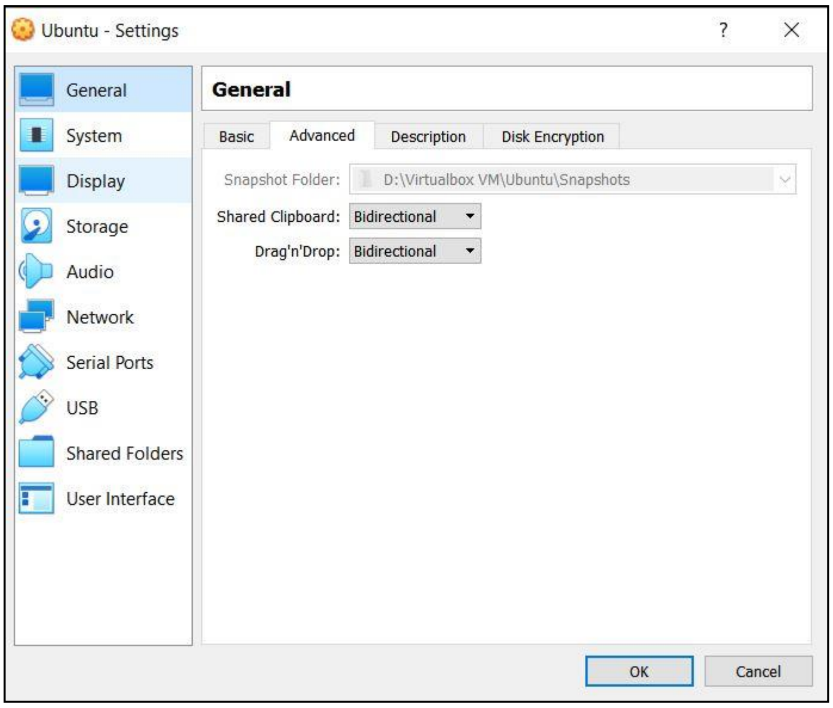
C:\Users\risha\VirtualBox VMs\ubuntu\ubuntu.vdi

Select the size of the virtual hard disk in megabytes. This size is the limit on the amount of file data that a virtual machine will be able to store on the hard disk.

100 GB

After the virtual machine is created, open “Settings” and select the options as shown in the figure

-General

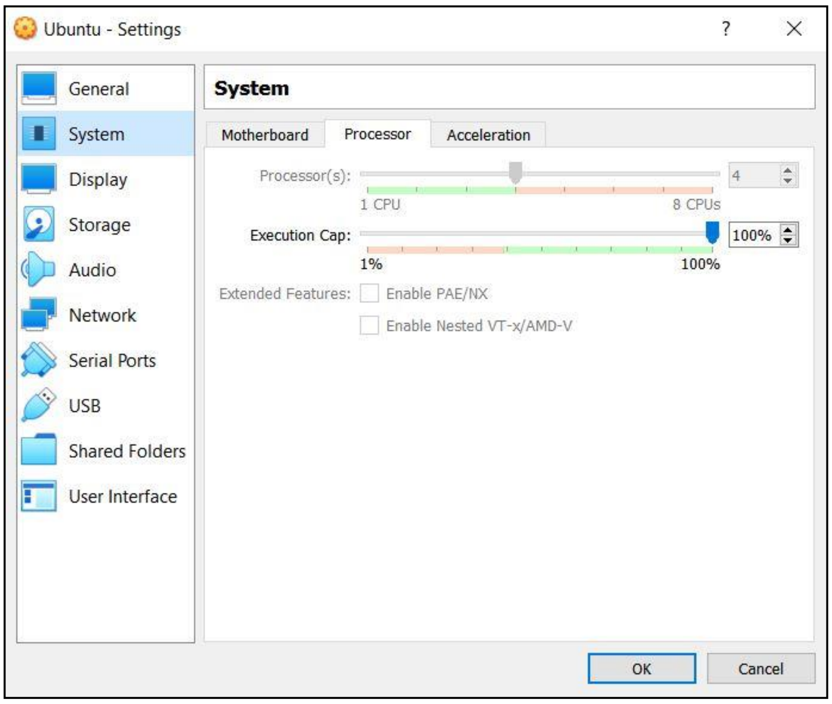


Ubuntu - Settings

General

Snapshot Folder: D:\Virtualbox VM\Ubuntu\Snapshots
Shared Clipboard: Bidirectional
Drag'n'Drop: Bidirectional

-System Storage



System

Motherboard | Processor | Acceleration

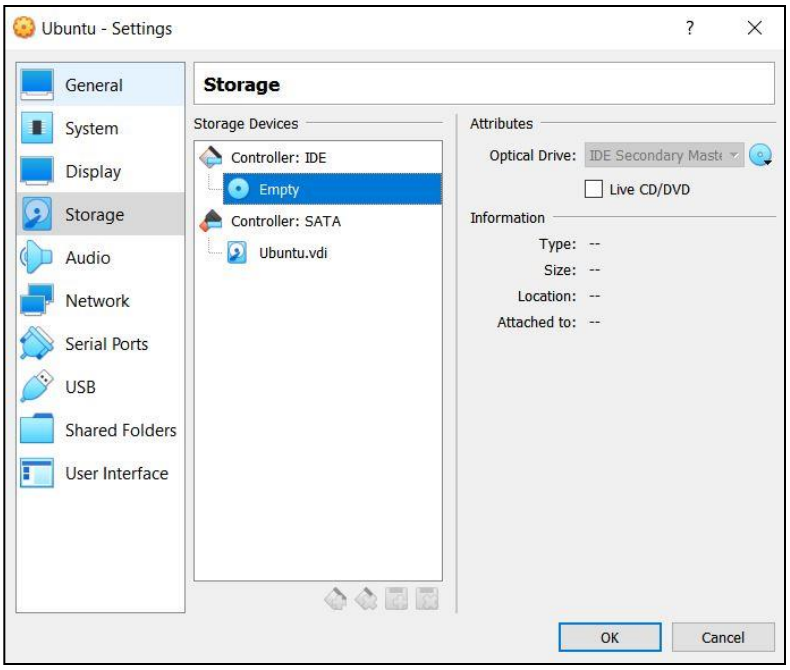
Processor(s): 4 (1 CPU – 8 CPUs)

Execution Cap: 100% (1% – 100%)

Extended Features: ☐ Enable PAE/NX

☐ Enable Nested VT-x/AMD-V

On the top right corner select the “empty disk” option and choose the Ubuntu ISO file previously downloaded.



