Detecting Fake vs Real News using Machine Learning

Software Engineering Project: EEC 626
Developers Guide

Anubhuti Dayal – 2824826

Oyindoubra Timi – 2822556

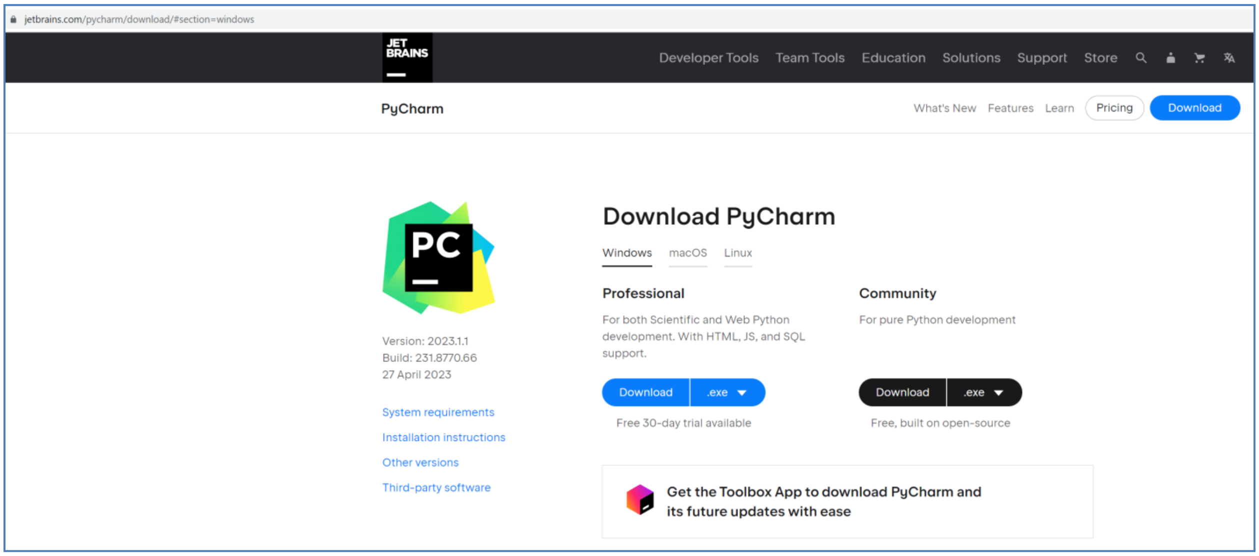
Developers Guide

Please find Execution Steps for the project “Detecting Fake vs Real News”

- ➢ Get the Fake news detection code folder Downloaded on your machine and unzip it

1. Download and install PyCharm or any python supported IDE. For this project, PyCharm and VS Code were used.

   You can follow this link - https://youtu.be/qC6-Uv9m_Ls
   https://www.jetbrains.com/pycharm/download/#section=windows



Download PyCharm

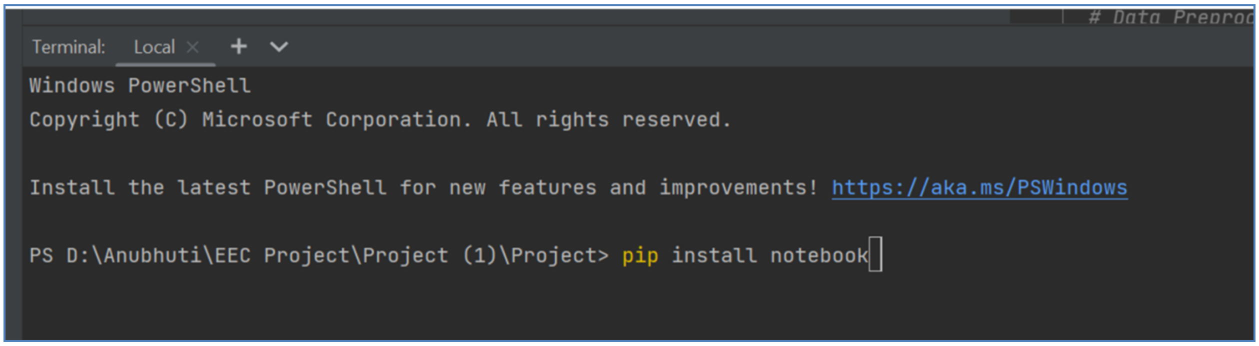
Version: 2023.1.1
Build: 231.8770.66
27 April 2023

