

Configuration Manual

MSc Research Project
Data Analytics

Sanket Sonu
Student ID: x19206071

School of Computing
National College of Ireland

Supervisor: Dr. Rejwanul Haque

National College of Ireland
Project Submission Sheet
School of Computing



Student Name:	Sanket Sonu
Student ID:	x19206071
Programme:	Data Analytics
Year:	2021
Module:	MSc Research Project
Supervisor:	Dr. Rejwanul Haque
Submission Due Date:	16/12/2021
Project Title:	Configuration Manual
Word Count:	675
Page Count:	5

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Sanket Sonu
Date:	15th December 2021

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	<input type="checkbox"/>
Attach a Moodle submission receipt of the online project submission , to each project (including multiple copies).	<input type="checkbox"/>
You must ensure that you retain a HARD COPY of the project , both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Sanket Sonu
x19206071

1 Introduction

The objective of this 'Configuration Manual' is to demonstrate all the steps required to perform the execution of this project 'Identifying Emotions for Code Mixed Hindi-English Tweets'. This project requires a list of Python libraries that need to be installed on local machine or cloud notebooks such as 'Google Colab' and 'Kaggle Notebook'.

2 System Configurations

2.1 Hardware Configuration

This project has been performed on the below hardware configuration. Figure 1 shows the Windows hardware configuration and Figure 2 show NVIDIA graphics card information.

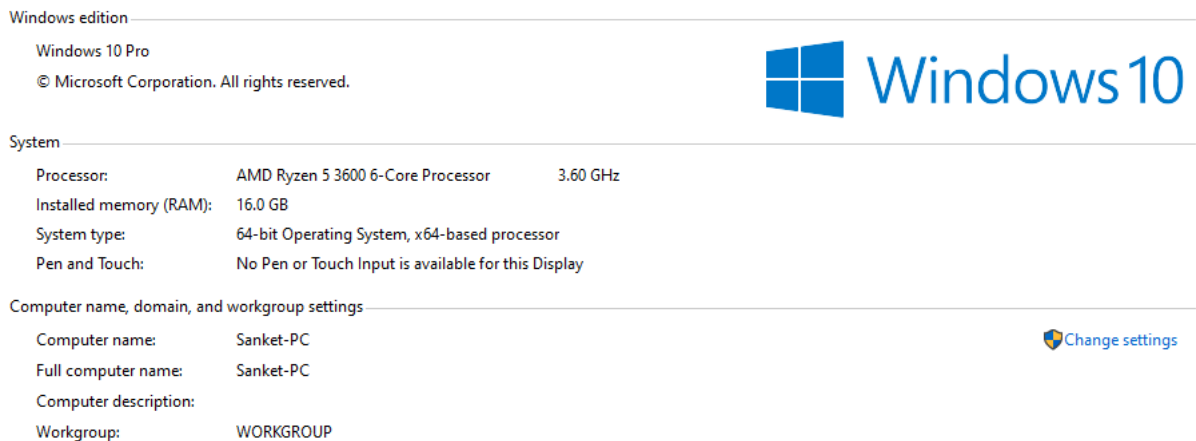


Figure 1: Hardware Configuration

2.2 Software Specification

This project has been performed using both 'Anaconda Jupyter Notebook' and 'Google Colab'.

1. **Anaconda:** Anaconda Individual Edition is free-to-use software, which has by default Python 3.8.8 version installed. Anaconda's Jupyter notebook has been used to execute the project.

