PROJECT REPORT

On

MBTI PERSONALITY PREDICTION FROM SOCIAL NETWORKS USING TEXT SEMANTICS

Submitted in partial fulfilment for the requirement of the award
for the degree of

MASTERS IN COMPUTER APPLICATION

Submitted To-
Internal Guide
Dr. Deepali Kamthania
Professor
VSIT, VIPS

Submitted By-
Bhaswati Kalita
42717704418
MCA-5B

Vivekananda Institute of Professional Studies
(Affiliated to Guru Gobind Singh Indraprastha University, Delhi)

CERTIFICATE

This is to certify that the project report entitled "MBTI Personality Prediction based on social networks using text semantics" submitted in partial fulfillment of the fifth semester of MCA to the VIPS, done by Ms. Bhaswati Kalita, Roll No. 42717704418 is an authentic work carried out under my guidance. The matter embodied in this project work has not been submitted earlier for the award of any degree or diploma to the best of my knowledge and belief.

Date: 02/01/21

Signature of the Guide:

Name of the Guide: Dr Deepali Kamthania

Designation: Professor, VSIT, VIPS

Address: Rohini, Delhi

ACKNOWLEDGEMENT

At the outset, I thank God almighty for making my endeavour a success. I express my sincere gratitude to Dr. Deepali Kamthania, for her constant support and valuable suggestions without which the successful completion of this project would not have been possible. I express my immense pleasure and thankfulness to all the teachers and staff of the Department of Vivekananda School of Information Technology for their cooperation and support.

Finally, I am thankful to all my classmates and all the people who were involved directly or indirectly with their cooperation and support for the successful completion of this work.

Bhaswati Kalita

(42717704418)

Abstract

MBTI personality types can be predicted through many ways. The most commonly used methodology has been questionnaires that are time consuming and needs the focus of participant. This project will explore the area of predicting personalities without questionnaires. People are increasingly using digital platforms like Facebook, twitter, etc. This gives us an opportunity to study if there's a way to predict their personality using these platforms. With the development of social networks, a broad variety of techniques have been developed to identify user personalities based on their social activities and language usage practices. In this project, we analysed the performance of Naïve Bayes Classifier Algorithm in predicting Kaggle user's personality, based on their user profile and comments. The user data is extracted and mapped on the Mayer's Brigg Personality Model. All sixteen co-ordinates of the MBTI Model are considered in this study. In this project, we want to study the correlation between the language of individuals used on their social medias and their respective personality traits. This will help us know that to what extent can we predict personality traits from various linguistic features.

Keywords

Human Personality, Natural Language Processing, Psychology Analysis, Social Networks, Machine Learning, Naïve Bayes Algorithm.

Contents

1. Introduction

In the field of psychology, personality is studied as it speaks volumes about how people behave in their life. As the world is advancing, people are using digital platforms like social medias to express themselves. Personality can be identified through their status and posts on social media like Facebook, Twitter etc. It has been shown to be relevant to many types of interactions. [3] Personality can be predicted using different models. One such model is the Myers-Briggs Type Indicator (MBTI) where personalities are divided into 16 different types. The MBTI divides the traits into four classes such as: Introversion (I) or Extroversion (E), Sensing (S) or Intuition (N), Thinking (T) or Feeling (F), and Judging (J) or Prospecting (P). Eg: INFP.

Problem Statement and Objectives

The MBTI tests use many multiple choice questions to determine the personality of an individual. But, this approach is time consuming and requires people to be focused enough to answer correctly. Thus, we think of minimizing the efforts of users and making it more efficient to predict a personality. We can thus model this as a classification problem. A successful implementation of such a classifier would demonstrate a good connection between linguistic features and potential personality in general [2]. This won't just help users know their personality but can also be used in psychoanalysis to help find the relation between natural language and personality type.

- Predicting MBTI personality type using texts from social medias.
- Study the relation between natural language and MBTI personality.

1.2 Motivation

If we happen to find a correlation between natural language and MBTI personalities, it can be a contribution towards psychoanalysis [5]. People don't always realise how do they think like and a lot of what they post on social media may have a significant relation with their true selves. This project will also help people know about themselves without solving a lot of questions that many a people find tiresome and thus don't participate into finding their own selves and if they participate then a lot of times they don't answer correctly. Employers can find their employees using public information provided by employees if we achieve sufficient accuracy in this project.

1.3 Scope and Limitation

As we mentioned above, the project has scope in psychoanalysis however it has certain limitations.

• The project is based on texts. It does not include images, URLs etc. People don't always express their original thoughts and writing texts isn't as interesting for everyone. They prefer sharing images, blogs, etc. written by another person than writing something about the topic on their own. This makes it difficult for our code to find sufficient data. If we include other factors like the images they share, the types of blogs they share, the content of articles they shared, the connects they have or prefer, their likes or dislikes. This may add value to our accuracy and provide sufficient data.

2. Related Work

Mihai Gavrilescu [7] and Champa H N [8] used deep feed forwards neural networks for small datasets that are textual. This was proven to be successful in predicting personality. They used a 3 layered feed forward architecture on textual data which is handwritten. Even though, this model that they used had handwritten features than just text, they proved it that MBTI personalities can be predicted using deep neural architectures.

3. Methodology

3.1 Dataset Description

For this project, we used the Myers-Briggs Personality Type Dataset available on Kaggle [1]. This data which is available on Kaggle was collected through the PersonalityCafe forum that provides a large number of people, their respective MBTI personality types and what they have posted.

Details	Count
Number of instances(posts)	8,675
Number of unique attributes	16

Table 1: Details of the dataset.

Every dataset also comprises of data attributes. Table 2 describes attributes of data. In case of supervised learning, clearly mention which attribute(s) would be considered as the labels.

Data Attributes	Brief Explanation
Personality type	Posts written by that personality type

Table 2: Details of Data Attributes.

3.2 **Description of attributes**

- I - Introversion: This indicates how an individual will interact and respond the world around him. Introverts tend to spend more time alone.
- E - Extroversion: Extroverts are termed to be action oriented and tend to have more frequent social interactions.
- S - Sensing: Individual who prefers sensing tends to rely on and oriented towards the things that are real, and is more interested in facts and details.
- N - Intuitive: An individual who prefers intuition rely more on imagining things like possibilities, abstract, theories, etc.
- T - Thinking: This scale is all about decision making capability of an individual. Thinking will arrive at conclusion using logic, facts and figures.
- F - Feeling: Those who prefer feeling may consider people's emotions before coming to conclusion.
- P - Prospecting: The ones who tend towards perception are more flexible and adaptable.
- J - Judging: Individuals who tend towards judging are more likely to take firm decisions.
- A combination of I or E, S or N, F or T and P or J gives us an MBTI personality. [6]

3.3 **Proposed Methodology**

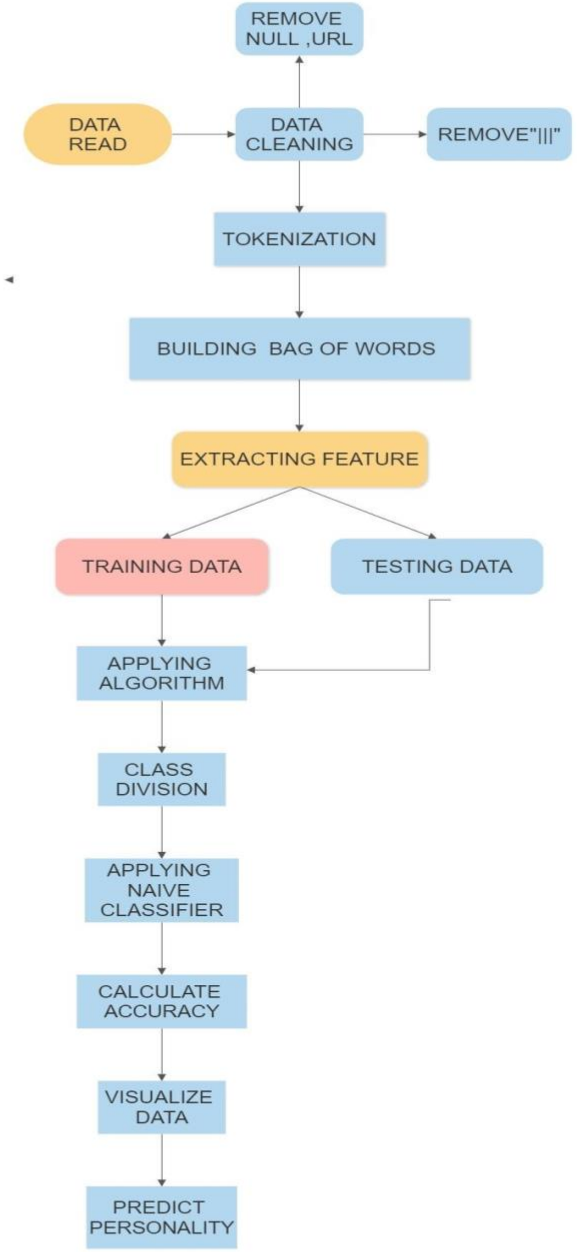


Figure 1: Flowchart

- Domain: Predictive analytics

– Task of classification and prediction is the key

– Predicting/filling the information that is unknown

– In our project, we are predicting the personality trait of a person

- Machine learning task: Supervised learning

– Given data and associated target response, model is trained and then is used to predict the correct response for a new data.

In this project,

∗ Data: post of user (post column in dataset)

∗ Target response: personality type of user (type column in dataset)

∗ New data: New user whose personality we want to predict

- Type of problem under supervised learning: Classification (multi-class classification)

– Class: 16 personality type, a user is assigned one out of these class

3.4 <u>Data Pre-processing</u>

Data Pre-processing is done for all respondent's tweets [4]. Since this project is based on textual data, some tweets may be irrelevant and needs to be pre-processed with text classification in order to transform the raw data in a useful and efficient format. Data pre-processing is done for each tweets in the following steps:

3.4.1 <u>Data Proportion</u>

As shown in Figure 2 given below, the data is clearly not in proportion. The number of posts are higher for INFP than any other type.

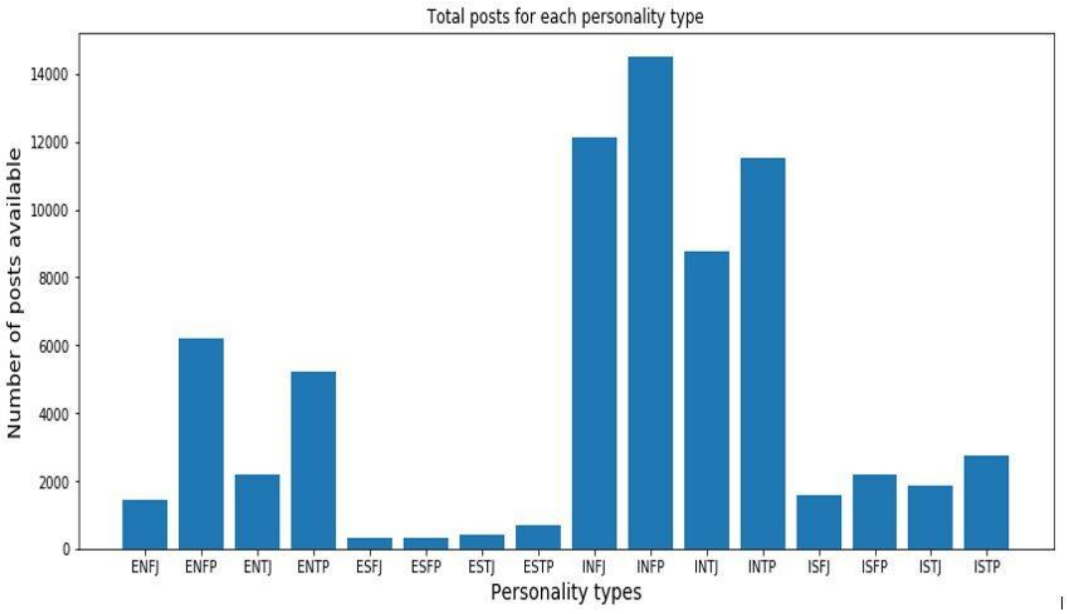


Figure 2: Graphical Representation of available data.

3.4.2 Data Cleaning

Since this project is strictly based on text, removal of URLs was necessary. We also removed all the NULL values. Next step was to remove common fillers like “or”, “a”, “the”, etc. This we did using python’s NLTK. In order to preserve the data, we replace the null values with the hyphen symbol.

3.4.3 Lemmatization

We used imported WordNetLemmatizer from nltk.stem to lemmatize the text which means that infected forms of the same word are treated as one form of the root word (e.g. “running”, “ran”, “run” all become “run”)[3].

3.4.4 Tokenization

Tokenization is necessary. Here, we split the available text into words using python’s Natural Language ToolKit (NLTK). We tokenized further find the useless words. To apply this, we needed bag of words. We have defined a set of useless words with nltk.stopwords to tokenize correct posts.

3.4.5 Bag of words

We built bag of words by removing all the stopwords and punctuation marks in order to have only necessary data on which we can apply our machine learning algorithm. The bag of words created are Dear, ENTJ, sub, long, time, see, sincerely etc which is then scored to mark the presence of words as Boolean value.

3.5 Splitting

Since each number of personality type has different number of posts, they must split accordingly. We have split the data into 2 parts. 80 percent is for training and 20 percent is for testing.