Professional
For both Scientific and Web Python development. With HTML, JS, and SQL support.
Free 30-day trial available

Community
For pure Python development
Free, built on open-source

Get the Toolbox App to download PyCharm and its future updates with ease

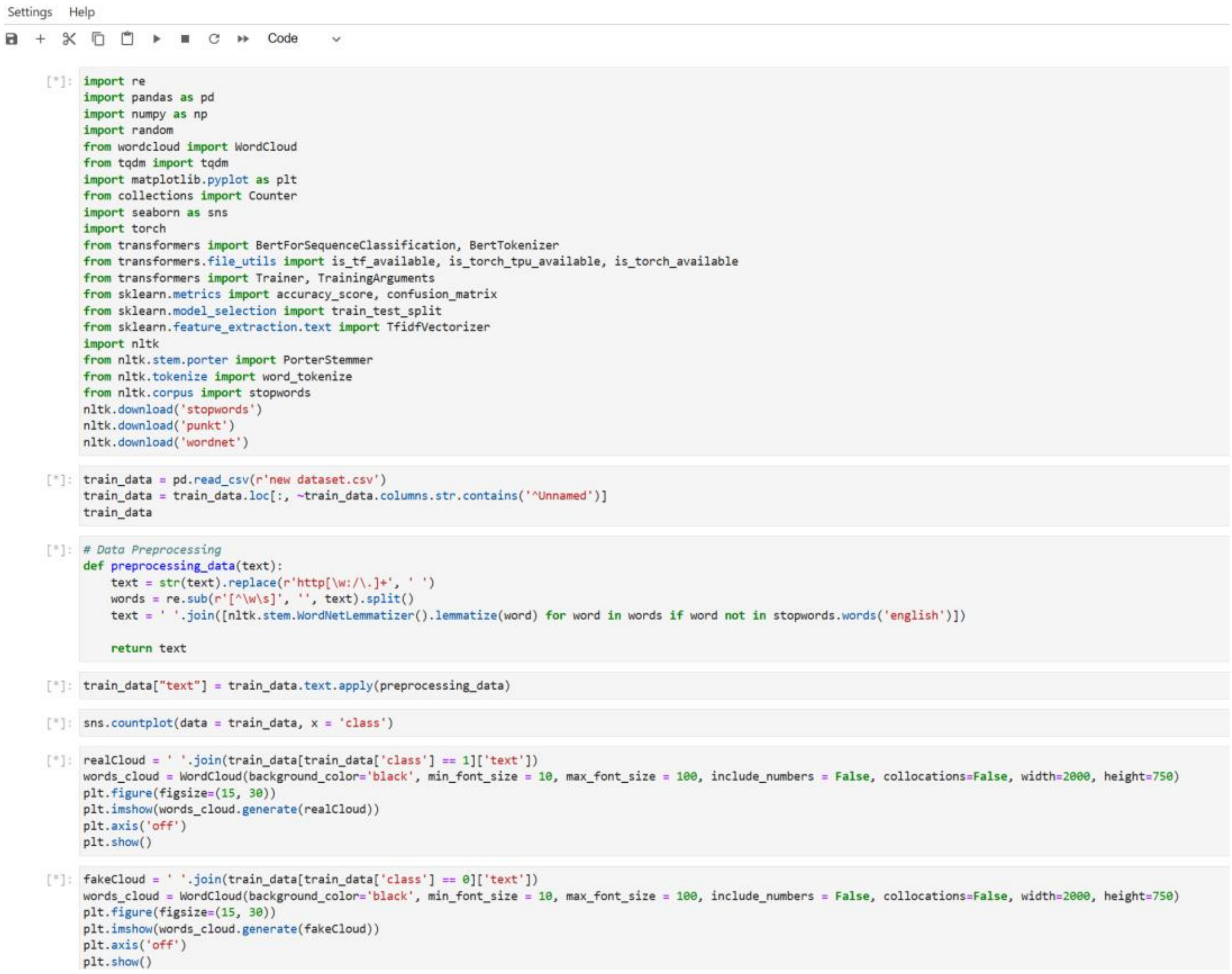
2. Open the project folder in the PyCharm environment on your machine after installation
3. Open a terminal on the IDE
4. If jupyterlab is not already installed in your system, type “pip install notebook” and hit “Enter” on your keyboard