Storage

Storage Devices

Controller: IDE
- Empty

Controller: SATA
- Ubuntu.vdi

Attributes

Optical Drive: IDE Secondary Maste

☐ Live CD/DVD

Information

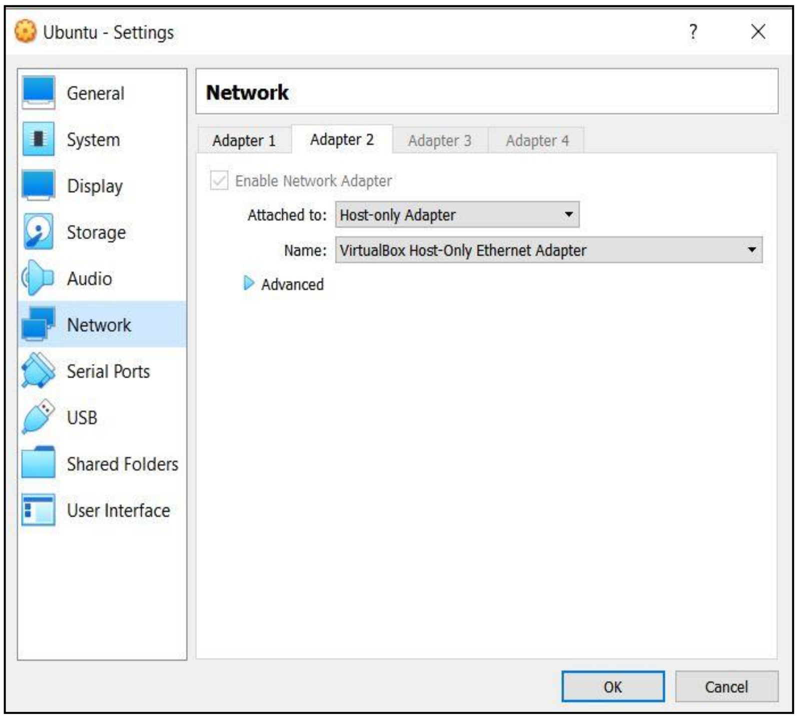
Type: --

Size: --

Location: --

Attached to: --

-Network


Ubuntu - Settings

Network

Adapter 1 | Adapter 2 | Adapter 3 | Adapter 4

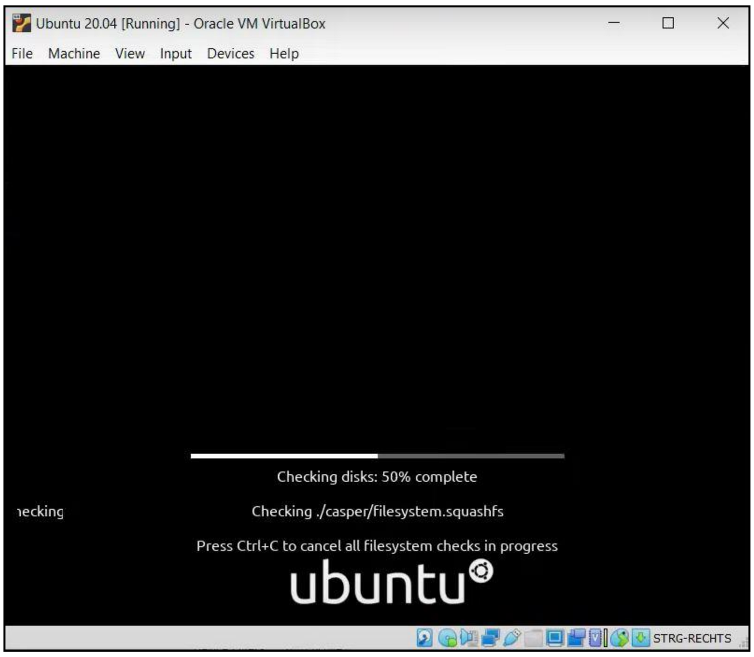
Enable Network Adapter

Attached to: Host-only Adapter

Name: VirtualBox Host-Only Ethernet Adapter

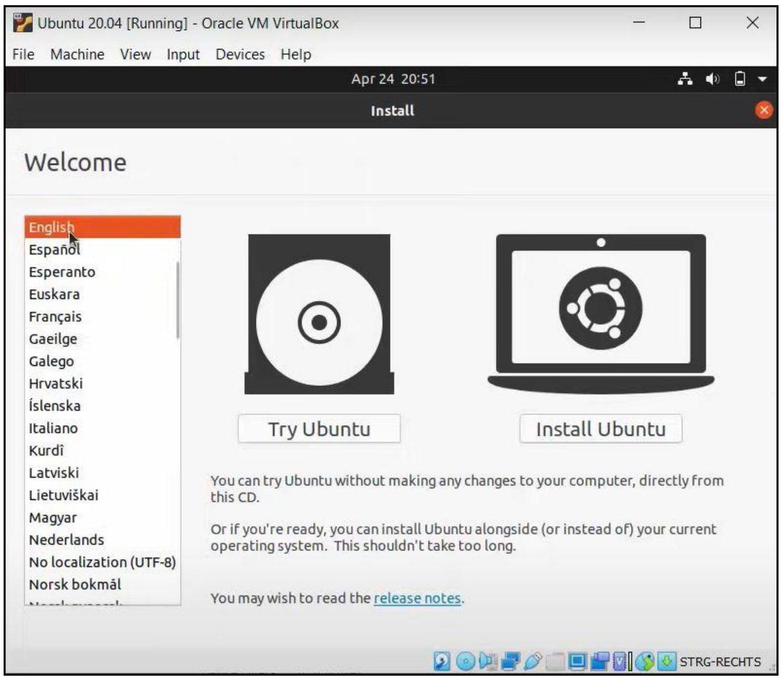
Advanced

4. Ubuntu Setup –
   After VirtualBox settings configuration, close it and select the “Start” option. This will start the virtual machine.


Ubuntu 20.04 [Running] - Oracle VM VirtualBox

```
Checking disks: 50% complete
Checking ./casper/filesystem.squashfs
Press Ctrl+C to cancel all filesystem checks in progress
```

Select the choice of language and click "Install Ubuntu"

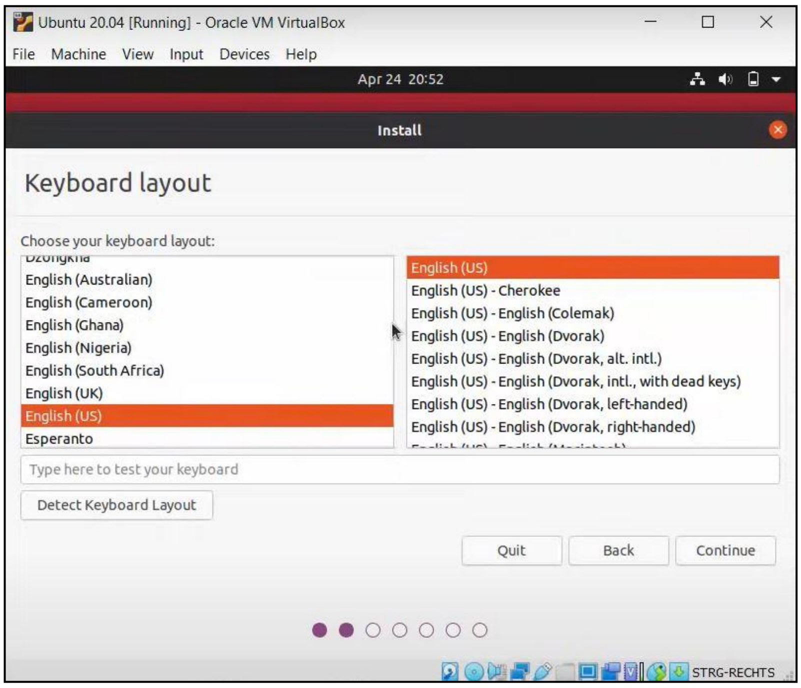


You can try Ubuntu without making any changes to your computer, directly from this CD.

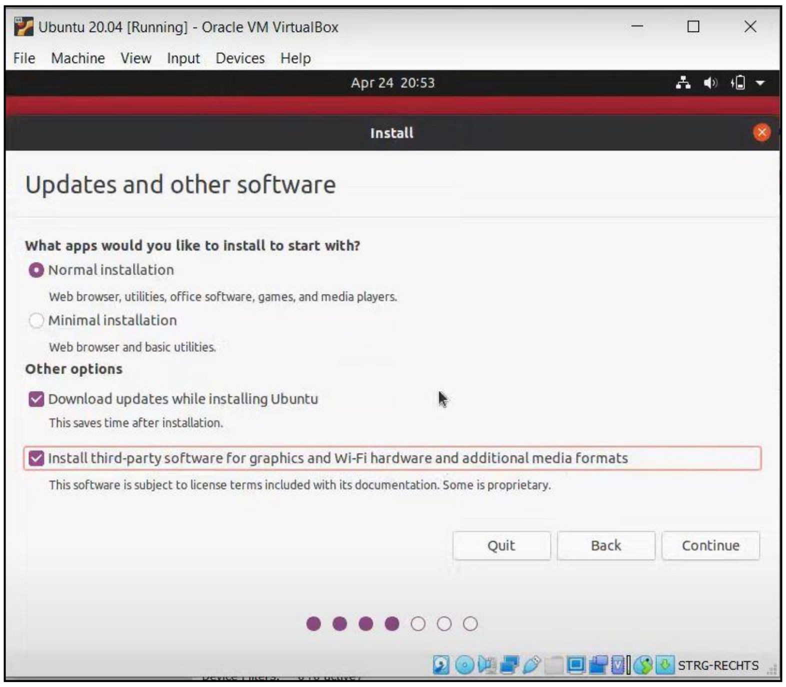
Or if you're ready, you can install Ubuntu alongside (or instead of) your current operating system. This shouldn't take too long.

You may wish to read the release notes.