3.5.1 Algorithm [6]

```
split ← []
for i in range 16 do
    split += [len(features[i])*0.6]
    split ← np.array(split, dtype = int)end for
```

```
train ← []
for i in range 16 do
    train +=features[i][:split[i]
end for
sentimentclassif ier = N aiveBayesClassif ier.train(train)
nltk.classify.util.accuracy(sentimentclassif ier, train) ∗ 100
test ← []
for i in range 16 do
    test +=feautures[i][split[i]:]
end for
nltk.classify.util.accuracy(sentimentclassif ier, test) ∗ 100
```

The algorithm that we use here is **Naive Bayes Classifier Algorithm**. This is a classification technique which is based on Bayes Theorem with an assumption of independence among the given data values/set [5]. In our project, text given in several posts on social media platforms are the data values which forms our data set. This works well on the text/categorical data instead of numeric data. A classifier under the supervised learning based on probabilistic logic (bayes theorem). In lay-man language, we can say that an existence of the number of posts of a particular person/ individual is unrelated / independent from the existence of number of posts of a another person /individual and that's the assumption in naives bayes classifier. For each attribute from each class set, it uses probability to make predictions.

$$\{X1, X2, ..., Xn\} \text{———}\geq \{C1, ..., Ck\} \qquad (1)$$

In our project,

X1 denotes the no of posts of a particular person/ individual.

X2 denotes the no of posts of another person/ individual.

Xn denotes the no of posts of nth person/individual.

C1is the probability of the number of posts describing a particular trait personality trait.

Ck is the probability of the number of posts describing a kth – personality trait. The data model which is yielded is called as Predictive model with probabilistic problems at foundation.

4. Experiment Setup and Results

We are splitting our data set into training and test data. We are calculating accuracy in 4 trials (50:50, 60:40, 70:30, 80:20). This splitting is done on complete dataset where we have 16 classes each class representing a personality type. The accuracy through this method turned out to be 10% approximately as shown in Figure 3.

Hence, instead of selecting all 16 personalities as a unique feature, we decided to simplify the dataset. The MBTI personality type divides everyone into 16 personality types across 4 axis.

1. Introversion(I) or Extroversion(E)
2. Intuition(N) or Sensing(S)
3. Thinking (T) or Feeling (F)
4. Prospecting (P) or Judging (J)

Bayes theorem gives us a way to calculate posterior probability by the given equation: -

Posterior probability= Likelihood* Class Prior Probability /Predictor Prior Probability

Input=50 posts

Trained Data	**58.354826823876195**
Test Data	**10.4463235294117645**

Table 3: Result of the model based on 16 traits of personality

Output: - The language used in the number of posts (twitter posts) reflecting/describing each 16 types/ traits of personality and predicting the personality trait people possess.

Now we have 4 classes, we create 4 classifiers (Naive Bayes Classifier to classify the person into a particular personality) as shown in figure 3.

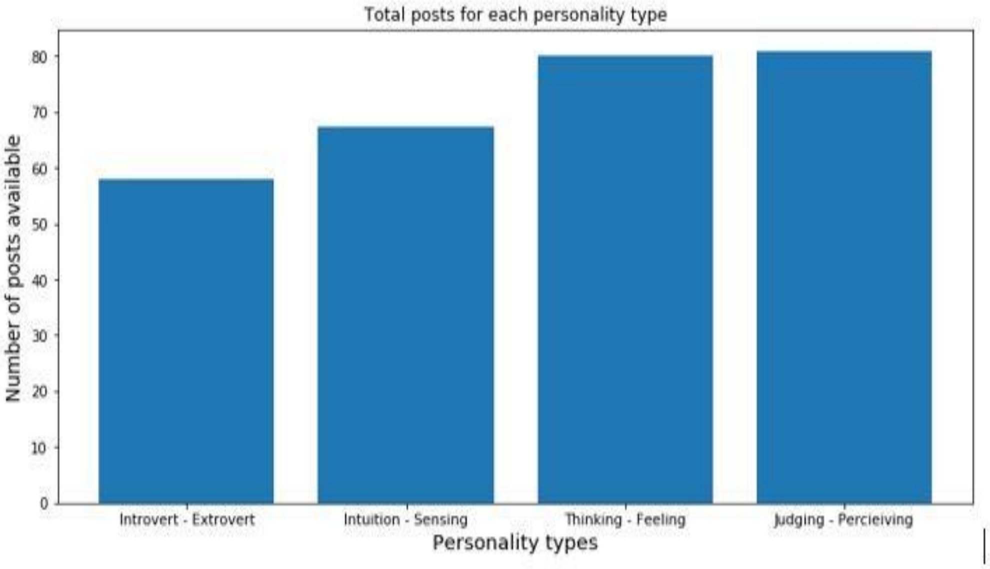


Figure 3: After classification into 4 classes

To increase the accuracy of our model, we took 4 classifiers of personality traits to classify the individual's personality (using MBTI)

Input=50 posts

Data	**Introvert-Extrovert**	**Intuition-Sensing**	**Thinking-Feeling**	**Prospecting-Judging**
Trained	**57.401437**	**67.658438**	79.504422	73.613670
Test	**49.705882**	**73.566176**	53.198529	48.768382

Table 4: Summarizing the results of the 4 classifiers

Output: -

The language used in the number of posts (twitter posts) reflecting/describing each (4) classifiers of personality and predicting the MBTI Trait using the classifiers.

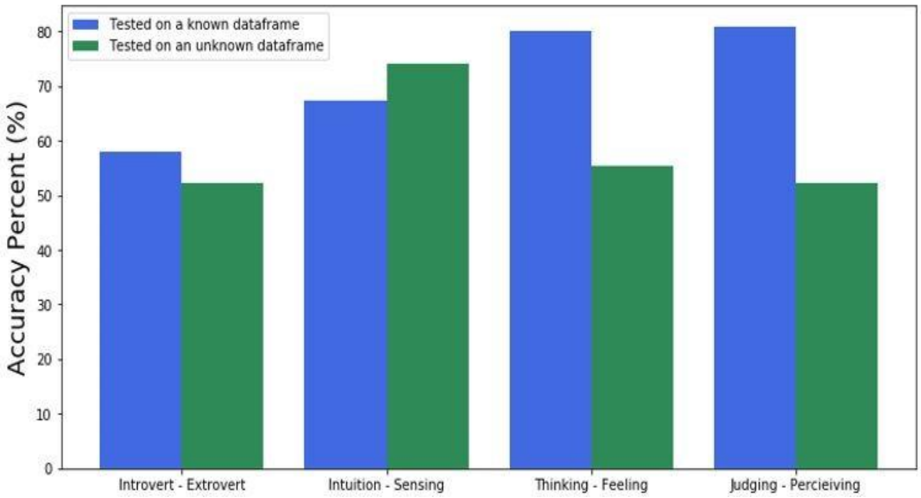


Figure 4: Model Classifying trait

We got approximately 53% accuracy after classifying the personality types into 4 classes rather than 16 types. In Figure 5, the graph shows which trait has higher percentage than the other and thus chooses the higher trait to predict the personality type.

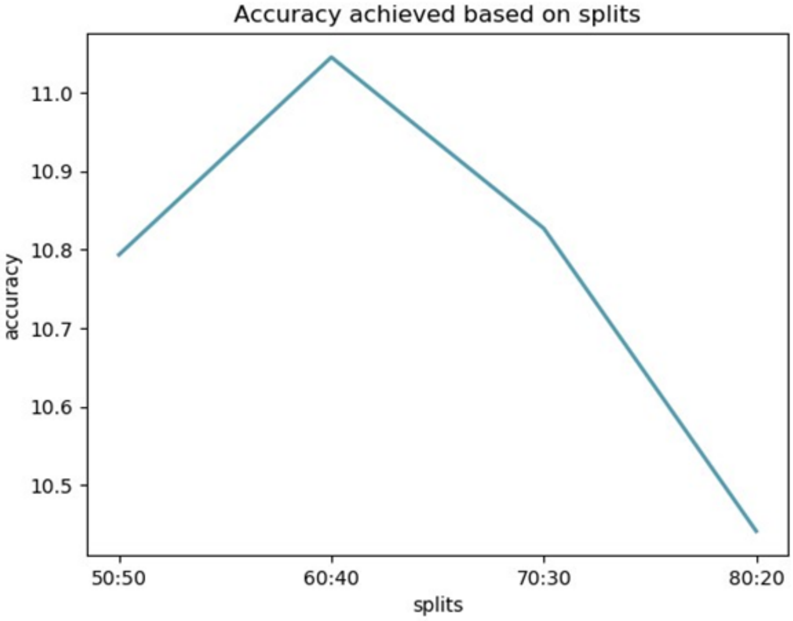


Figure 5: Splits vs Accuracy

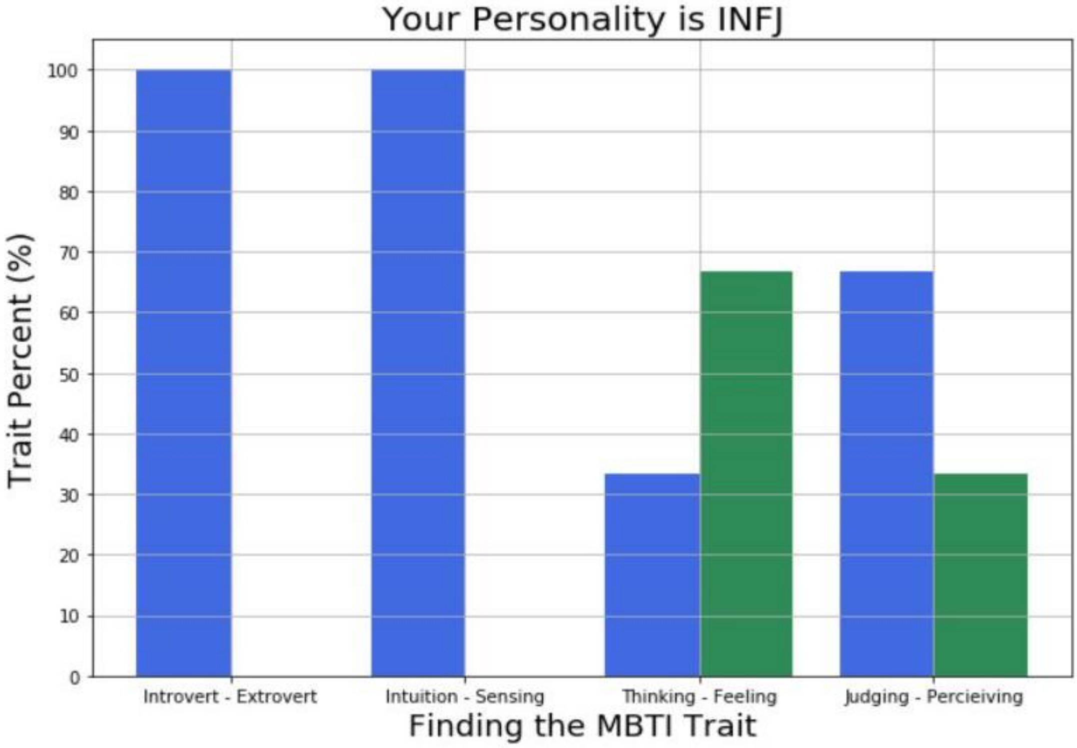


Figure 6: Results from Barack Obama's tweets

In Figure 6, we tried predicting the personality of Barack Obama based on his tweets and we got INFJ which is different from his original personality which is ENFJ.

5. Conclusion and Future Work

There is a slight difference between the personality predicted by the model and the personality predicted by 16 personalities. This might be because:

1. We have not scraped the profile but have copied few posts of the user into the test file.
2. We are using Naive Bayes classifier, the accuracy of which is 53%, so according to the accuracy of the model, we are getting a good result.
3. We didn't proportionalise the data and thus it's more likely that our code predicts INFP or traits related to INFP as it has the highest number of posts. Our data is very imbalanced.

- For future work, we want to include more personality traits so that we can provide a more detailed personality to the user as well as to predict personality using textual data and sentiment analysis [9].
- There can be module where user will be provided with career guidance and counselling sessions which matches his personality.

6. **References**

[1] (MBTI) Myers-Briggs Personality Type Dataset | Kaggle

[2] Shristi Chaudhary, Ritu Singh, Syed Tausif Hasan, Ms. Inderpreet Kaur. A Comparative Study of Different Classifiers for Myers-Brigg Personality Prediction Model. *International Research Journal of Engineering and Technology (IRJET)*, May, 2018.

[3] Mamta Bhamare, K. Ashok Kumar. Personality Prediction from Social Networks text using Machine Learning. *International Journal of Recent Technology and Engineering (IJRTE)*, November 2019.

[4] Louis Christy Lukito, Alva Erwin, James Purnama, and Wulan Danoekoesoemo. Social Media User Personality Classification using Computational Linguistic. *8th International Conference on Information Technology and Electrical Engineering (ICITEE), Yogyakarta, Indonesia*, 2016.

[5] Francois Mairesse, Marilyn A. Walker, Matthias R. Mehl and Roger K. Moore. Using Linguistic Cues for the Automatic Recognition of Personality in Conversation and Text. *Journal of Artificial Intelligence Research*,11, 2007.