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\Anubhuti\EEC Project\Project (1)\Project> pip install notebook
```

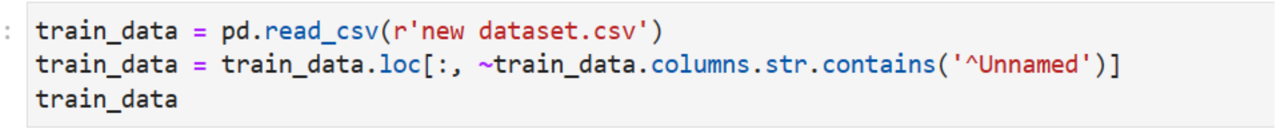
5. Next, type “jupyter notebook” and hit “Enter.” This should open a tab on your browser that shows all the documents and files in the project folder

6. Scroll to “FakeNews.ipynb" file and click on it. This should open in a new tab



```
[*]: import re
     import pandas as pd
     import numpy as np
     import random
     from wordcloud import WordCloud
     from tqdm import tqdm
     import matplotlib.pyplot as plt
     from collections import Counter
     import seaborn as sns
     import torch
     from transformers import BertForSequenceClassification, BertTokenizer
     from transformers.file_utils import is_tf_available, is_torch_tpu_available, is_torch_available
     from transformers import Trainer, TrainingArguments
     from sklearn.metrics import accuracy_score, confusion_matrix
     from sklearn.model_selection import train_test_split
     from sklearn.feature_extraction.text import TfidfVectorizer
     import nltk
     from nltk.stem.porter import PorterStemmer
     from nltk.tokenize import word_tokenize
     from nltk.corpus import stopwords
     nltk.download('stopwords')
     nltk.download('punkt')
     nltk.download('wordnet')

[*]: train_data = pd.read_csv(r'new dataset.csv')
     train_data = train_data.loc[:, ~train_data.columns.str.contains('^Unnamed')]
     train_data

[*]: # Data Preprocessing
     def preprocessing_data(text):
         text = str(text).replace(r'http[\w:/\.]+', ' ')
         words = re.sub(r'[^\w\s]', '', text).split()
         text = ' '.join([nltk.stem.WordNetLemmatizer().lemmatize(word) for word in words if word not in stopwords.words('english')])

         return text

[*]: train_data["text"] = train_data.text.apply(preprocessing_data)

[*]: sns.countplot(data = train_data, x = 'class')

[*]: realCloud = ' '.join(train_data[train_data['class'] == 1]['text'])
     words_cloud = WordCloud(background_color='black', min_font_size = 10, max_font_size = 100, include_numbers = False, collocations=False, width=2000, height=750)
     plt.figure(figsize=(15, 30))
     plt.imshow(words_cloud.generate(realCloud))
     plt.axis('off')
     plt.show()