Select the keyboard layout style


Ubuntu 20.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Apr 24 20:52
Install
Keyboard layout
Choose your keyboard layout:
Dzongkha
English (Australian)
English (Cameroon)
English (Ghana)
English (Nigeria)
English (South Africa)
English (UK)
English (US)
Esperanto
English (US)
English (US) - Cherokee
English (US) - English (Colemak)
English (US) - English (Dvorak)
English (US) - English (Dvorak, alt. intl.)
English (US) - English (Dvorak, intl., with dead keys)
English (US) - English (Dvorak, left-handed)
English (US) - English (Dvorak, right-handed)
Type here to test your keyboard
Detect Keyboard Layout
Quit
Back
Continue
STRG-RECHTS

Select all the checkboxes except Minimal installation.


Ubuntu 20.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Apr 24 20:53
Install

Updates and other software

What apps would you like to install to start with?

[✓ Normal installation]

Web browser, utilities, office software, games, and media players.

Minimal installation

Web browser and basic utilities.

Other options

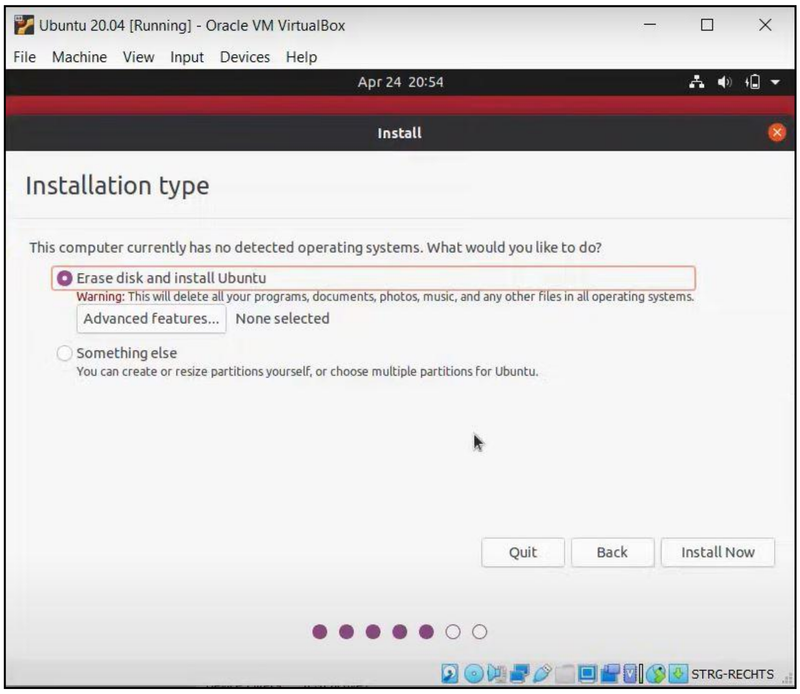
[✓ Download updates while installing Ubuntu]

This saves time after installation.

[✓ Install third-party software for graphics and Wi-Fi hardware and additional media formats]

This software is subject to license terms included with its documentation. Some is proprietary.

Choose the installation type.


Ubuntu 20.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Apr 24 20:54
Install

Installation type

This computer currently has no detected operating systems. What would you like to do?

[✓ Erase disk and install Ubuntu]

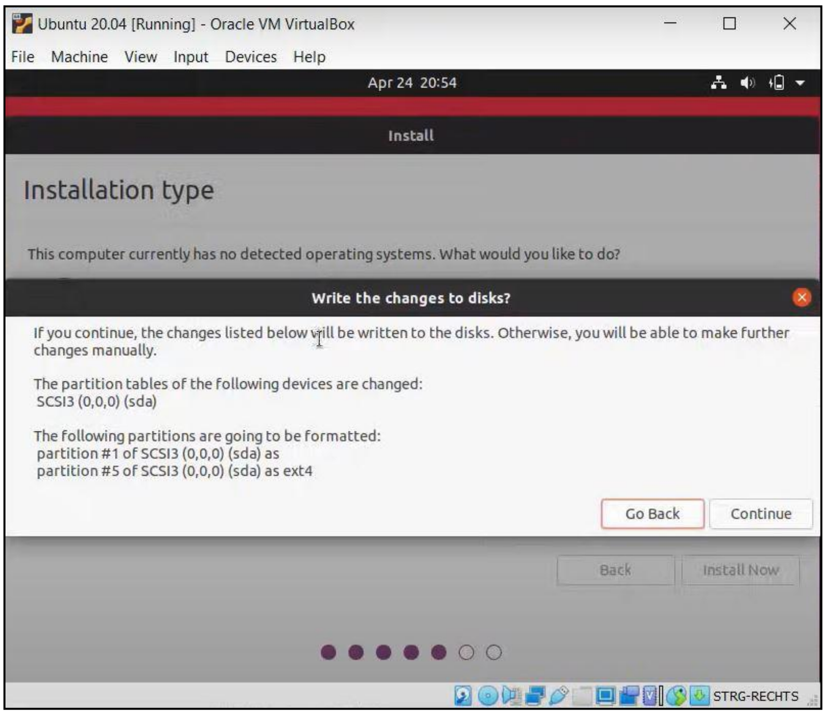
Warning: This will delete all your programs, documents, photos, music, and any other files in all operating systems.

Advanced features... None selected

Something else

You can create or resize partitions yourself, or choose multiple partitions for Ubuntu.

The following screen will request for disk formatting This will **not** make any changes to the host system. The clean-up and formatting are **only** for the virtual machine.


Ubuntu 20.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Apr 24 20:54
Install
Back
Install Now
STRG-RECHTS

Installation type

This computer currently has no detected operating systems. What would you like to do?

Write the changes to disks?

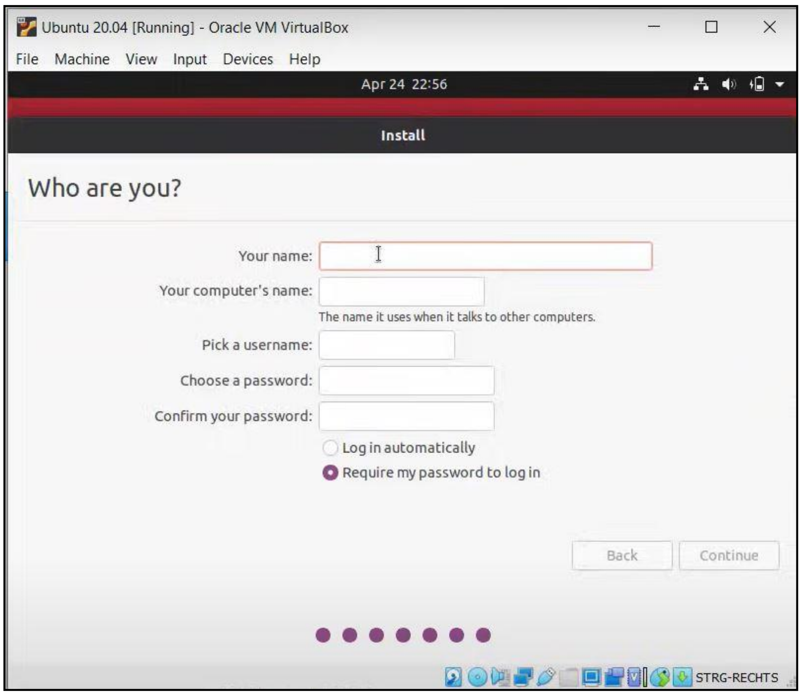
If you continue, the changes listed below will be written to the disks. Otherwise, you will be able to make further changes manually.

The partition tables of the following devices are changed:
SCSI3 (0,0,0) (sda)

The following partitions are going to be formatted:
partition #1 of SCSI3 (0,0,0) (sda) as
partition #5 of SCSI3 (0,0,0) (sda) as ext4

Go Back | Continue

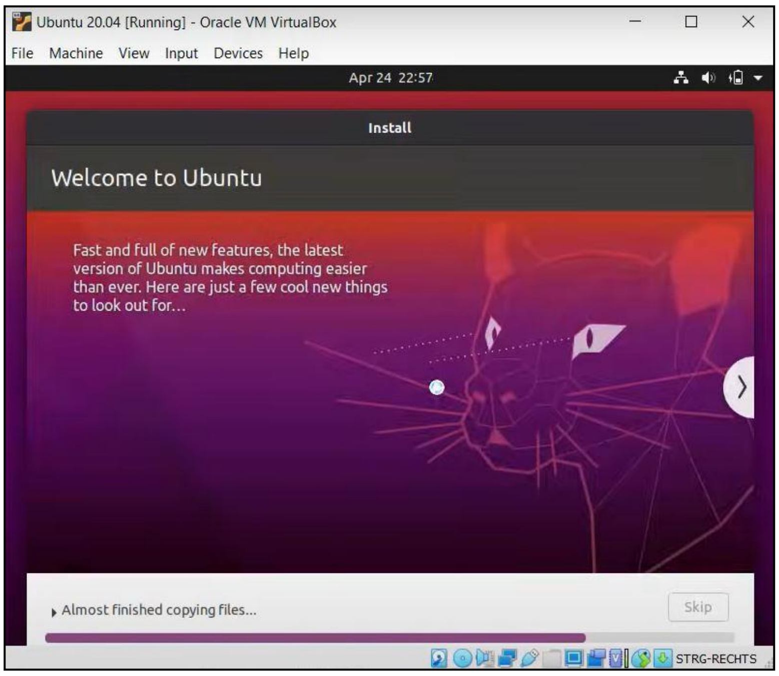
Enter the username and password. **DO NOT FORGET THE LOGIN DETAILS. SAVE IT SOMEWHERE IF NECESSARY**


Ubuntu 20.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Apr 24 22:56
Install
Back
Continue
STRG-RECHTS

Who are you?