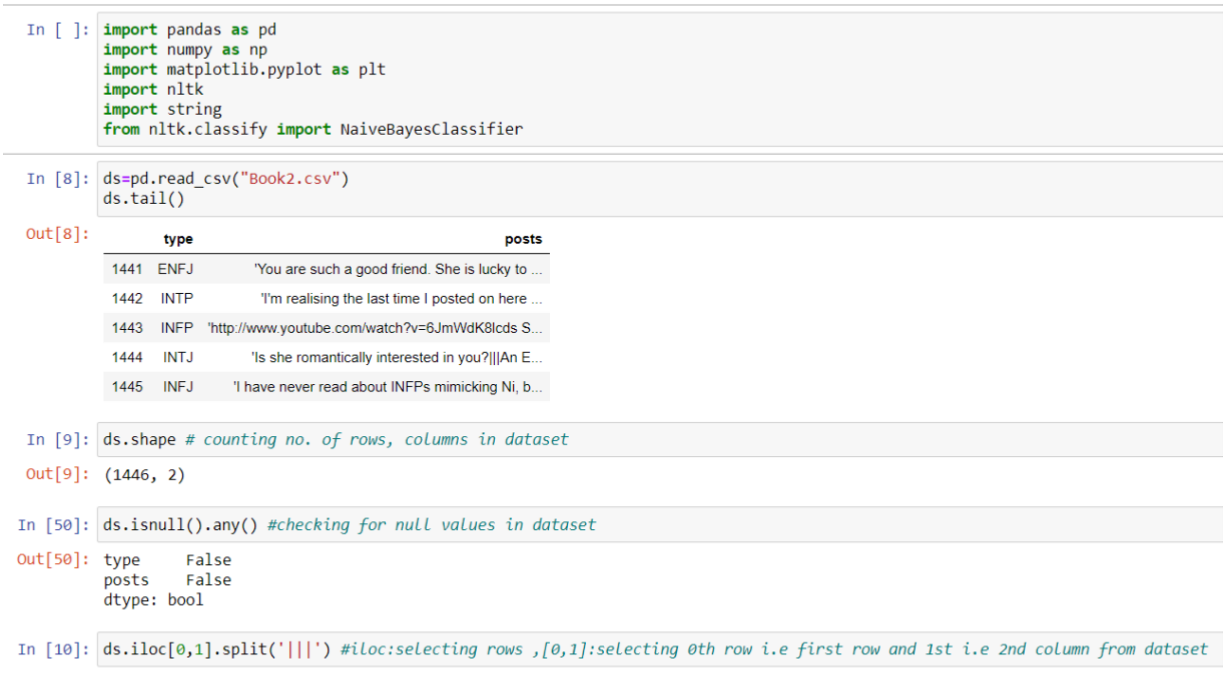
[6] Prajwal S, Shahid Afridi, Patel Sana Riyaj, Srihari Hegde G.K. and Aditya C.R. Traits and Learning Models for Personality Prediction Using Social Media. *International Journal of Scientific & Technology Research Volume 9*, 04 April 2020

[7] Mihai Gavrilesku. Study on determining the myers briggs personality type based on individual's handwriting. *The fifth IEEE International Conference on E-Health and bioengineering*, 11,2015.

[8] Champa H N and Dr. K R Anandakumar. Artificial neural network for human behaviour prediction through handwriting analysis. *International Journal of Computer Applications*, 05, 2010.

[9] Bhawna Singh, Swasti Singhal. Automated Personality Classification Using Data Mining Techniques. Galgotia's College of Engineering & Technology, Greater Noida-201301, U.P, India.

6. Appendix



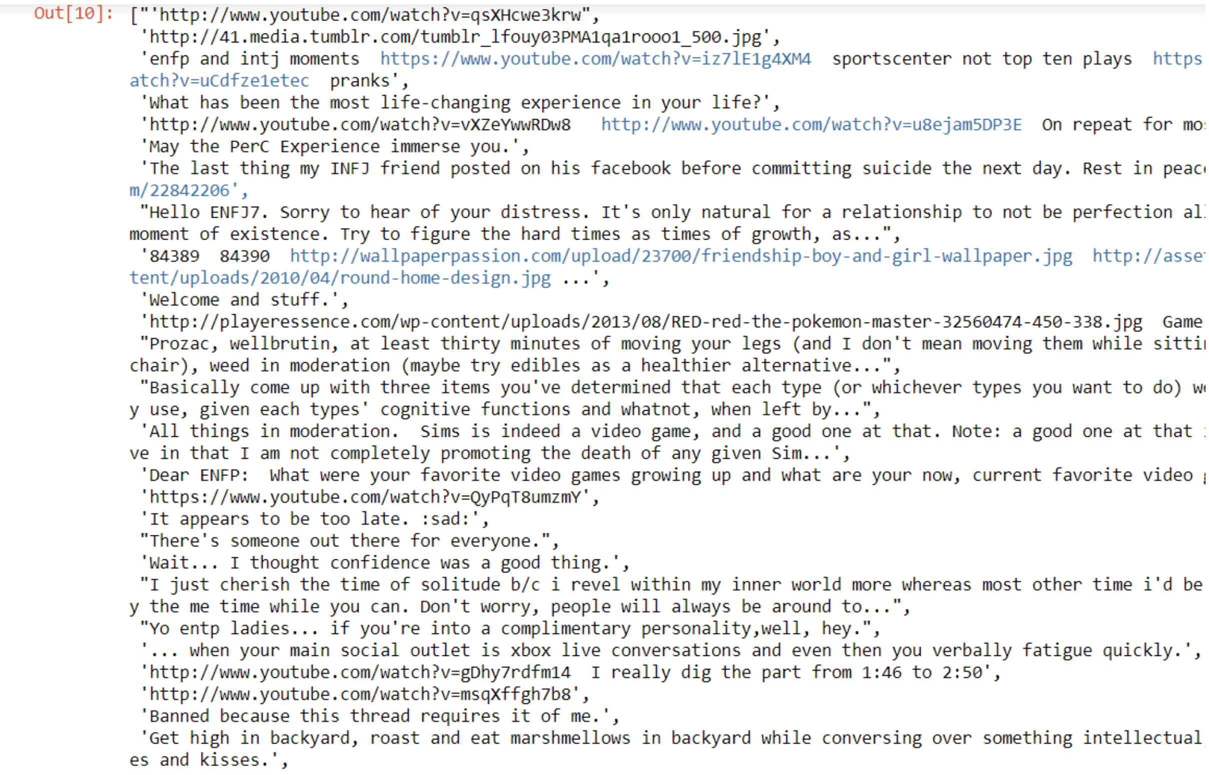
```
In [ ]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        import nltk
        import string
        from nltk.classify import NaiveBayesClassifier
```

```
In [8]: ds=pd.read_csv("Book2.csv")
        ds.tail()
```

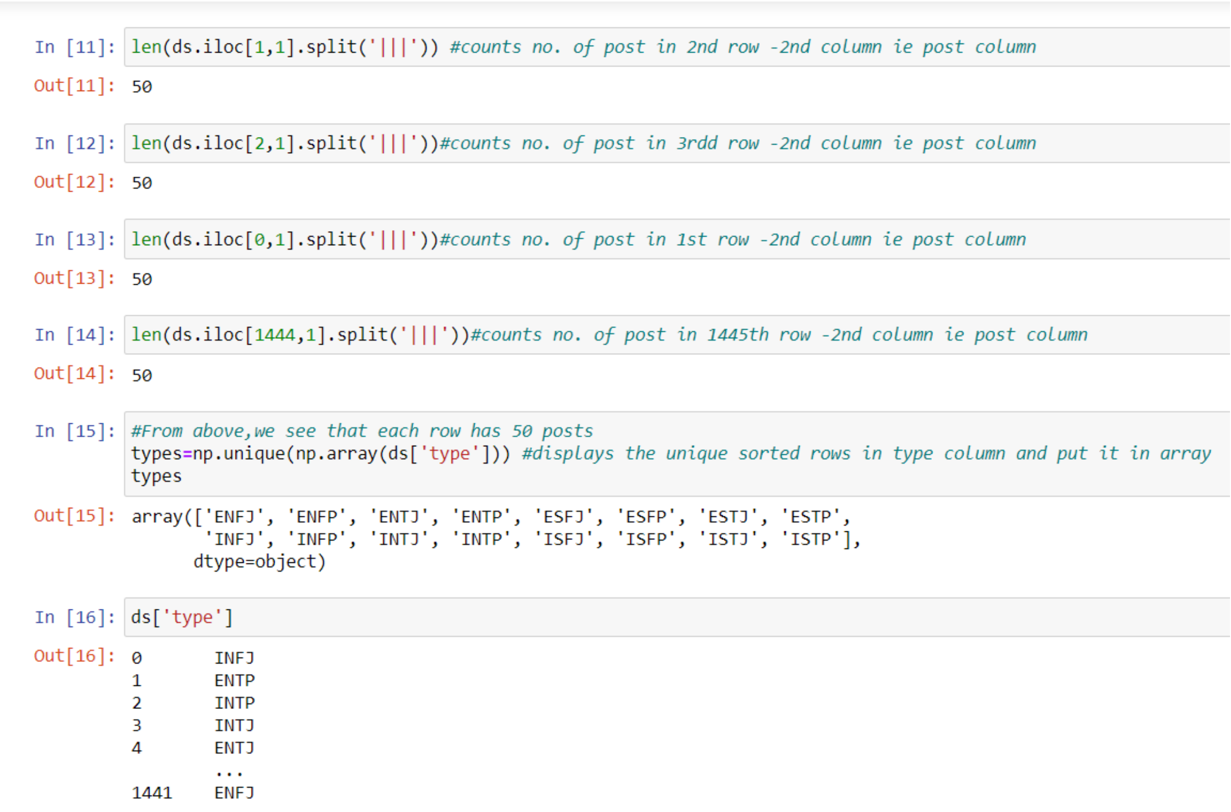
Out[8]:

	type	posts
1441	ENFJ	'You are such a good friend. She is lucky to ...
1442	INTP	'I'm realising the last time I posted on here ...
1443	INFP	'http://www.youtube.com/watch?v=6JmWdK8lcds S...
1444	INTJ	'Is she romantically interested in you?|||An E...
1445	INFJ	'I have never read about INFPs mimicking Ni, b...

```
In [9]: ds.shape # counting no. of rows, columns in dataset
Out[9]: (1446, 2)

In [50]: ds.isnull().any() #checking for null values in dataset
Out[50]: type     False
         posts    False
         dtype: bool

In [10]: ds.iloc[0,1].split('|||') #iloc:selecting rows ,[0,1]:selecting 0th row i.e first row and 1st i.e 2nd column from dataset
```



```
Out[10]: ["'http://www.youtube.com/watch?v=qsXHcwe3krw",
          'http://41.media.tumblr.com/tumblr_lfouy03PMA1qa1rooo1_500.jpg',
          'enfp and intj moments  https://www.youtube.com/watch?v=iz7lE1g4XM4  sportscenter not top ten plays  https
         atch?v=uCdfze1etec  pranks',
          'What has been the most life-changing experience in your life?',
          'http://www.youtube.com/watch?v=vXZeYwwRDw8   http://www.youtube.com/watch?v=u8ejam5DP3E  On repeat for mo
          'May the PerC Experience immerse you.',
          'The last thing my INFJ friend posted on his facebook before committing suicide the next day. Rest in peac
         m/22842206',
          "Hello ENFJ7. Sorry to hear of your distress. It's only natural for a relationship to not be perfection al
         moment of existence. Try to figure the hard times as times of growth, as...",
          '84389  84390  http://wallpaperpassion.com/upload/23700/friendship-boy-and-girl-wallpaper.jpg  http://asse
         tent/uploads/2010/04/round-home-design.jpg ...',
          'Welcome and stuff.',
          'http://playeressence.com/wp-content/uploads/2013/08/RED-red-the-pokemon-master-32560474-450-338.jpg  Game
          "Prozac, wellbrutin, at least thirty minutes of moving your legs (and I don't mean moving them while sitti
         chair), weed in moderation (maybe try edibles as a healthier alternative...",
          "Basically come up with three items you've determined that each type (or whichever types you want to do) w
         y use, given each types' cognitive functions and whatnot, when left by...",
          'All things in moderation.  Sims is indeed a video game, and a good one at that. Note: a good one at that
         ve in that I am not completely promoting the death of any given Sim...',
          'Dear ENFP:  What were your favorite video games growing up and what are your now, current favorite video
          'https://www.youtube.com/watch?v=QyPqT8umzmY',
          'It appears to be too late. :sad:',
          "There's someone out there for everyone.",
          'Wait... I thought confidence was a good thing.',
          "I just cherish the time of solitude b/c i revel within my inner world more whereas most other time i'd be
         y the me time while you can. Don't worry, people will always be around to...",
          "Yo entp ladies... if you're into a complimentary personality,well, hey.",
          '... when your main social outlet is xbox live conversations and even then you verbally fatigue quickly.',
          'http://www.youtube.com/watch?v=gDhy7rdfm14  I really dig the part from 1:46 to 2:50',
          'http://www.youtube.com/watch?v=msqXffgh7b8',
          'Banned because this thread requires it of me.',
          'Get high in backyard, roast and eat marshmellows in backyard while conversing over something intellectual
         es and kisses.',
```



In [11]:
```
len(ds.iloc[1,1].split('|||')) #counts no. of post in 2nd row -2nd column ie post column
```
Out[11]:
```
50
```