[*]: fakeCloud = ' '.join(train_data[train_data['class'] == 0]['text'])
     words_cloud = WordCloud(background_color='black', min_font_size = 10, max_font_size = 100, include_numbers = False, collocations=False, width=2000, height=750)
     plt.figure(figsize=(15, 30))
     plt.imshow(words_cloud.generate(fakeCloud))
     plt.axis('off')
     plt.show()
```

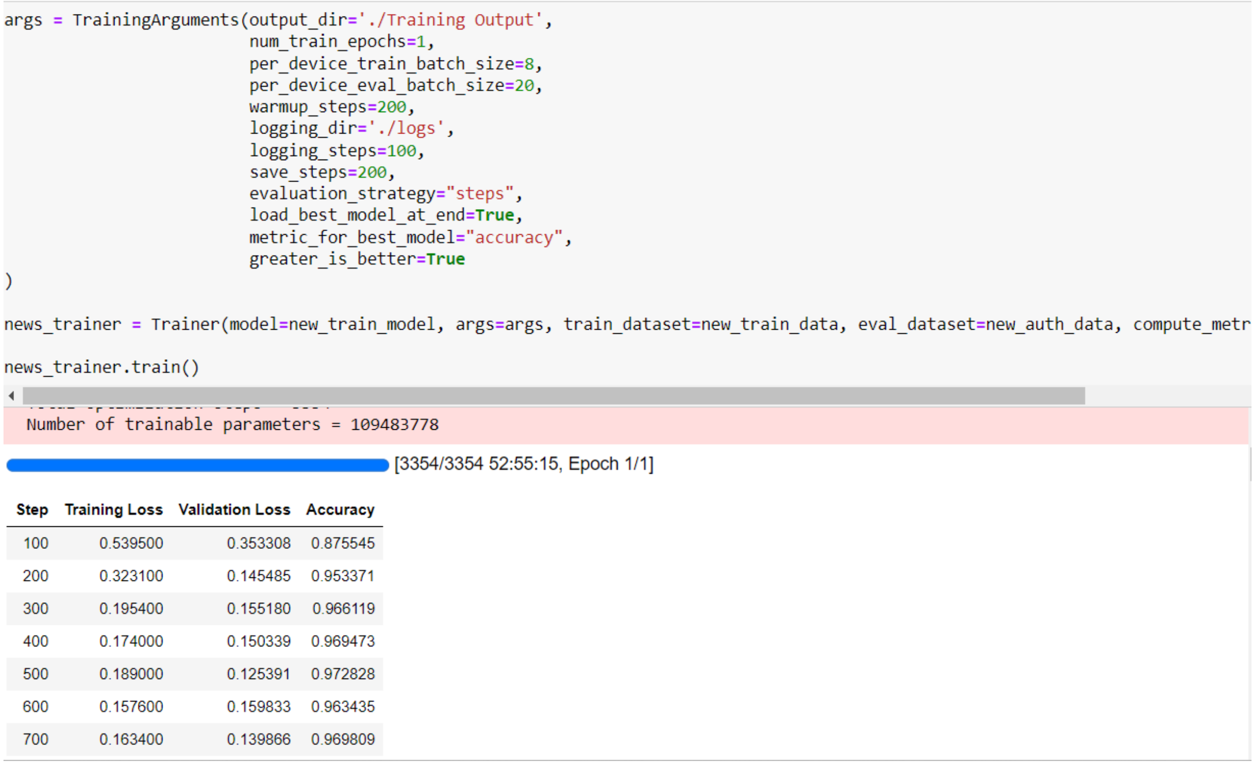
- ➢ Change/update data Path in the code