Your name:
Your computer's name:
The name it uses when it talks to other computers.
Pick a username:
Choose a password:
Confirm your password:
Log in automatically [✓ Require my password to log in]

The installation will commence. It will take some time for the OS to completely installed.

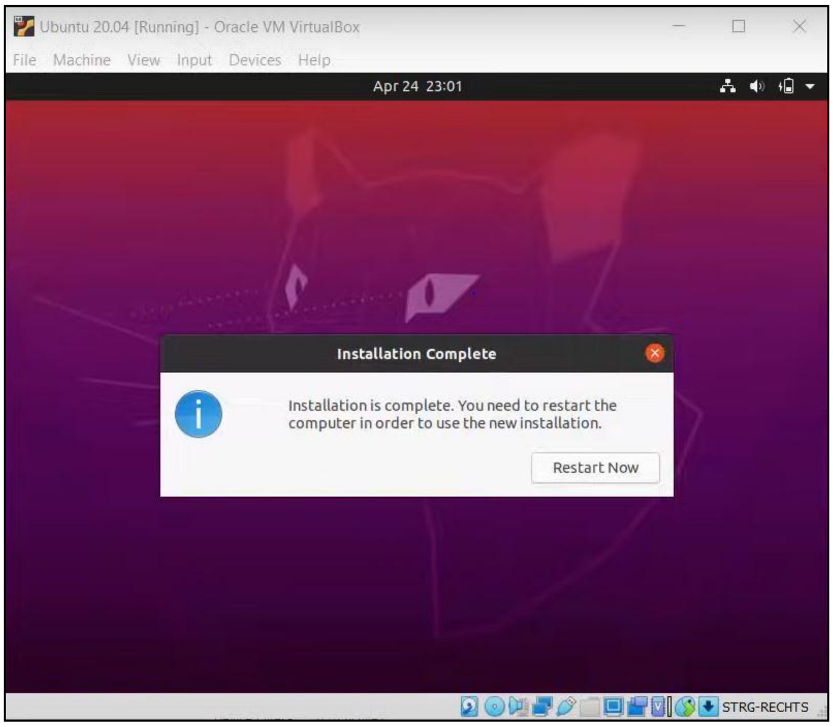

Ubuntu 20.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Apr 24 22:57
Install
Skip
STRG-RECHTS

Welcome to Ubuntu

Fast and full of new features, the latest version of Ubuntu makes computing easier than ever. Here are just a few cool new things to look out for...

Almost finished copying files...

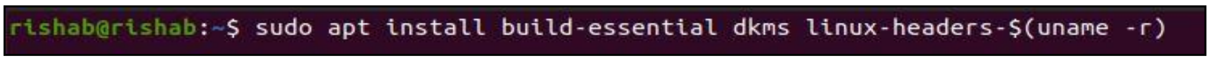
Post installation the virtual machine will request for a reboot. Press “Restart Now”


Ubuntu 20.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Apr 24 23:01
Restart Now
STRG-RECHTS

Installation Complete

Installation is complete. You need to restart the computer in order to use the new installation.

After the system restarts, enter the login details and open the terminal by pressing "Ctrl + Alt + t" and enter the following command as shown in the figure. This will install all essential Ubuntu packages.



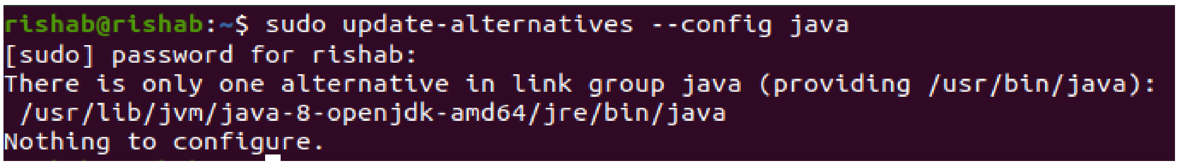
```
rishab@rishab:~$ sudo apt install build-essential dkms linux-headers-$(uname -r)
```

## 2.2 Hadoop Installation

1. Open the Terminal (Ctrl + Alt + t) and update Ubuntu OS using the commands –
   **sudo apt-get update**
   **sudo apt-get upgrade**

2. Install Java-8 jdk using the following command –
   **sudo apt install openjdk-8-jdk**

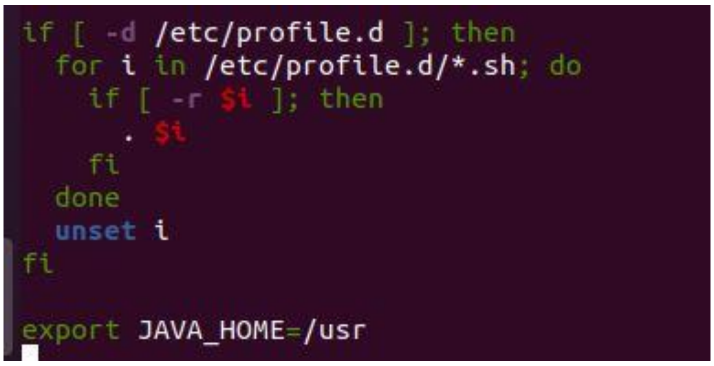
3. Check where Java is installed (Java Path). This is required to configure Hadoop.



```
rishab@rishab:~$ sudo update-alternatives --config java
[sudo] password for rishab:
There is only one alternative in link group java (providing /usr/bin/java):
 /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java
Nothing to configure.
```

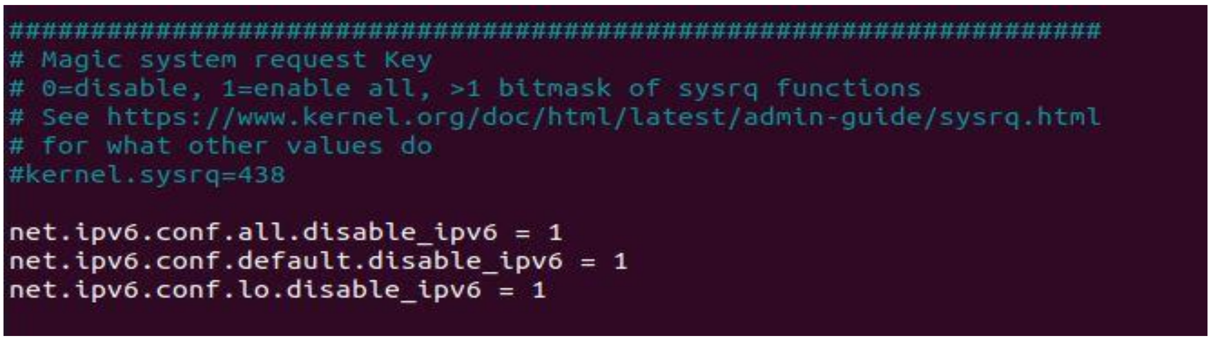
4. Open the **/etc/profile** file using the command –
   **sudo nano /etc/profile**

   Add the following lines of code to set the path for Java. And then save the file using the command –
   **source /etc/profile**



```
if [ -d /etc/profile.d ]; then
  for i in /etc/profile.d/*.sh; do
    if [ -r $i ]; then
      . $i
    fi
  done
  unset i
fi

export JAVA_HOME=/usr
```

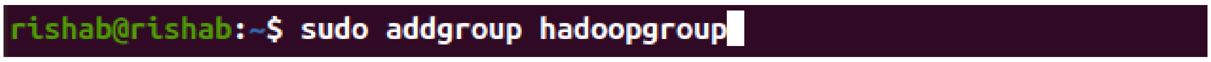
5. Open the sysctl.conf file using the command –
   **sudo nano /etc/profile/sysctl.conf**
   Add the following lines of code.