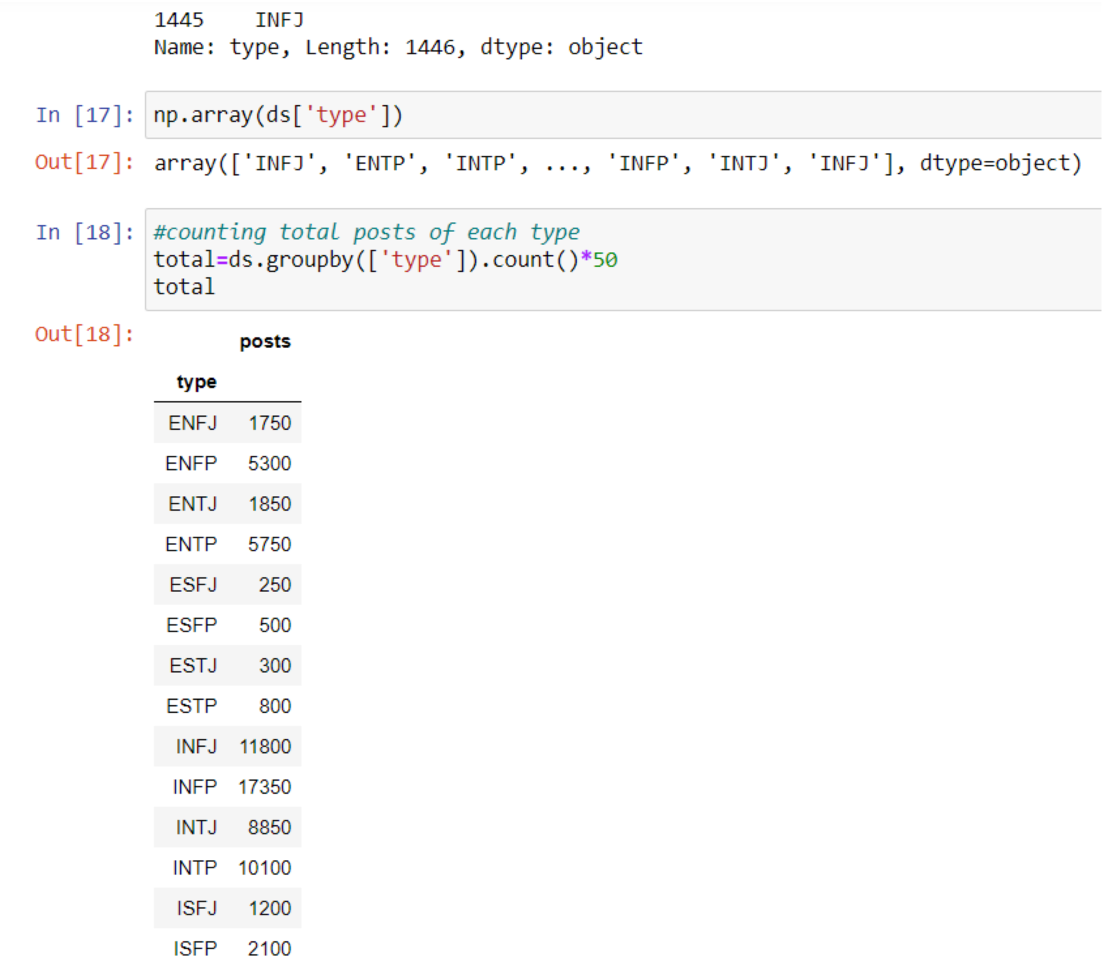
In [12]:
```
len(ds.iloc[2,1].split('|||'))#counts no. of post in 3rdd row -2nd column ie post column
```
Out[12]:
```
50
```

In [13]:
```
len(ds.iloc[0,1].split('|||'))#counts no. of post in 1st row -2nd column ie post column
```
Out[13]:
```
50
```

In [14]:
```
len(ds.iloc[1444,1].split('|||'))#counts no. of post in 1445th row -2nd column ie post column
```
Out[14]:
```
50
```

In [15]:
```
#From above,we see that each row has 50 posts
types=np.unique(np.array(ds['type'])) #displays the unique sorted rows in type column and put it in array
types
```
Out[15]:
```
array(['ENFJ', 'ENFP', 'ENTJ', 'ENTP', 'ESFJ', 'ESFP', 'ESTJ', 'ESTP',
       'INFJ', 'INFP', 'INTJ', 'INTP', 'ISFJ', 'ISFP', 'ISTJ', 'ISTP'],
      dtype=object)
```

In [16]:
```
ds['type']
```
Out[16]:
```
0       INFJ
1       ENTP
2       INTP
3       INTJ
4       ENTJ
        ...
1441    ENFJ
```

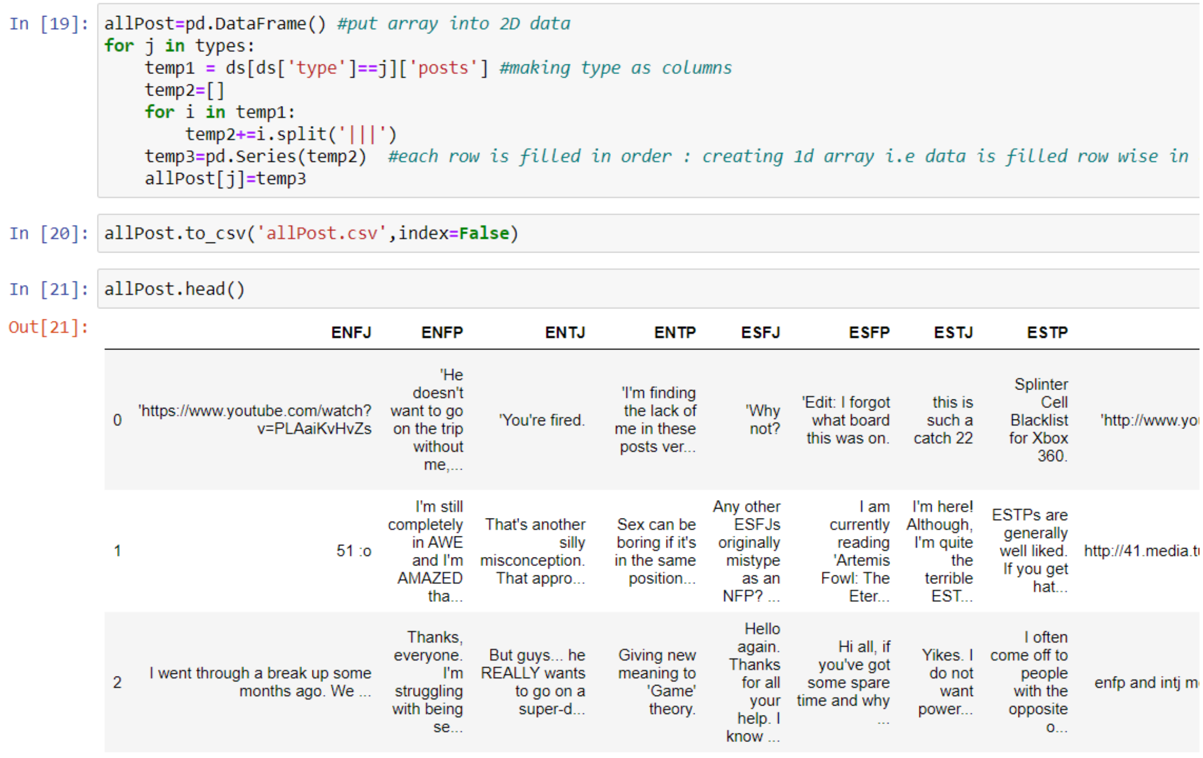


```
1445    INFJ
Name: type, Length: 1446, dtype: object
```

In [17]:
```
np.array(ds['type'])
```
Out[17]:
```
array(['INFJ', 'ENTP', 'INTP', ..., 'INFP', 'INTJ', 'INFJ'], dtype=object)
```

In [18]:
```
#counting total posts of each type
total=ds.groupby(['type']).count()*50
total
```
Out[18]:

	posts
type	
ENFJ	1750
ENFP	5300
ENTJ	1850
ENTP	5750
ESFJ	250
ESFP	500
ESTJ	300
ESTP	800
INFJ	11800
INFP	17350
INTJ	8850
INTP	10100
ISFJ	1200
ISFP	2100



In [19]:

```
allPost=pd.DataFrame() #put array into 2D data
for j in types:
    temp1 = ds[ds['type']==j]['posts'] #making type as columns
    temp2=[]
    for i in temp1:
        temp2+=i.split('|||')
    temp3=pd.Series(temp2)  #each row is filled in order : creating 1d array i.e data is filled row wise in
    allPost[j]=temp3
```

In [20]:

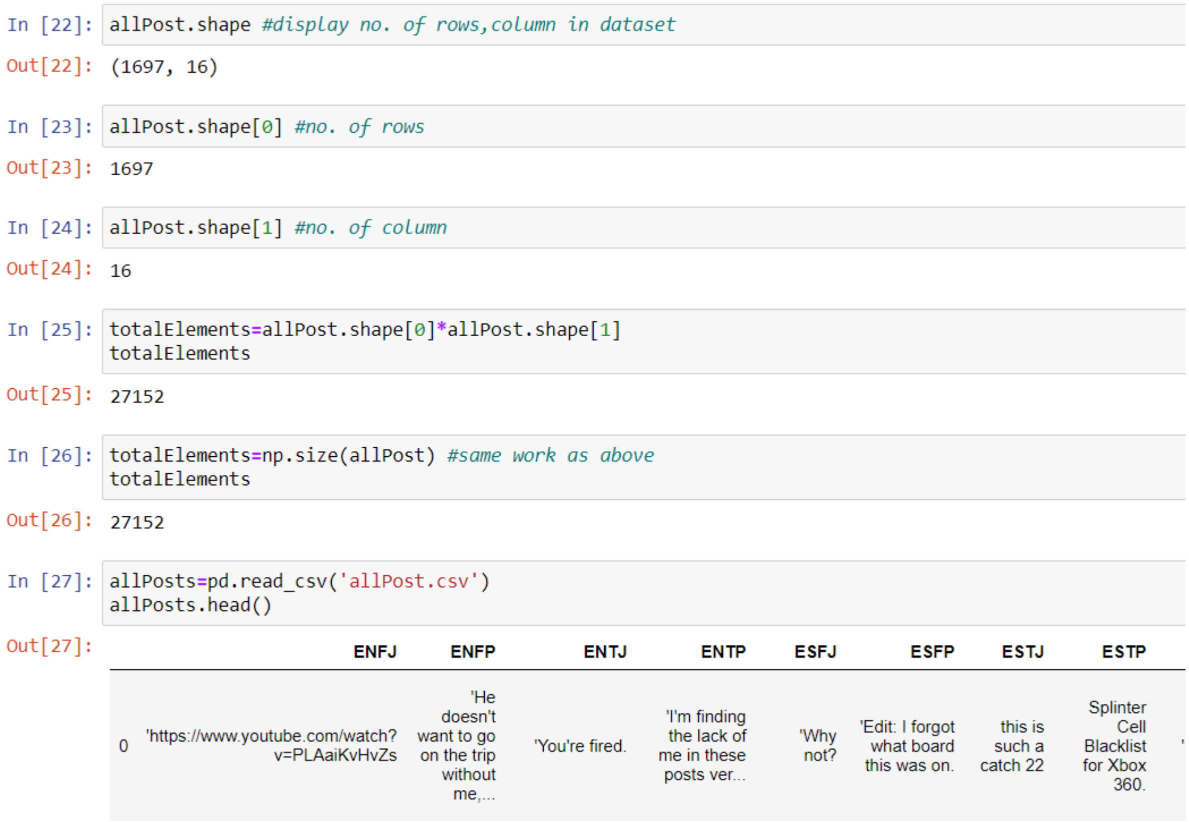
```
allPost.to_csv('allPost.csv',index=False)
```

In [21]:

```
allPost.head()
```

Out[21]:

	ENFJ	ENFP	ENTJ	ENTP	ESFJ	ESFP	ESTJ	ESTP	
0	'https://www.youtube.com/watch?v=PLAaiKvHvZs	'He doesn't want to go on the trip without me,...	'You're fired.	'I'm finding the lack of me in these posts ver...	'Why not?	'Edit: I forgot what board this was on.	this is such a catch 22	Splinter Cell Blacklist for Xbox 360.	'http://www.yo
1	51 :o	I'm still completely in AWE and I'm AMAZED tha...	That's another silly misconception. That appro...	Sex can be boring if it's in the same position...	Any other ESFJs originally mistype as an NFP? ...	I am currently reading 'Artemis Fowl: The Eter...	I'm here! Although, I'm quite the terrible EST...	ESTPs are generally well liked. If you get hat...	http://41.media.t
2	I went through a break up some months ago. We ...	Thanks, everyone. I'm struggling with being se...	But guys... he REALLY wants to go on a super-d...	Giving new meaning to 'Game' theory.	Hello again. Thanks for all your help. I know ...	Hi all, if you've got some spare time and why ...	Yikes. I do not want power...	I often come off to people with the opposite o...	enfp and intj m



In [22]:

```
allPost.shape #display no. of rows,column in dataset
```

Out[22]:

```
(1697, 16)
```

In [23]:

```
allPost.shape[0] #no. of rows
```

Out[23]:

```
1697
```

In [24]:

```
allPost.shape[1] #no. of column
```

Out[24]:

```
16
```

In [25]:

```
totalElements=allPost.shape[0]*allPost.shape[1]
totalElements
```

Out[25]:

```
27152
```

In [26]:

```
totalElements=np.size(allPost) #same work as above
totalElements
```

Out[26]:

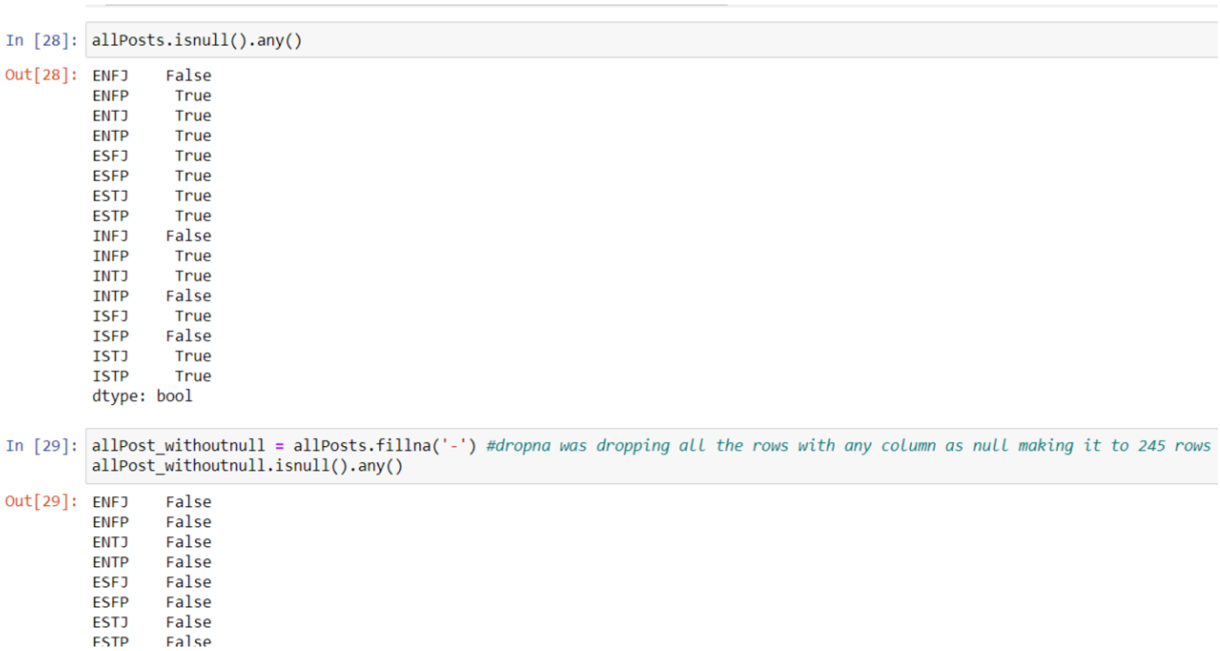
```
27152
```

In [27]:

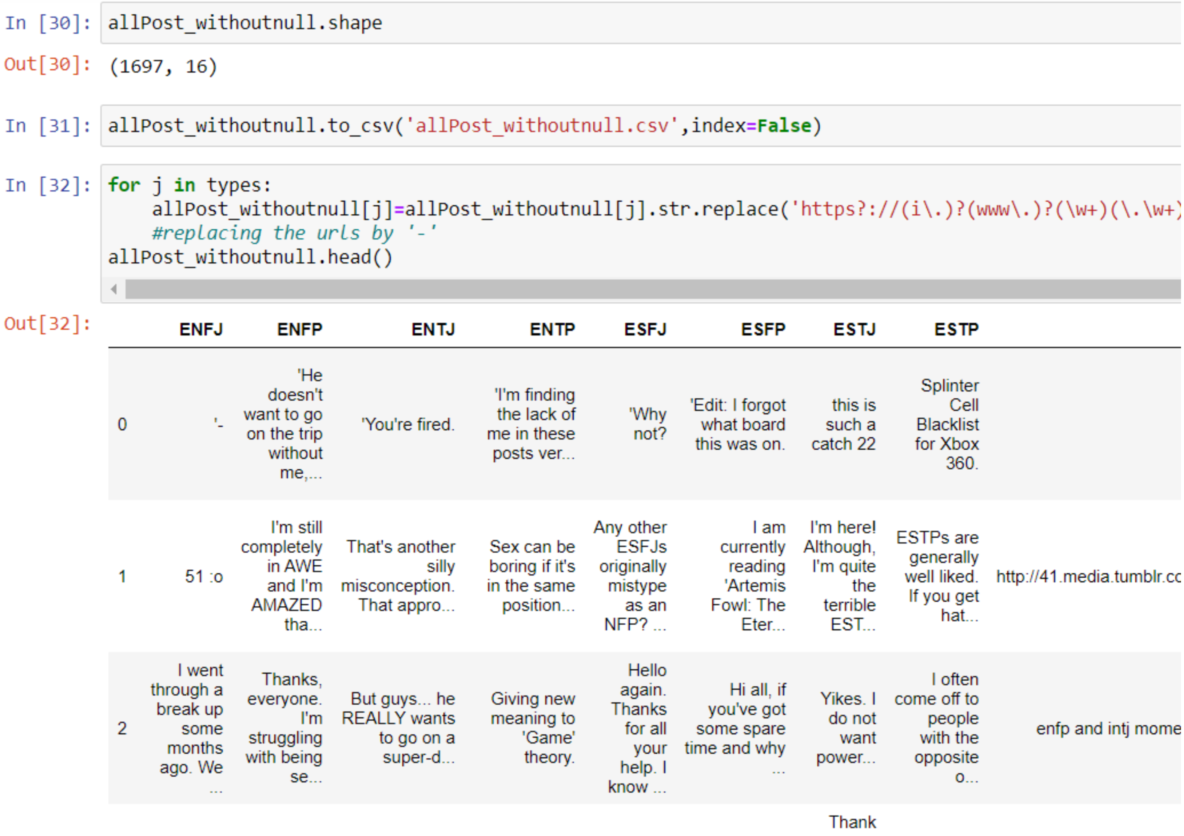
```
allPosts=pd.read_csv('allPost.csv')
allPosts.head()
```

Out[27]:

	ENFJ	ENFP	ENTJ	ENTP	ESFJ	ESFP	ESTJ	ESTP	
0	'https://www.youtube.com/watch?v=PLAaiKvHvZs	'He doesn't want to go on the trip without me,...	'You're fired.	'I'm finding the lack of me in these posts ver...	'Why not?	'Edit: I forgot what board this was on.	this is such a catch 22	Splinter Cell Blacklist for Xbox 360.	'



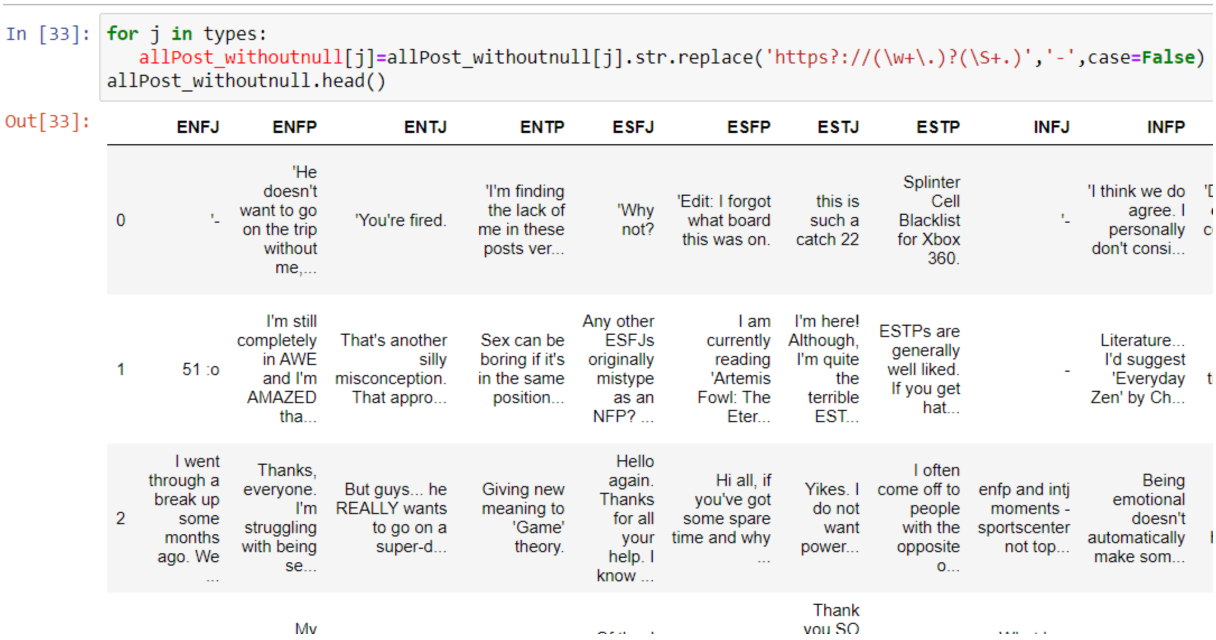
```
In [28]: allPosts.isnull().any()

Out[28]: ENFJ    False
         ENFP     True
         ENTJ     True
         ENTP     True
         ESFJ     True
         ESFP     True
         ESTJ     True
         ESTP     True
         INFJ    False
         INFP     True
         INTJ     True
         INTP    False
         ISFJ     True
         ISFP    False
         ISTJ     True
         ISTP     True
         dtype: bool

In [29]: allPost_withoutnull = allPosts.fillna('-') #dropna was dropping all the rows with any column as null making it to 245 rows
         allPost_withoutnull.isnull().any()

Out[29]: ENFJ    False
         ENFP    False
         ENTJ    False
         ENTP    False
         ESFJ    False
         ESFP    False
         ESTJ    False
         ESTP    False
```



```
In [30]: allPost_withoutnull.shape

Out[30]: (1697, 16)

In [31]: allPost_withoutnull.to_csv('allPost_withoutnull.csv',index=False)

In [32]: for j in types:
             allPost_withoutnull[j]=allPost_withoutnull[j].str.replace('https?://(i\.)?(www\.)?(\w+)(\.\w+)
             #replacing the urls by '-'
         allPost_withoutnull.head()
```

Out[32]:

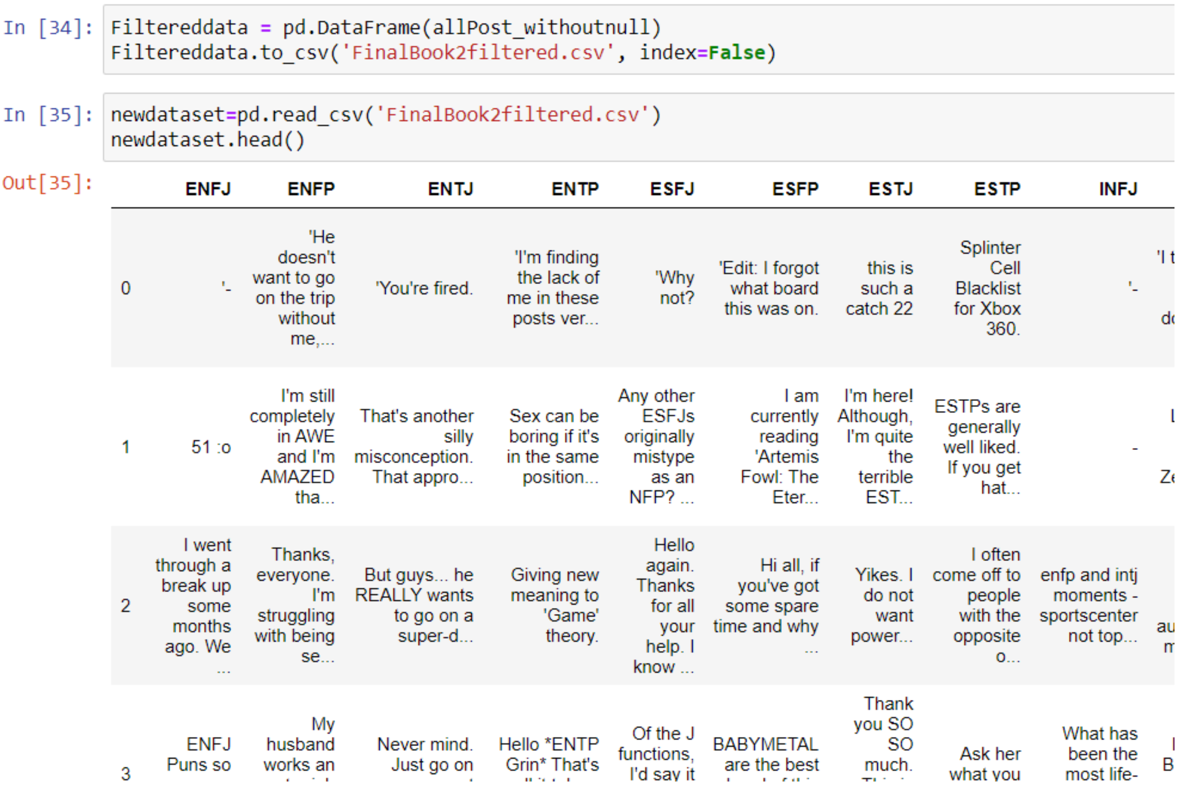
	ENFJ	ENFP	ENTJ	ENTP	ESFJ	ESFP	ESTJ	ESTP	
0	'-	'He doesn't want to go on the trip without me,...	'You're fired.	'I'm finding the lack of me in these posts ver...	'Why not?	'Edit: I forgot what board this was on.	this is such a catch 22	Splinter Cell Blacklist for Xbox 360.	
1	51 :o	I'm still completely in AWE and I'm AMAZED tha...	That's another silly misconception. That appro...	Sex can be boring if it's in the same position...	Any other ESFJs originally mistype as an NFP? ...	I am currently reading 'Artemis Fowl: The Eter...	I'm here! Although, I'm quite the terrible EST...	ESTPs are generally well liked. If you get hat...	http://41.media.tumblr.cc
2	I went through a break up some months ago. We ...	Thanks, everyone. I'm struggling with being se...	But guys... he REALLY wants to go on a super-d...	Giving new meaning to 'Game' theory.	Hello again. Thanks for all your help. I know ...	Hi all, if you've got some spare time and why ...	Yikes. I do not want power...	I often come off to people with the opposite o...	enfp and intj mome
							Thank		