```
train_data = pd.read_csv(r'new dataset.csv')
train_data = train_data.loc[:, ~train_data.columns.str.contains('^Unnamed')]
train_data
```

7. Run each cell. Some cells might take hours or days to finish running depending on the speed of your device

8. After running the last cell, an input box will be displayed. Input your news in the box and hit "Enter"

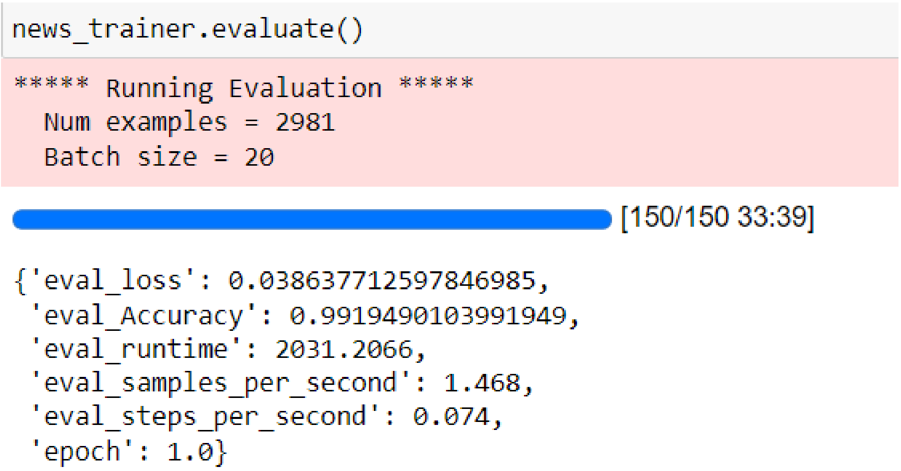


```
args = TrainingArguments(output_dir='./Training Output',
                         num_train_epochs=1,
                         per_device_train_batch_size=8,
                         per_device_eval_batch_size=20,
                         warmup_steps=200,
                         logging_dir='./logs',
                         logging_steps=100,
                         save_steps=200,
                         evaluation_strategy="steps",
                         load_best_model_at_end=True,
                         metric_for_best_model="accuracy",
                         greater_is_better=True
)

news_trainer = Trainer(model=new_train_model, args=args, train_dataset=new_train_data, eval_dataset=new_auth_data, compute_metr

news_trainer.train()
```

```
  Number of trainable parameters = 109483778
```

[3354/3354 52:55:15, Epoch 1/1]

Step	Training Loss	Validation Loss	Accuracy
100	0.539500	0.353308	0.875545
200	0.323100	0.145485	0.953371
300	0.195400	0.155180	0.966119
400	0.174000	0.150339	0.969473
500	0.189000	0.125391	0.972828
600	0.157600	0.159833	0.963435
700	0.163400	0.139866	0.969809



```
news_trainer.evaluate()
```

```
***** Running Evaluation *****
  Num examples = 2981
  Batch size = 20
```

[150/150 33:39]

```
{'eval_loss': 0.038637712597846985,
 'eval_Accuracy': 0.9919490103991949,
 'eval_runtime': 2031.2066,
 'eval_samples_per_second': 1.468,
 'eval_steps_per_second': 0.074,
 'epoch': 1.0}
```

9. The result will be displayed to tell you if the news is real or fake

```python
def get_prediction(text, convert_to_label=False):
    inputs = bert_token(text, padding=True, truncation=True, max_length=512, return_tensors="pt")
    outputs = new_train_model(**inputs)
    probs = outputs[0].softmax(1)
    d = {0: "Fake", 1: "Real"}
    if convert_to_label:
        return d[int(probs.argmax())]
    else:
        return int(probs.argmax())
```

```python
news = str(input())

get_prediction(news, convert_to_label=True)
```

```
Former Vice President Mike Pence testified on Thursday to a federal grand jury investigating the aftermath of the 2020 election
and the actions of then-President Donald Trump and others, sources familiar with the matter told CNN.  The testimony marks a mo
mentous juncture in the criminal investigation and the first time in modern history a vice president has been compelled to test
ify about the president he served beside.  Pence testified for more than five hours, a source familiar with the matter told CN
N, and while adviser Marc Short did not confirm the appearance on Thursday, he addressed the legal back-and-forth over the test
imony.  “I think that the vice president, you know, had his own case based on the Speech and Debate Clause. He was pleased that
for the first time a judge acknowledged that it applied to the vice president of the United States,” Short said in an interview
on NewsNation afterward. “But he was willing to comply with the law, and courts have ordered him to testify.”
```

```
'Real'
```