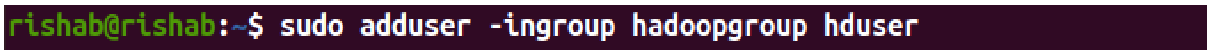
```
###################################################################
# Magic system request Key
# 0=disable, 1=enable all, >1 bitmask of sysrq functions
# See https://www.kernel.org/doc/html/latest/admin-guide/sysrq.html
# for what other values do
#kernel.sysrq=438

net.ipv6.conf.all.disable_ipv6 = 1
net.ipv6.conf.default.disable_ipv6 = 1
net.ipv6.conf.lo.disable_ipv6 = 1
```

6. To make these changes to take immediate effect, reboot the system.

7. Configure the SSH for Hadoop. Create a user group for Hadoop using the following commands –

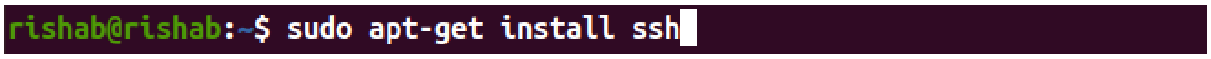


```
rishab@rishab:~$ sudo addgroup hadoopgroup
```



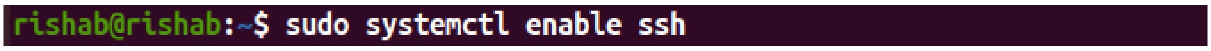
```
rishab@rishab:~$ sudo adduser -ingroup hadoopgroup hduser
```

8. Install SSH



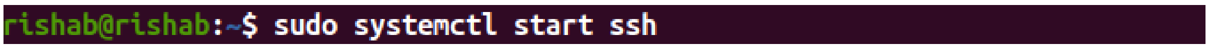
```
rishab@rishab:~$ sudo apt-get install ssh
```

Enable SSH



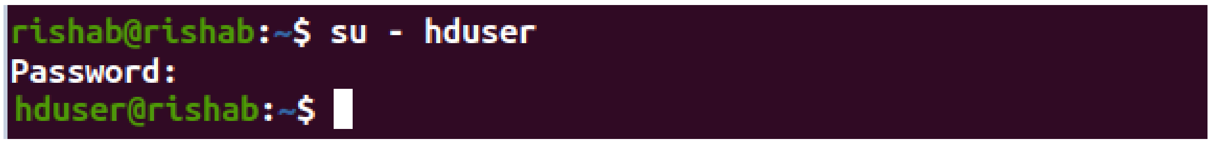
```
rishab@rishab:~$ sudo systemctl enable ssh
```

Start SSH



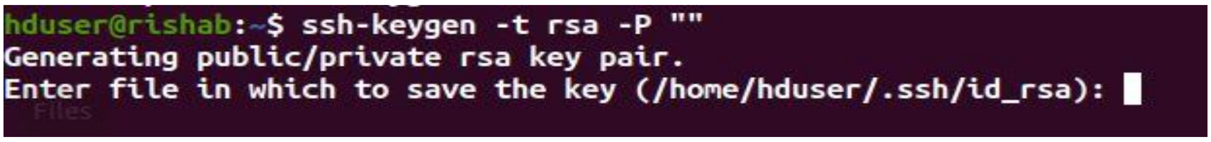
```
rishab@rishab:~$ sudo systemctl start ssh
```

9. Switch user by typing the following command



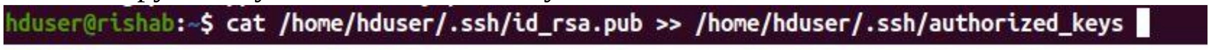
```
rishab@rishab:~$ su - hduser
Password:
hduser@rishab:~$
```

10. Generating public/private keys folder



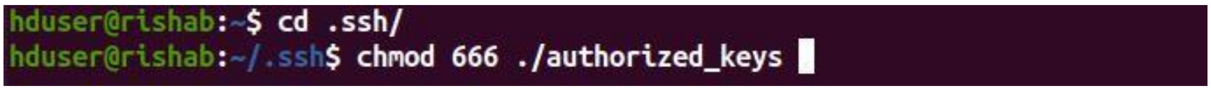
```
hduser@rishab:~$ ssh-keygen -t rsa -P ""
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hduser/.ssh/id_rsa):
```

11. Copy the keys to the authorized keys folder



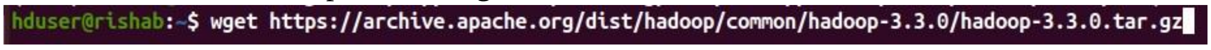
```
hduser@rishab:~$ cat /home/hduser/.ssh/id_rsa.pub >> /home/hduser/.ssh/authorized_keys
```

12. Change the permissions for the keys



```
hduser@rishab:~$ cd .ssh/
hduser@rishab:~/.ssh$ chmod 666 ./authorized_keys
```

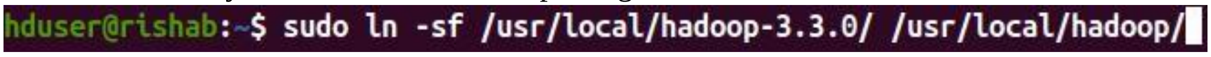
13. Download Hadoop 3.3.0 using the command



```
hduser@rishab:~$ wget https://archive.apache.org/dist/hadoop/common/hadoop-3.3.0/hadoop-3.3.0.tar.gz
```

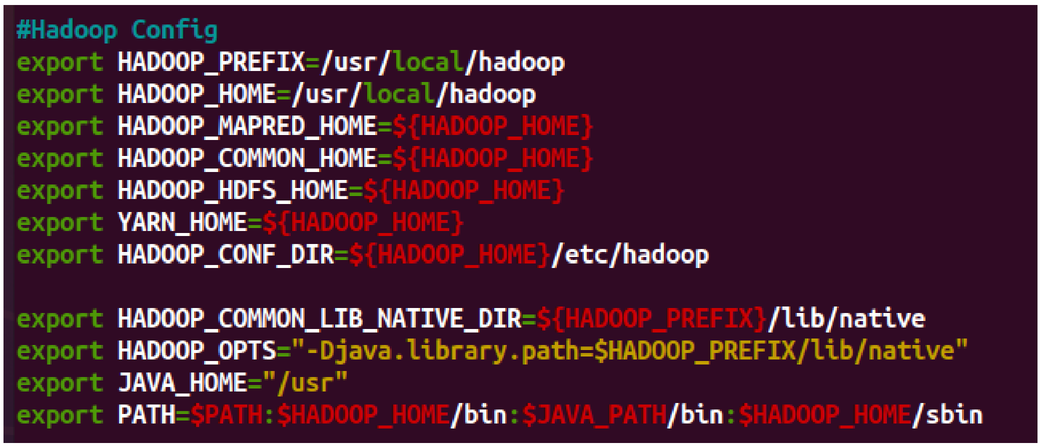
14. Extract the contents of the Hadoop zip file using the command -
    **tar -xvf hadoop 3.3.0.tar.gz**
15. Move the extracted file into **cd/usr/local.** Change ownership using the following command -
    **sudo chown -R hduser:hadoopgroup /usr/local/Hadoop-3.3.0**

16. Create a symbolic link with hadoop using the command



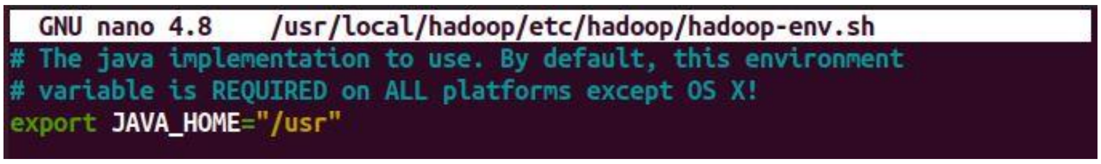
```
hduser@rishab:~$ sudo ln -sf /usr/local/hadoop-3.3.0/ /usr/local/hadoop/
```