In [33]:
```
for j in types:
    allPost_withoutnull[j]=allPost_withoutnull[j].str.replace('https?://(\w+\.)?(\S+.)','-',case=False)
allPost_withoutnull.head()
```

Out[33]:

	ENFJ	ENFP	ENTJ	ENTP	ESFJ	ESFP	ESTJ	ESTP	INFJ	INFP	
0	'-	'He doesn't want to go on the trip without me,...	'You're fired.	'I'm finding the lack of me in these posts ver...	'Why not?	'Edit: I forgot what board this was on.	this is such a catch 22	Splinter Cell Blacklist for Xbox 360.	'-	'I think we do agree. I personally don't consi...	'D c
1	51 :o	I'm still completely in AWE and I'm AMAZED tha...	That's another silly misconception. That appro...	Sex can be boring if it's in the same position...	Any other ESFJs originally mistype as an NFP? ...	I am currently reading 'Artemis Fowl: The Eter...	I'm here! Although, I'm quite the terrible EST...	ESTPs are generally well liked. If you get hat...	-	Literature... I'd suggest 'Everyday Zen' by Ch...	t
2	I went through a break up some months ago. We ...	Thanks, everyone. I'm struggling with being se...	But guys... he REALLY wants to go on a super-d...	Giving new meaning to 'Game' theory.	Hello again. Thanks for all your help. I know ...	Hi all, if you've got some spare time and why ...	Yikes. I do not want power...	I often come off to people with the opposite o...	enfp and intj moments - sportscenter not top...	Being emotional doesn't automatically make som...	I
		My					Thank you SO				

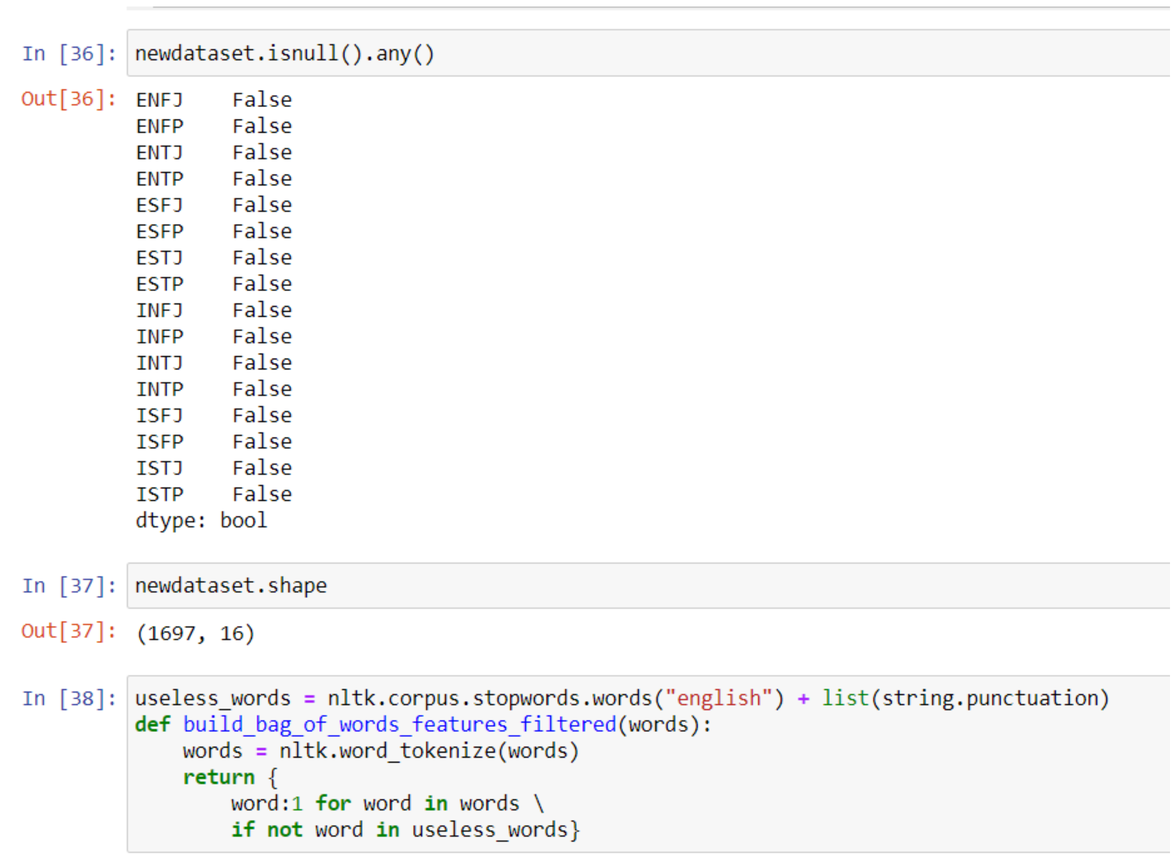


In [34]:
```
Filtereddata = pd.DataFrame(allPost_withoutnull)
Filtereddata.to_csv('FinalBook2filtered.csv', index=False)
```

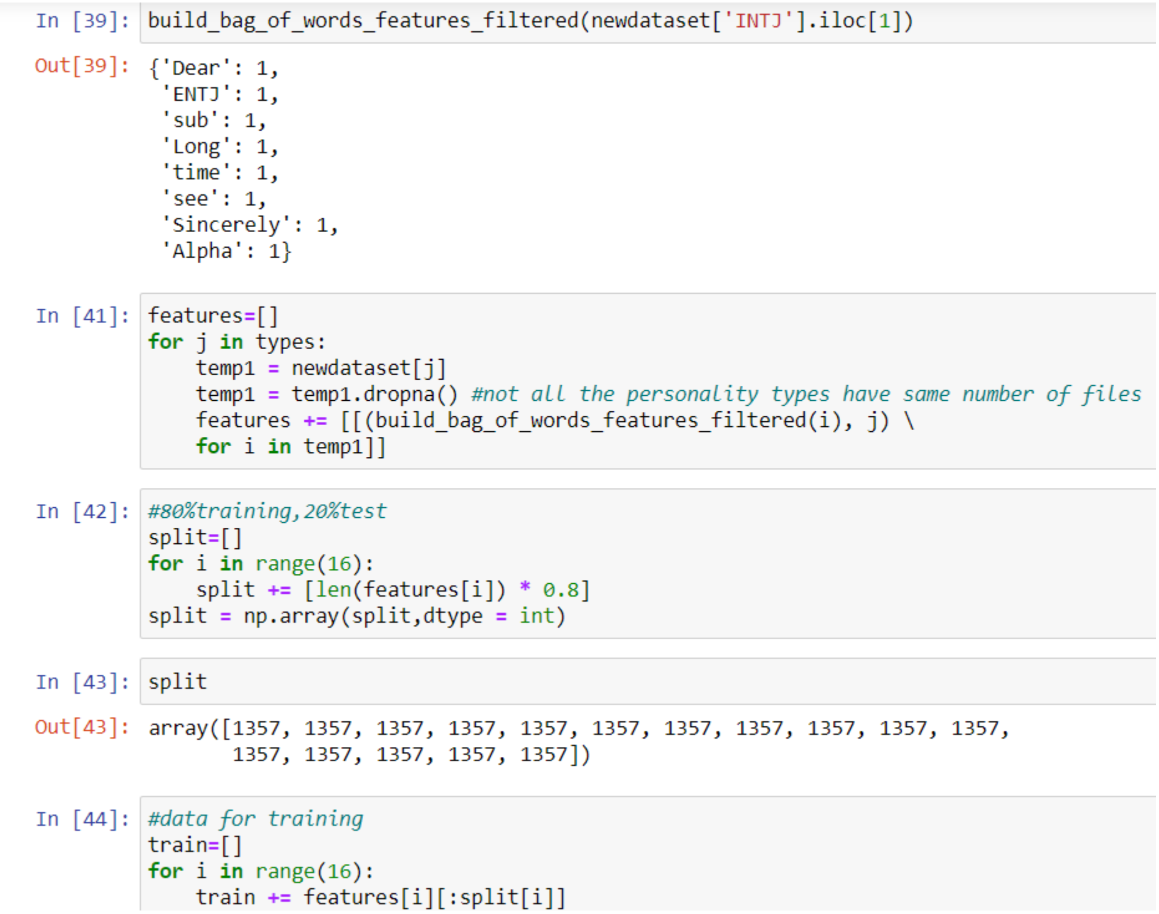
In [35]:
```
newdataset=pd.read_csv('FinalBook2filtered.csv')
newdataset.head()
```

Out[35]:

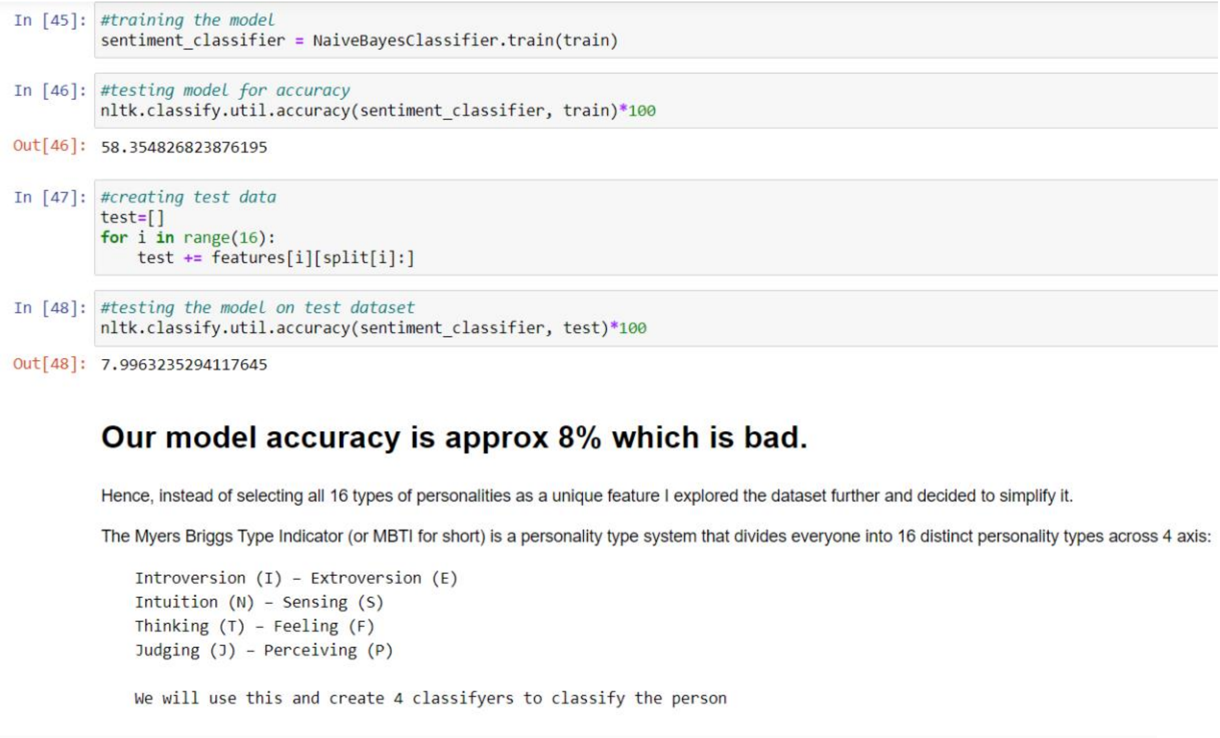
	ENFJ	ENFP	ENTJ	ENTP	ESFJ	ESFP	ESTJ	ESTP	INFJ	
0	'-	'He doesn't want to go on the trip without me,...	'You're fired.	'I'm finding the lack of me in these posts ver...	'Why not?	'Edit: I forgot what board this was on.	this is such a catch 22	Splinter Cell Blacklist for Xbox 360.	'-	'I t d
1	51 :o	I'm still completely in AWE and I'm AMAZED tha...	That's another silly misconception. That appro...	Sex can be boring if it's in the same position...	Any other ESFJs originally mistype as an NFP? ...	I am currently reading 'Artemis Fowl: The Eter...	I'm here! Although, I'm quite the terrible EST...	ESTPs are generally well liked. If you get hat...	-	l Ze
2	I went through a break up some months ago. We ...	Thanks, everyone. I'm struggling with being se...	But guys... he REALLY wants to go on a super-d...	Giving new meaning to 'Game' theory.	Hello again. Thanks for all your help. I know ...	Hi all, if you've got some spare time and why ...	Yikes. I do not want power...	I often come off to people with the opposite o...	enfp and intj moments - sportscenter not top...	au m
3	ENFJ Puns so	My husband works an	Never mind. Just go on	Hello *ENTP Grin* That's	Of the J functions, I'd say it	BABYMETAL are the best	Thank you SO SO much.	Ask her what you	What has been the most life-	I B



```
In [36]: newdataset.isnull().any()

Out[36]: ENFJ    False
         ENFP    False
         ENTJ    False
         ENTP    False
         ESFJ    False
         ESFP    False
         ESTJ    False
         ESTP    False
         INFJ    False
         INFP    False
         INTJ    False
         INTP    False
         ISFJ    False
         ISFP    False
         ISTJ    False
         ISTP    False
         dtype: bool

In [37]: newdataset.shape

Out[37]: (1697, 16)

In [38]: useless_words = nltk.corpus.stopwords.words("english") + list(string.punctuation)
         def build_bag_of_words_features_filtered(words):
             words = nltk.word_tokenize(words)
             return {
                 word:1 for word in words \
                 if not word in useless_words}
```



```
In [39]: build_bag_of_words_features_filtered(newdataset['INTJ'].iloc[1])

Out[39]: {'Dear': 1,
          'ENTJ': 1,
          'sub': 1,
          'Long': 1,
          'time': 1,
          'see': 1,
          'Sincerely': 1,
          'Alpha': 1}

In [41]: features=[]
         for j in types:
             temp1 = newdataset[j]
             temp1 = temp1.dropna() #not all the personality types have same number of files
             features += [[(build_bag_of_words_features_filtered(i), j) \
             for i in temp1]]

In [42]: #80%training,20%test
         split=[]
         for i in range(16):
             split += [len(features[i]) * 0.8]
         split = np.array(split,dtype = int)

In [43]: split

Out[43]: array([1357, 1357, 1357, 1357, 1357, 1357, 1357, 1357, 1357, 1357, 1357,
                1357, 1357, 1357, 1357, 1357])

In [44]: #data for training
         train=[]
         for i in range(16):
             train += features[i][:split[i]]
```



```
In [45]: #training the model
         sentiment_classifier = NaiveBayesClassifier.train(train)
```

```
In [46]: #testing model for accuracy
         nltk.classify.util.accuracy(sentiment_classifier, train)*100

Out[46]: 58.354826823876195
```

```
In [47]: #creating test data
         test=[]
         for i in range(16):
             test += features[i][split[i]:]
```