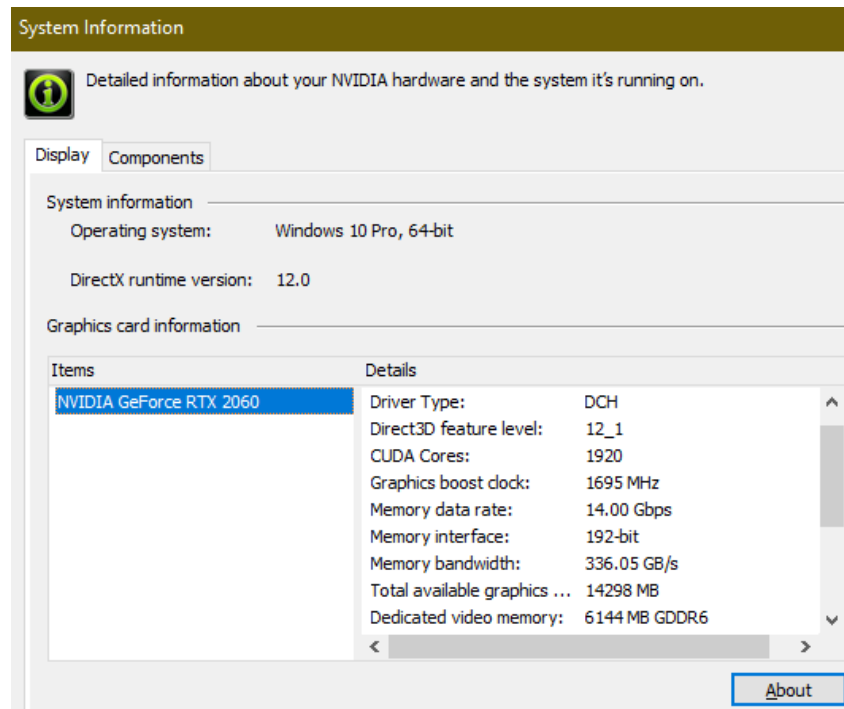


Figure 2: Graphics Card Information

2. **Google Colab:** Google Colab is a free notebook introduced by Google, which can be used to perform similar tasks just like Anaconda. Google Colab is pre-installed and ready with lots of libraries, which are not available on Jupyter notebook by default and have to be installed individually on Jupyter Notebook. Google Colab comes with free basic GPU and TPU run-time for faster training of the machine and deep learning models.

3 Installations

3.1 Anaconda Installation:

1. The Anaconda software can be downloaded from **Link**
2. Anaconda can be installed by following the basic instructions given by the software prompt while installing.
3. First, we need to install *Microsoft Visual Redistributable C++* from this **Link**
4. We have to install libraries that need to be installed to execute the project. This can be performed using *Anaconda Prompt* which can be found after pressing *Windows* button and type *CMD*. *Anaconda Prompt* need to be *Run as administrator*, which can be found by right-clicking on *Anaconda Prompt*.
5. We have to create a virtual environment where we have to install all libraries. This can be done using following command: `'conda create new_env python==3.8'`

6. Once the new environment *'new_env'* is created, it need to be accessed or activated in order to use, by using this command: *'conda activate new_env'*.
7. Now we have to install following libraries using PIP or CONDA command:
 - conda install pandas
 - conda install numpy
 - conda install matplotlib
 - conda install seaborn
 - conda install scikit-learn
 - conda install nltk
 - conda install gensim
 - conda install wordcloud
 - conda install tqsm
 - conda install textblob
 - conda install keras
 - conda install -c lightsource2-tag collection
 - In order to use GPU, we have to install cudatoolkit and cudnn, Version can be found on this **Link** CUDA = 11.2 and cuDNN = 8.1 These can be installed using this command: "conda install cudatoolkit=11.2 cudnn=8.1 -c=conda-forge"
 - once cudatoolkit and cudnn are installed we have to upgrade tensorflow to latest version tensorflow-gpu = 2.7.0 which can be done using: "pip install -upgrade tensorflow-gpu==2.7.0"
 - pip install tweepy - *'tweepy'* is used to extract Tweets from twitter database. Figure 3 shows the code.
 - pip install pydotplus
 - pip install graphviz

3.2 Google Colab environment setup

Google can be accessed by any web browser such as Chrome, Firefox, Microsoft edge, etc. To open google colab, visit this **Link**.