17. Open the **bashrc file** using the command sudo nano ./.bashrc and configure it using the following lines of code. Then save the file using **source**



```
#Hadoop Config
export HADOOP_PREFIX=/usr/local/hadoop
export HADOOP_HOME=/usr/local/hadoop
export HADOOP_MAPRED_HOME=${HADOOP_HOME}
export HADOOP_COMMON_HOME=${HADOOP_HOME}
export HADOOP_HDFS_HOME=${HADOOP_HOME}
export YARN_HOME=${HADOOP_HOME}
export HADOOP_CONF_DIR=${HADOOP_HOME}/etc/hadoop

export HADOOP_COMMON_LIB_NATIVE_DIR=${HADOOP_PREFIX}/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_PREFIX/lib/native"
export JAVA_HOME="/usr"
export PATH=$PATH:$HADOOP_HOME/bin:$JAVA_PATH/bin:$HADOOP_HOME/sbin
```

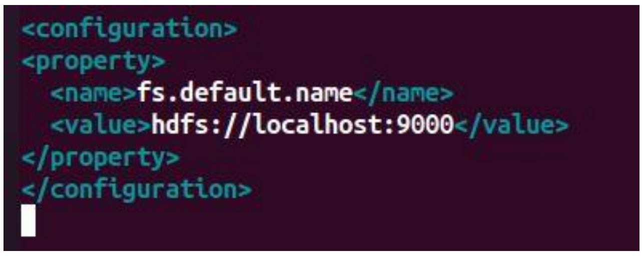
18. Open the Hadoop environment to set the java path



```
  GNU nano 4.8     /usr/local/hadoop/etc/hadoop/hadoop-env.sh
# The java implementation to use. By default, this environment
# variable is REQUIRED on ALL platforms except OS X!
export JAVA_HOME="/usr"
```

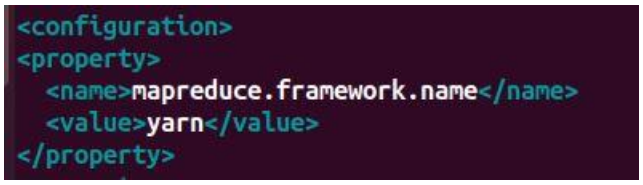
19. Navigate to the hadoop directory using **cd /usr/local/hadoop/etc/hadoop** and navigate the .xml files and add the following –

    **sudo nano core-site.xml**



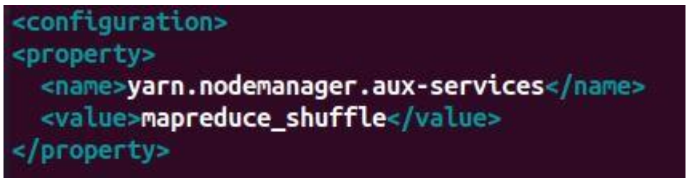
```
<configuration>
<property>
  <name>fs.default.name</name>
  <value>hdfs://localhost:9000</value>
</property>
</configuration>
```

**sudo nano mapred-site.xml**



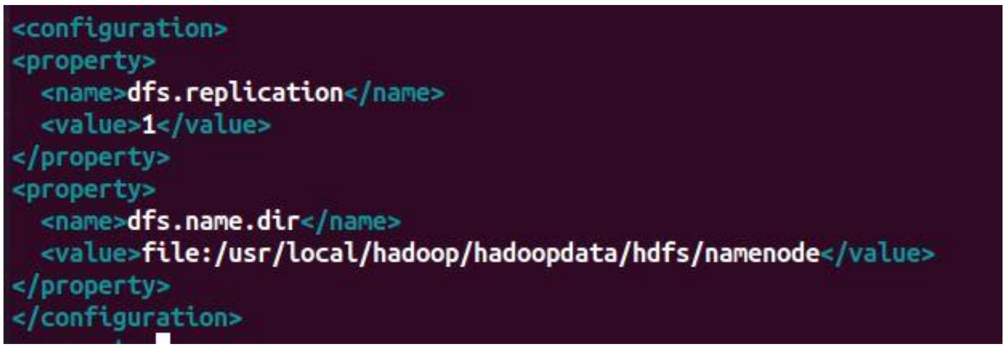
```
<configuration>
<property>
  <name>mapreduce.framework.name</name>
  <value>yarn</value>
</property>
```

**sudo nano yarn-site.xml**



```
<configuration>
<property>
  <name>yarn.nodemanager.aux-services</name>
  <value>mapreduce_shuffle</value>
</property>
```

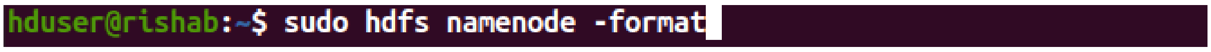
**sudo nano hdfs-site.xml**



```
<configuration>
<property>
  <name>dfs.replication</name>
  <value>1</value>
</property>
<property>
  <name>dfs.name.dir</name>
  <value>file:/usr/local/hadoop/hadoopdata/hdfs/namenode</value>
</property>
</configuration>
```

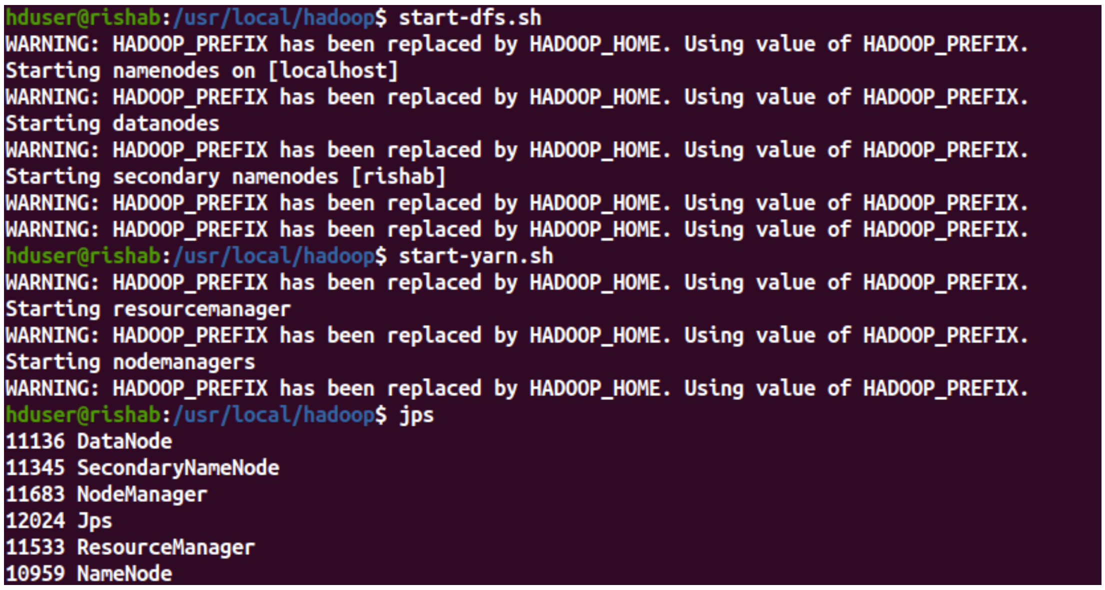
20. Navigate to the home directory using the command **cd**

21. Format the namenode



```
hduser@rishab:~$ sudo hdfs namenode -format
```

22. Start Hadoop services by first navigating to the Hadoop directory
**cd /usr/local/hadoop**



```
hduser@rishab:/usr/local/hadoop$ start-dfs.sh
WARNING: HADOOP_PREFIX has been replaced by HADOOP_HOME. Using value of HADOOP_PREFIX.
Starting namenodes on [localhost]
WARNING: HADOOP_PREFIX has been replaced by HADOOP_HOME. Using value of HADOOP_PREFIX.
Starting datanodes
WARNING: HADOOP_PREFIX has been replaced by HADOOP_HOME. Using value of HADOOP_PREFIX.
Starting secondary namenodes [rishab]
WARNING: HADOOP_PREFIX has been replaced by HADOOP_HOME. Using value of HADOOP_PREFIX.
WARNING: HADOOP_PREFIX has been replaced by HADOOP_HOME. Using value of HADOOP_PREFIX.
hduser@rishab:/usr/local/hadoop$ start-yarn.sh
WARNING: HADOOP_PREFIX has been replaced by HADOOP_HOME. Using value of HADOOP_PREFIX.
Starting resourcemanager
WARNING: HADOOP_PREFIX has been replaced by HADOOP_HOME. Using value of HADOOP_PREFIX.
Starting nodemanagers
WARNING: HADOOP_PREFIX has been replaced by HADOOP_HOME. Using value of HADOOP_PREFIX.
hduser@rishab:/usr/local/hadoop$ jps
11136 DataNode
11345 SecondaryNameNode
11683 NodeManager
12024 Jps
11533 ResourceManager
10959 NameNode
```