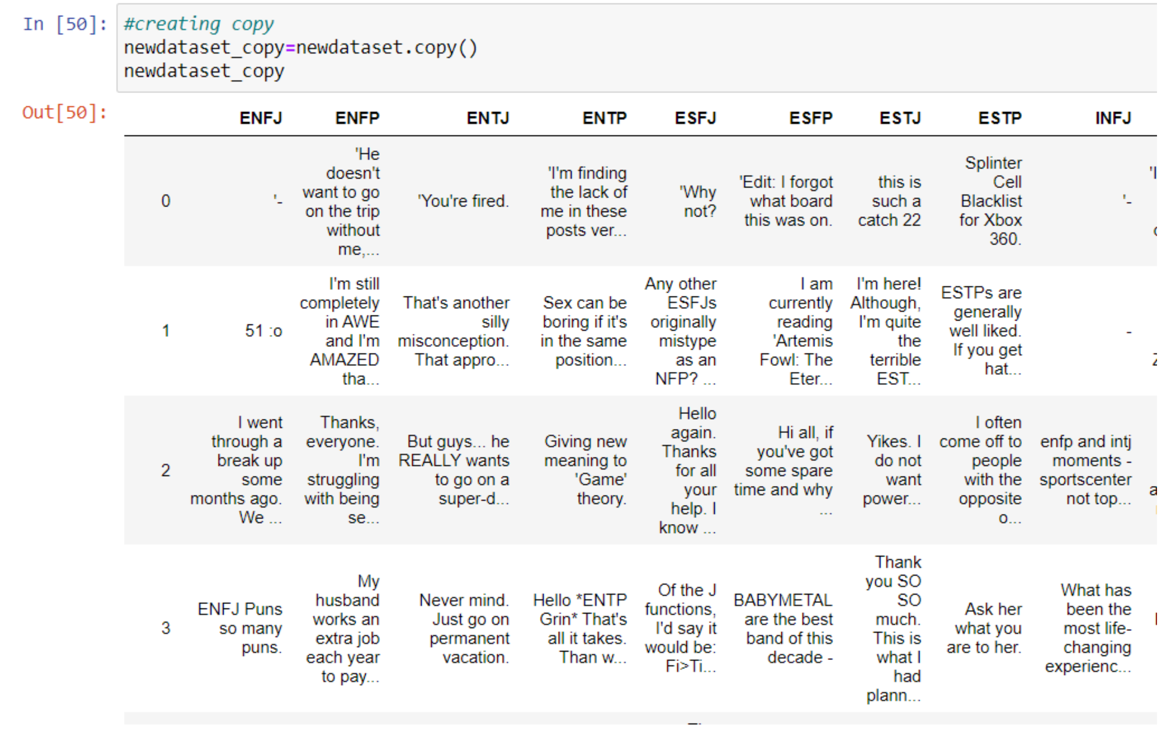
```
In [48]: #testing the model on test dataset
         nltk.classify.util.accuracy(sentiment_classifier, test)*100

Out[48]: 7.9963235294117645
```

Our model accuracy is approx 8% which is bad.

Hence, instead of selecting all 16 types of personalities as a unique feature I explored the dataset further and decided to simplify it.

The Myers Briggs Type Indicator (or MBTI for short) is a personality type system that divides everyone into 16 distinct personality types across 4 axis:

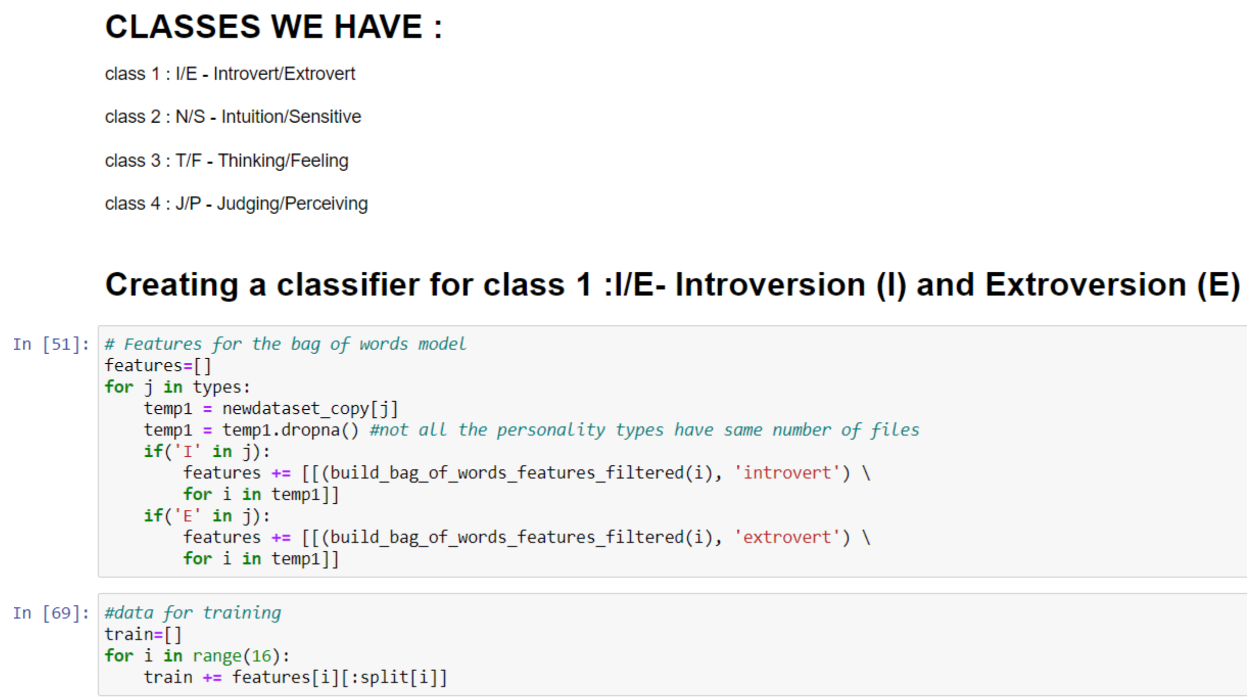
```
Introversion (I) – Extroversion (E)
Intuition (N) – Sensing (S)
Thinking (T) – Feeling (F)
Judging (J) – Perceiving (P)

We will use this and create 4 classifyers to classify the person
```



```
In [50]: #creating copy
         newdataset_copy=newdataset.copy()
         newdataset_copy
```

Out[50]:

	ENFJ	ENFP	ENTJ	ENTP	ESFJ	ESFP	ESTJ	ESTP	INFJ
0	'-	'He doesn't want to go on the trip without me,...	'You're fired.	'I'm finding the lack of me in these posts ver...	'Why not?	'Edit: I forgot what board this was on.	this is such a catch 22	Splinter Cell Blacklist for Xbox 360.	'-
1	51 :o	I'm still completely in AWE and I'm AMAZED tha...	That's another silly misconception. That appro...	Sex can be boring if it's in the same position...	Any other ESFJs originally mistype as an NFP? ...	I am currently reading 'Artemis Fowl: The Eter...	I'm here! Although, I'm quite the terrible EST...	ESTPs are generally well liked. If you get hat...	-
2	I went through a break up some months ago. We ...	Thanks, everyone. I'm struggling with being se...	But guys... he REALLY wants to go on a super-d...	Giving new meaning to 'Game' theory.	Hello again. Thanks for all your help. I know ...	Hi all, if you've got some spare time and why ...	Yikes. I do not want power...	I often come off to people with the opposite o...	enfp and intj moments - sportscenter not top...
3	ENFJ Puns so many puns.	My husband works an extra job each year to pay...	Never mind. Just go on permanent vacation.	Hello *ENTP Grin* That's all it takes. Than w...	Of the J functions, I'd say it would be: Fi>Ti...	BABYMETAL are the best band of this decade -	Thank you SO SO much. This is what I had plann...	Ask her what you are to her.	What has been the most life-changing experienc...



CLASSES WE HAVE :

class 1 : I/E - Introvert/Extrovert

class 2 : N/S - Intuition/Sensitive

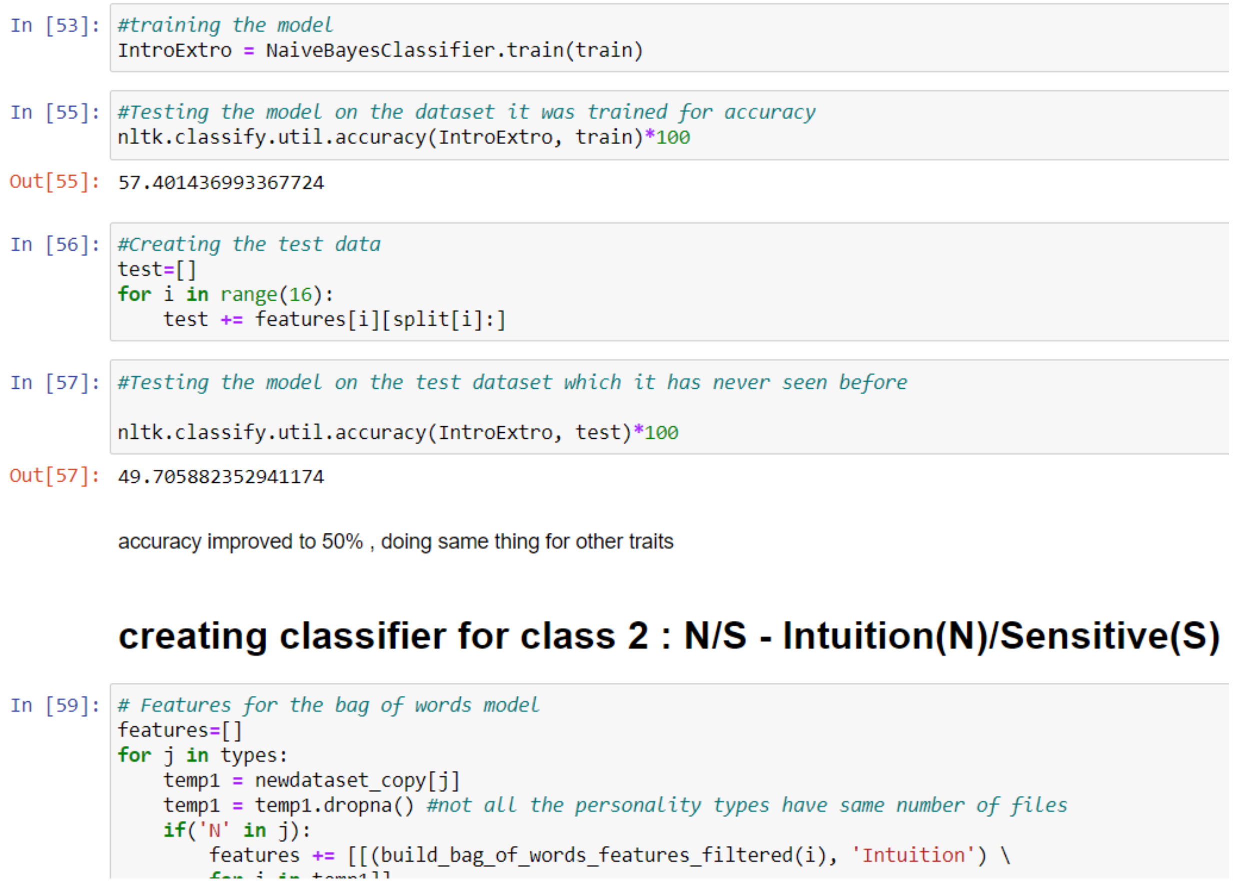
class 3 : T/F - Thinking/Feeling

class 4 : J/P - Judging/Perceiving

Creating a classifier for class 1 :I/E- Introversion (I) and Extroversion (E)

```
In [51]: # Features for the bag of words model
         features=[]
         for j in types:
             temp1 = newdataset_copy[j]
             temp1 = temp1.dropna() #not all the personality types have same number of files
             if('I' in j):
                 features += [[(build_bag_of_words_features_filtered(i), 'introvert') \
                 for i in temp1]]
             if('E' in j):
                 features += [[(build_bag_of_words_features_filtered(i), 'extrovert') \
                 for i in temp1]]
```

```
In [69]: #data for training
         train=[]
         for i in range(16):
             train += features[i][:split[i]]
```



```
In [53]: #training the model
         IntroExtro = NaiveBayesClassifier.train(train)
```

```
In [55]: #Testing the model on the dataset it was trained for accuracy
         nltk.classify.util.accuracy(IntroExtro, train)*100

Out[55]: 57.401436993367724
```