1. Click on *'File'* - select - *'New notebook'* Figure 4 New notebook.
2. A new notebook will open, now click on *'Runtime'* and select *'Change runtime'* and select *'GPU'* from the list of options available in *'notebook settings'*. Figure 5 shows the *'runtime'* options available.
3. Once the new notebook is ready with *'GPU runtime'*, we are good to go. Google colab is pre-installed with all the libraries that we had to install in the Anaconda environment.

```
[ ] !pip install tweepy
import tweepy
import pandas as pd

def tweepy_search(words):

    # Keys and Tokens for auth:
    consumer_key = "VWVzqIAdmKCyhs9io918UcdAB"
    consumer_secret = "utGK2Y3hrOnJCO341gX8K1MAUSHOXyMNF6nREIN17r0Aftdot"
    access_token = "I61382177-lm7qCg7ne1d78p25y3JE3wVskOvtV5QahPQ1P2gQ"
    access_token_secret = "QytdopdXBP4fYbJTMUK041E9nawFysIOkeqv80kvJp2zF"

    # authorization of consumer key and consumer secret
    auth = tweepy.OAuthHandler(consumer_key, consumer_secret)

    # set access to user's access key and access secret
    auth.set_access_token(access_token, access_token_secret)

    # calling the api
    api = tweepy.API(auth, wait_on_rate_limit=True)

    count = 2000
    global df
    df = pd.DataFrame(columns=['id', 'text']) # Creating DataFrame with columns
    tweets = tweepy.Cursor(api.search, q=words, lang="hi", tweet_mode="extended").items(count)
    list_tweets = [tweet for tweet in tweets]

    for tweet in list_tweets:
        id = tweet.id

        try:
            text = tweet.retweeted_status.full_text
        except AttributeError:
            text = tweet.full_text

        tweets = [id, text]

    df.loc[len(df)] = tweets
    df['id'] = df['id'].astype(str)
    filename = '5.xlsx'
    df.to_excel(filename) # Exporting DataFrame to a CSV file
    pd.set_option('display.max_colwidth', -1)

tweepy_search("tension ho rhi") # Calling the function
df
pd.set_option('display.max_rows', None)
```