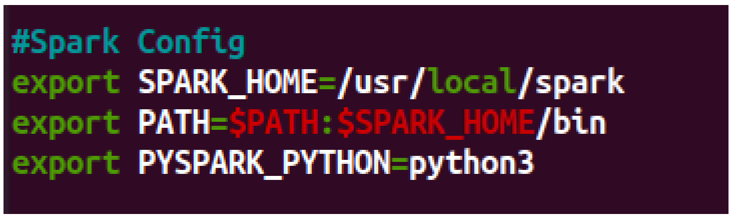
## 2.3 Apache Spark Installation

1. Download and extract Apache Spark 3.1.2 in **cd /usr/local** using the link https://www.apache.org/dyn/closer.lua/spark/spark-3.1.2/spark-3.1.2-bin-hadoop3.2.tgz

2. Create a symbolic link of spark using the command
   **sudo ln –s spark-3.1.2-bin-hadoop3.2 spark**

3. Change the ownership of the Spark directory to Hadoop group and ownership to hduser as seen in section 2.3 (**sudo chown -R hduser:hadoopgroup spark/**)

4. Configure the .bashrc file using the following lines of code and save it using **source**



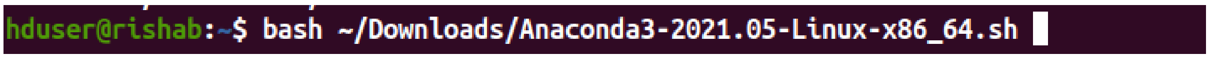
```
#Spark Config
export SPARK_HOME=/usr/local/spark
export PATH=$PATH:$SPARK_HOME/bin
export PYSPARK_PYTHON=python3
```

5. Navigate to **cd/usr/local/spark** and start the services using **sbin/start-master.sh**

## 2.4 Anaconda Installation

1. Download Anaconda using the following link:
   https://repo.anaconda.com/archive/Anaconda3-2021.05-Linux-x86_64.sh

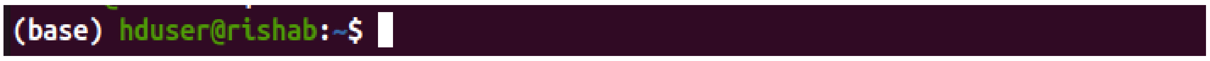
2. Once the download is complete, extract it using the command:



```
hduser@rishab:~$ bash ~/Downloads/Anaconda3-2021.05-Linux-x86_64.sh
```

   Once the installation commences, read the terms and conditions, and answer **"yes"** to everything. Keep in mind to answer "yes" when the installer requests permission to prepend anaconda3 path to the .bashrc file.

3. Then source the .bashrc file for the changes to take immediate effect
   The terminal should look like the following:

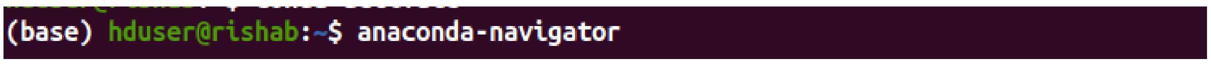


```
(base) hduser@rishab:~$
```

4. To implement Deep Neural Networks, create a separate environment on Anaconda and install TensorFlow and Keras.

5. First step is to install conda forge. This is used to download and install TensorFlow. Use the following commands on the terminal after installing Anaconda:
   **conda config –add channels conda-forge**
   **conda config –set channel_priority strict**

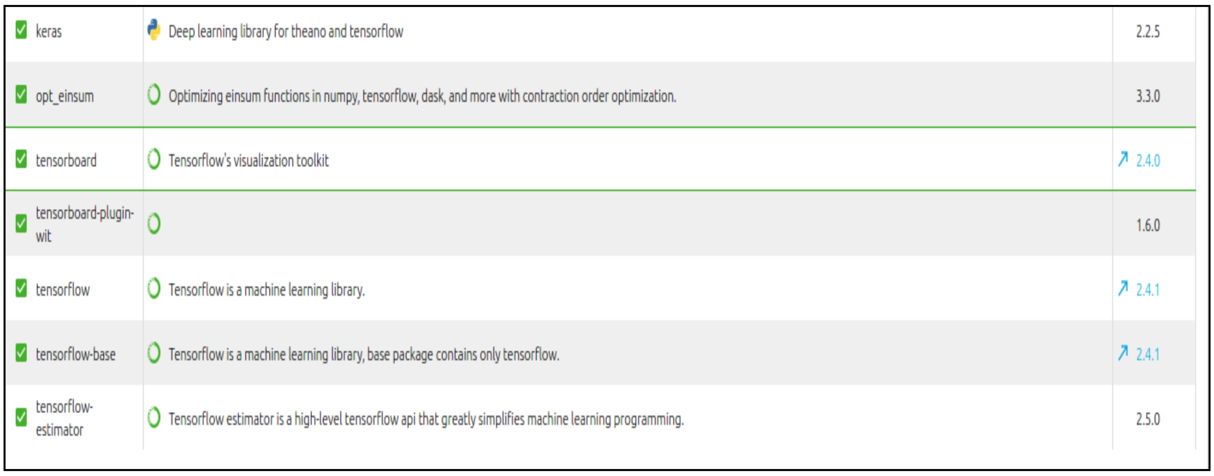
6. Before installing Keras and TensorFlow a new environment must be created. Open anaconda navigator



```
(base) hduser@rishab:~$ anaconda-navigator
```

Go to environments and create a new environment named "TensorFlow".

7. After the new environment is created, select the TensorFlow environment and install TensorFlow Package



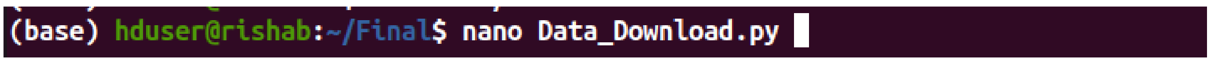
| | | | |
|---|---|---|---|
| ✓ | keras | Deep learning library for theano and tensorflow | 2.2.5 |
| ✓ | opt_einsum | Optimizing einsum functions in numpy, tensorflow, dask, and more with contraction order optimization. | 3.3.0 |
| ✓ | tensorboard | Tensorflow's visualization toolkit | ↗ 2.4.0 |
| ✓ | tensorboard-plugin-wit | | 1.6.0 |
| ✓ | tensorflow | Tensorflow is a machine learning library. | ↗ 2.4.1 |
| ✓ | tensorflow-base | Tensorflow is a machine learning library, base package contains only tensorflow. | ↗ 2.4.1 |
| ✓ | tensorflow-estimator | Tensorflow estimator is a high-level tensorflow api that greatly simplifies machine learning programming. | 2.5.0 |

## 3 Implementation of the research project

### 3.1 Download the dataset

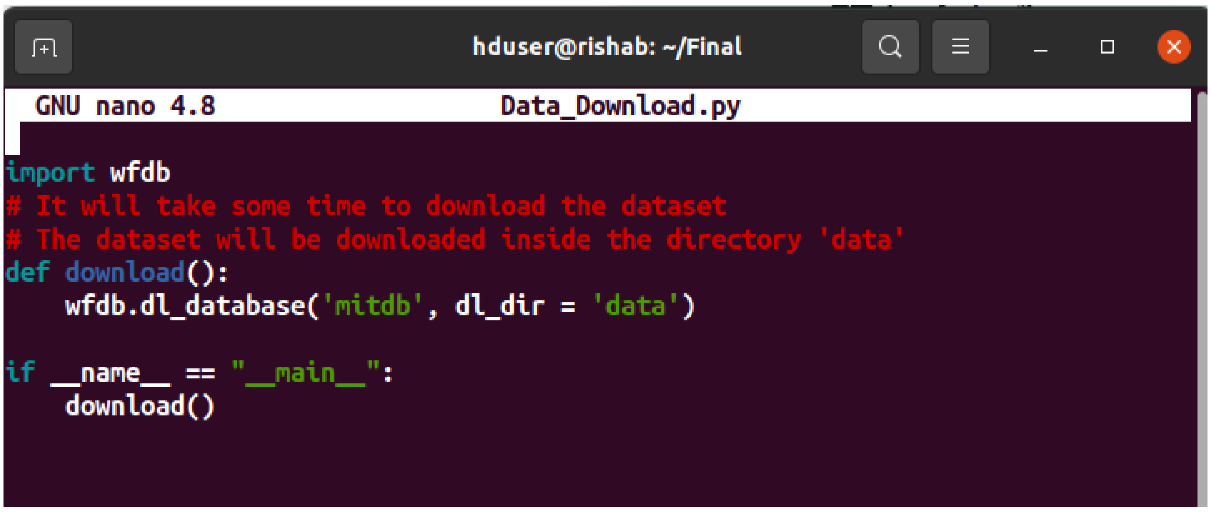
a) Download the wfdb package using the following line on the terminal:
**pip install wfdb**