Figure 3: Tweepy API - Code

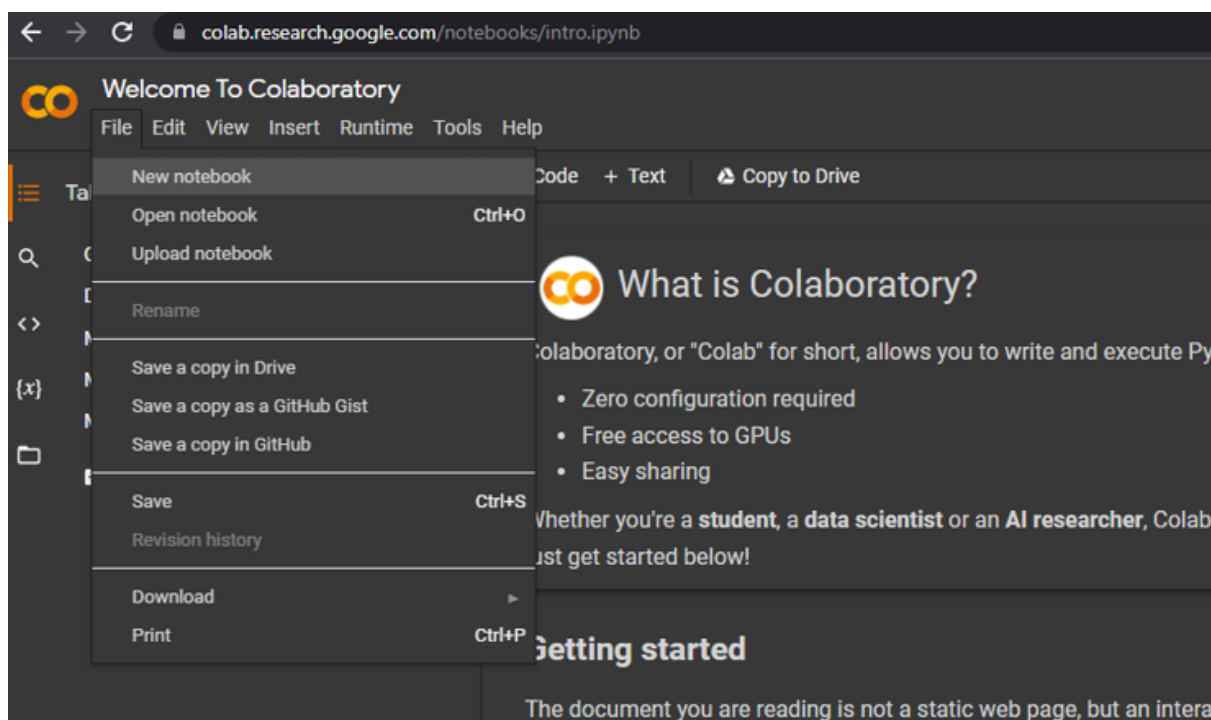


Figure 4: New notebook

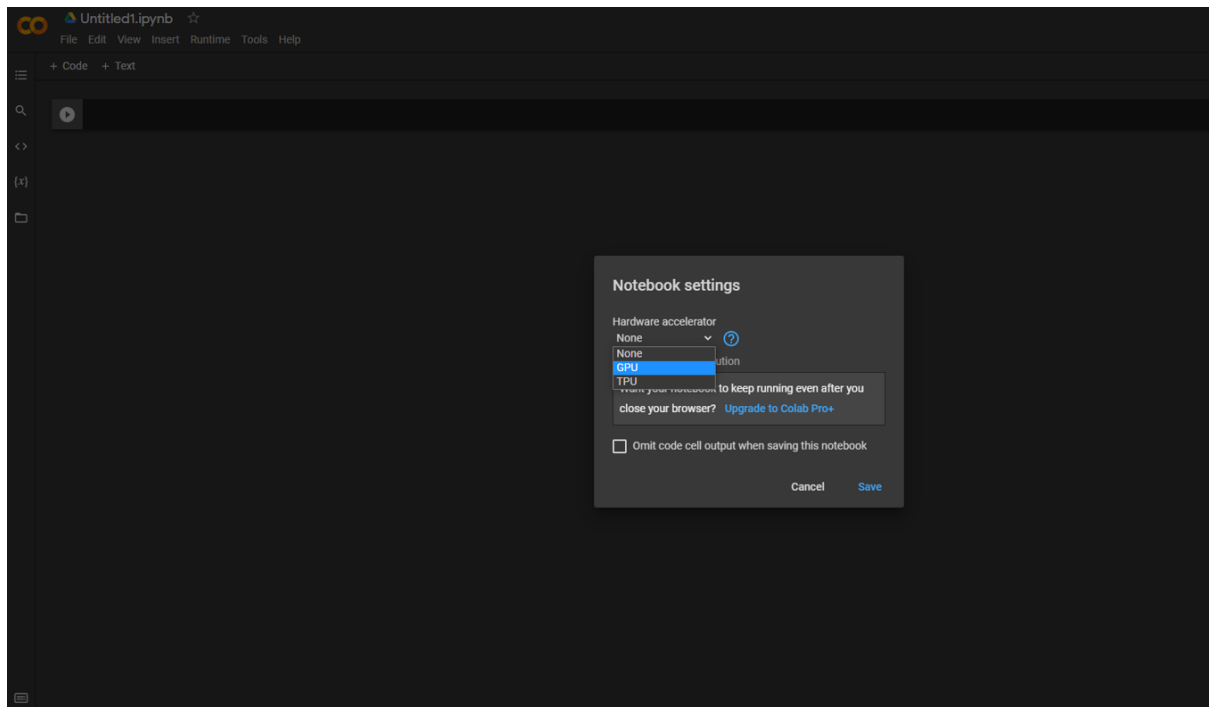


Figure 5: Select GPU as a runtime

4 Report Writing using Overleaf

After implementation of codes and result evaluation, report has been written using Overleaf: [Link](#). Figure 6 shows the overleaf that is used for report writing.

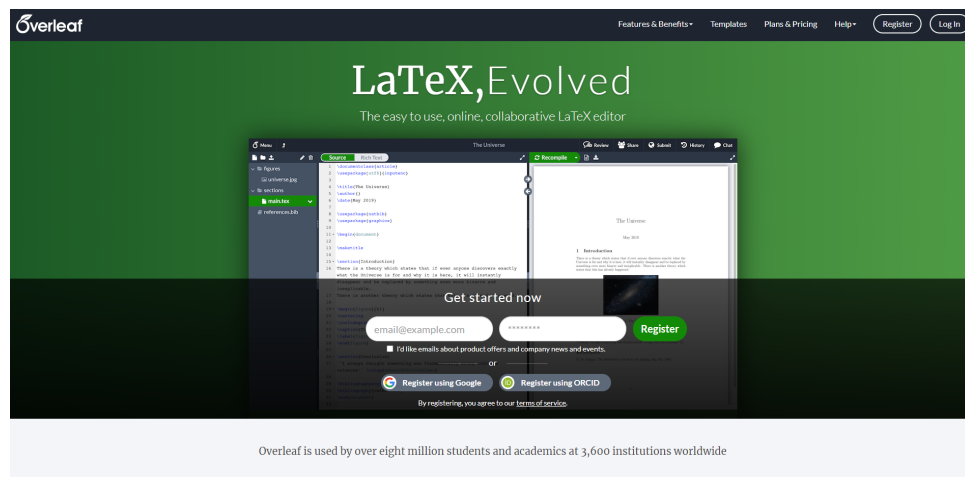


Figure 6: Overleaf