b) Open a new python file using the terminal as shown



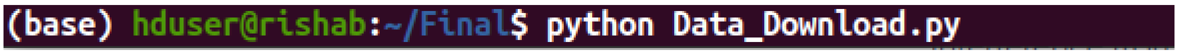
```
(base) hduser@rishab:~/Final$ nano Data_Download.py
```

c) Type the following lines of code to download the data



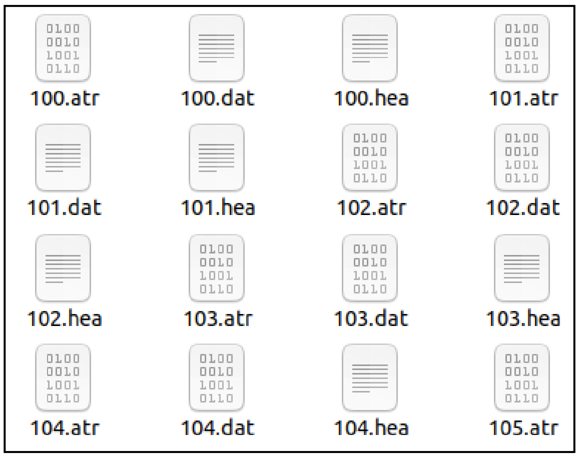
```
GNU nano 4.8                    Data_Download.py

import wfdb
# It will take some time to download the dataset
# The dataset will be downloaded inside the directory 'data'
def download():
    wfdb.dl_database('mitdb', dl_dir = 'data')

if __name__ == "__main__":
    download()
```

d) Save the file and exit the text editor. Run the python script using the following line:



```
(base) hduser@rishab:~/Final$ python Data_Download.py
```

e) The data will be downloaded under the folder name "Data". The contents of the folder will look like the following:



```
100.atr  100.dat  100.hea  101.atr
101.dat  101.hea  102.atr  102.dat
102.hea  103.atr  103.dat  103.hea
104.atr  104.dat  104.hea  105.atr
```

## 3.2 Data Preprocessing, Transformation and Exploratory Data Analysis

Run the python file using the command - **python preprocess.py**

## 3.3 Experiment 1 – Local implementation of the classification and prediction models

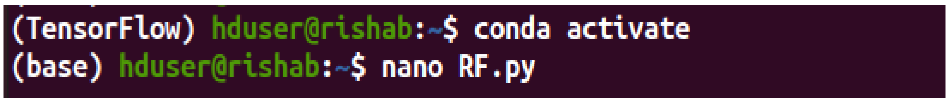
### 3.3.1 Implementation of Deep Neural Network

a) On the terminal change the environment using **conda activate TensorFlow.**
In this research project the name of the environment is TensorFlow.

b) Run the program using **python DNN.py**

### 3.3.2 Implementation of Random Forest model

a) On the terminal return to base environment and then open a python file to implement RF.

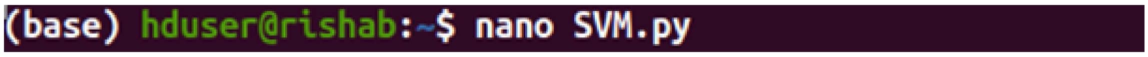


```
(TensorFlow) hduser@rishab:~$ conda activate
(base) hduser@rishab:~$ nano RF.py
```

b) Run the program using **python RF.py**

### 3.3.3 Implementation of Support Vector Machine model

a) Open a python file on the terminal to implement SVM.

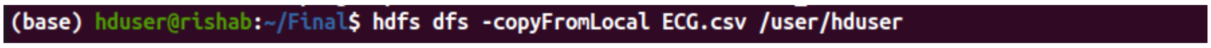


```
(base) hduser@rishab:~$ nano SVM.py
```

b) Execute the file using **python SVM.py**

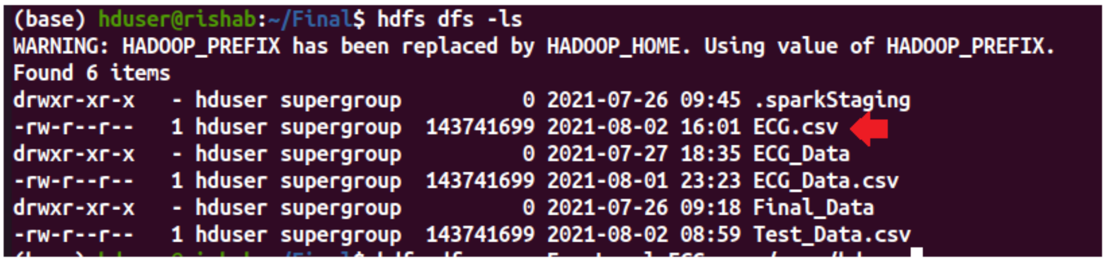
## 3.4 Experiment 2 – Implementation of the classification and prediction models on Apache Spark

The first step in this experiment is to load the data into Hadoop Distributed File System (HDFS).



```
(base) hduser@rishab:~/Final$ hdfs dfs -copyFromLocal ECG.csv /user/hduser
```

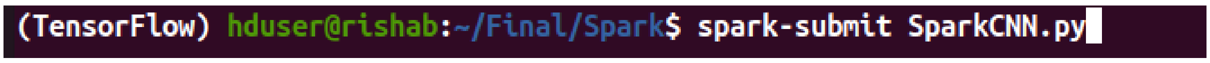
To check whether the dataset has been successfully loaded or not type the following command:



```
(base) hduser@rishab:~/Final$ hdfs dfs -ls
WARNING: HADOOP_PREFIX has been replaced by HADOOP_HOME. Using value of HADOOP_PREFIX.
Found 6 items
drwxr-xr-x   - hduser supergroup          0 2021-07-26 09:45 .sparkStaging
-rw-r--r--   1 hduser supergroup  143741699 2021-08-02 16:01 ECG.csv
drwxr-xr-x   - hduser supergroup          0 2021-07-27 18:35 ECG_Data
-rw-r--r--   1 hduser supergroup  143741699 2021-08-01 23:23 ECG_Data.csv
drwxr-xr-x   - hduser supergroup          0 2021-07-26 09:18 Final_Data
-rw-r--r--   1 hduser supergroup  143741699 2021-08-02 08:59 Test_Data.csv
```

### 3.4.1 Implementation of Deep Neural Network model

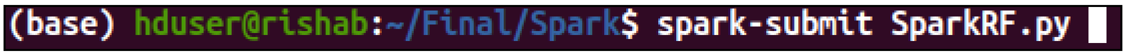
Run the python file using **spark-submit.**



```
(TensorFlow) hduser@rishab:~/Final/Spark$ spark-submit SparkCNN.py
```

### 3.4.2 Implementation of Random Forest model

Run the program using the following command



```
(base) hduser@rishab:~/Final/Spark$ spark-submit SparkRF.py
```

### 3.4.3 Implementation of Support Vector Machine model

Run the program using the following command –

**>>spark-submit SparkSVM.py**

# 4 Conclusion

All the above steps were implemented to create Electrocardiogram classification and prediction using the MIT-BIH dataset. All the tools used were mentioned throughout the implementation stages. Two experiments – Local implementation and Apache Spark were implemented to answer the research questions and research objectives.

## REFERENCES

[1] https://pywavelets.readthedocs.io/en/latest/\
[2] https://github.com/MIT-LCP/wfdb-python
[3] https://keras.io/guides/writing_a_training_loop_from_scratch/