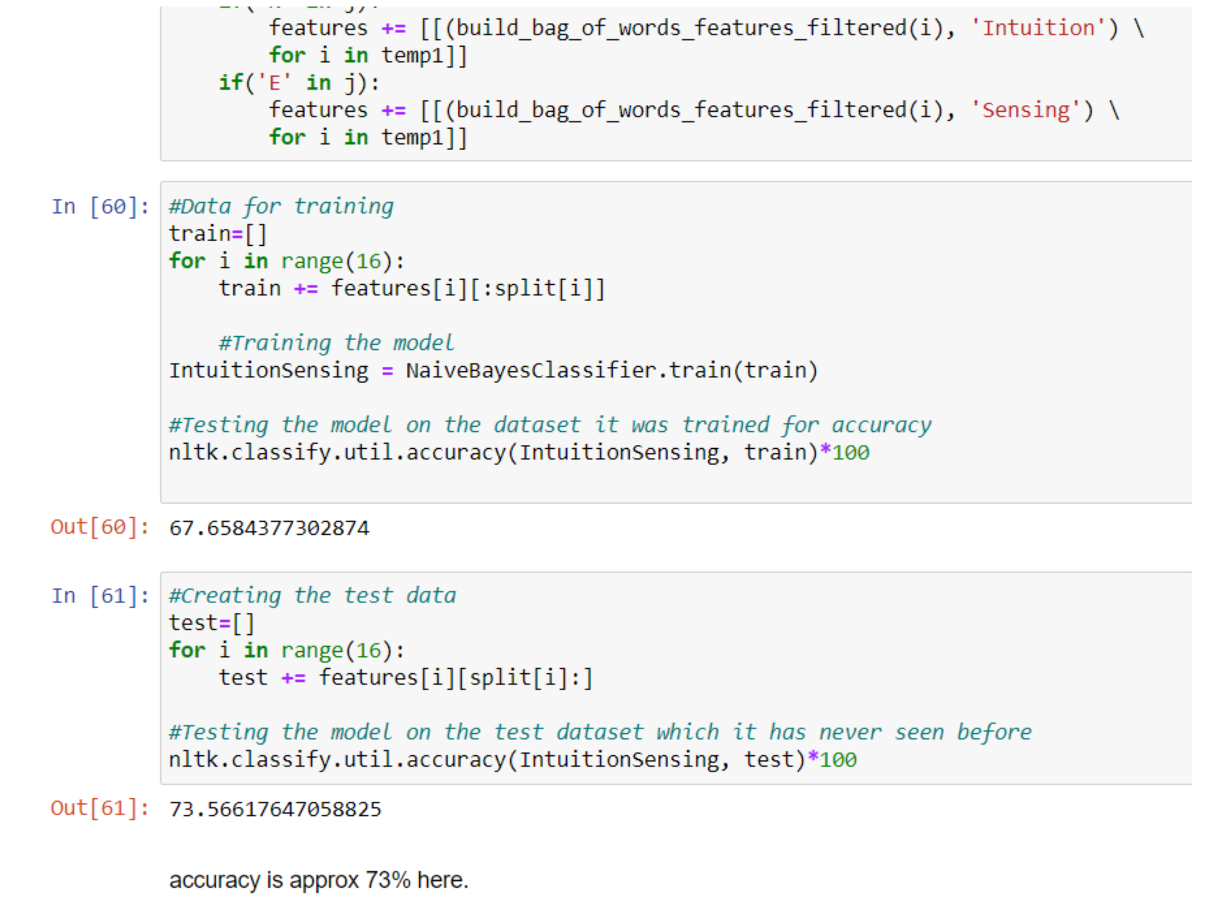
```
In [56]: #Creating the test data
         test=[]
         for i in range(16):
             test += features[i][split[i]:]
```

```
In [57]: #Testing the model on the test dataset which it has never seen before

         nltk.classify.util.accuracy(IntroExtro, test)*100

Out[57]: 49.705882352941174
```

accuracy improved to 50% , doing same thing for other traits

creating classifier for class 2 : N/S - Intuition(N)/Sensitive(S)

```
In [59]: # Features for the bag of words model
         features=[]
         for j in types:
             temp1 = newdataset_copy[j]
             temp1 = temp1.dropna() #not all the personality types have same number of files
             if('N' in j):
                 features += [[(build_bag_of_words_features_filtered(i), 'Intuition') \
                 for i in temp1]]
```

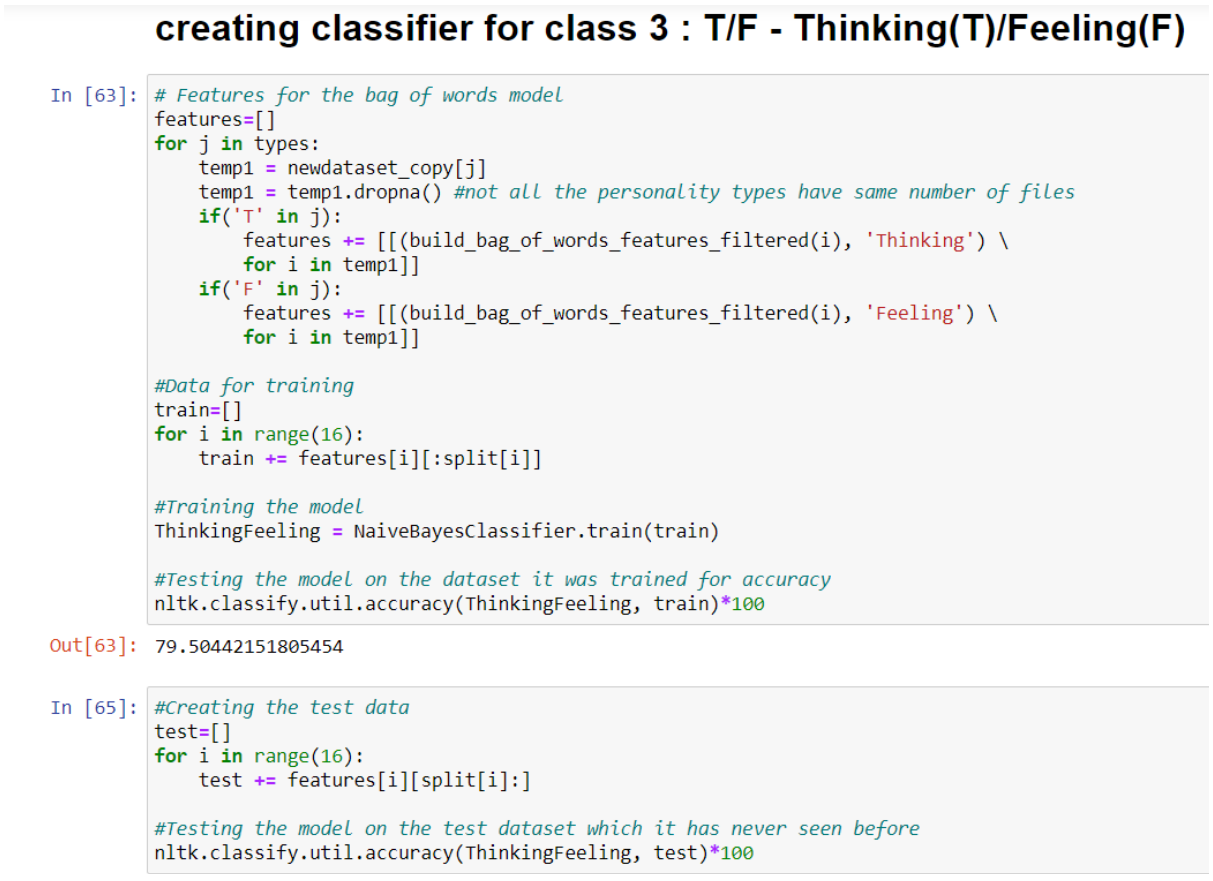


```
        features += [[(build_bag_of_words_features_filtered(i), 'Intuition') \
        for i in temp1]]
    if('E' in j):
        features += [[(build_bag_of_words_features_filtered(i), 'Sensing') \
        for i in temp1]]
```

```
In [60]: #Data for training
         train=[]
         for i in range(16):
             train += features[i][:split[i]]

             #Training the model
         IntuitionSensing = NaiveBayesClassifier.train(train)

         #Testing the model on the dataset it was trained for accuracy
         nltk.classify.util.accuracy(IntuitionSensing, train)*100

Out[60]: 67.6584377302874
```

```
In [61]: #Creating the test data
         test=[]
         for i in range(16):
             test += features[i][split[i]:]

         #Testing the model on the test dataset which it has never seen before
         nltk.classify.util.accuracy(IntuitionSensing, test)*100

Out[61]: 73.56617647058825
```

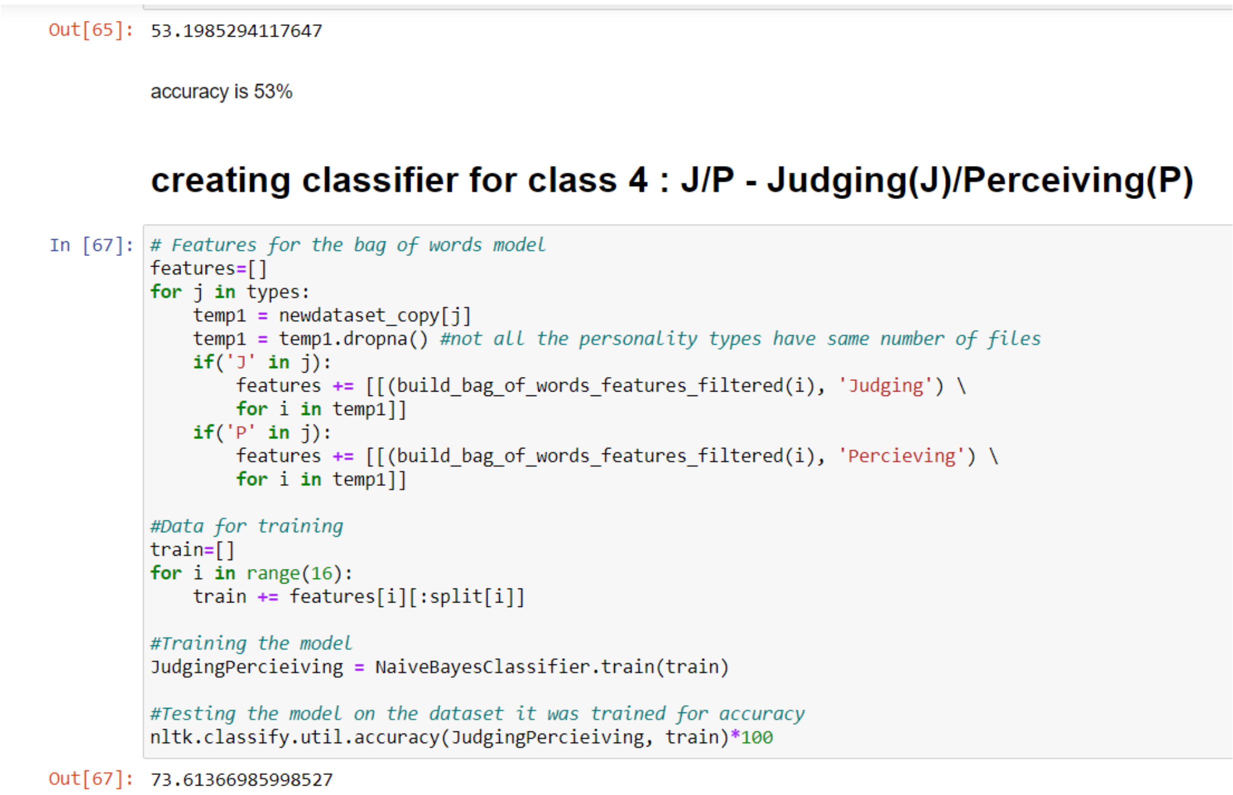
accuracy is approx 73% here.



creating classifier for class 3 : T/F - Thinking(T)/Feeling(F)

```
In [63]: # Features for the bag of words model
         features=[]
         for j in types:
             temp1 = newdataset_copy[j]
             temp1 = temp1.dropna() #not all the personality types have same number of files
             if('T' in j):
                 features += [[(build_bag_of_words_features_filtered(i), 'Thinking') \
                 for i in temp1]]
             if('F' in j):
                 features += [[(build_bag_of_words_features_filtered(i), 'Feeling') \
                 for i in temp1]]

         #Data for training
         train=[]
         for i in range(16):
             train += features[i][:split[i]]

         #Training the model
         ThinkingFeeling = NaiveBayesClassifier.train(train)

         #Testing the model on the dataset it was trained for accuracy
         nltk.classify.util.accuracy(ThinkingFeeling, train)*100

Out[63]: 79.50442151805454
```

```
In [65]: #Creating the test data
         test=[]
         for i in range(16):
             test += features[i][split[i]:]

         #Testing the model on the test dataset which it has never seen before
         nltk.classify.util.accuracy(ThinkingFeeling, test)*100
```

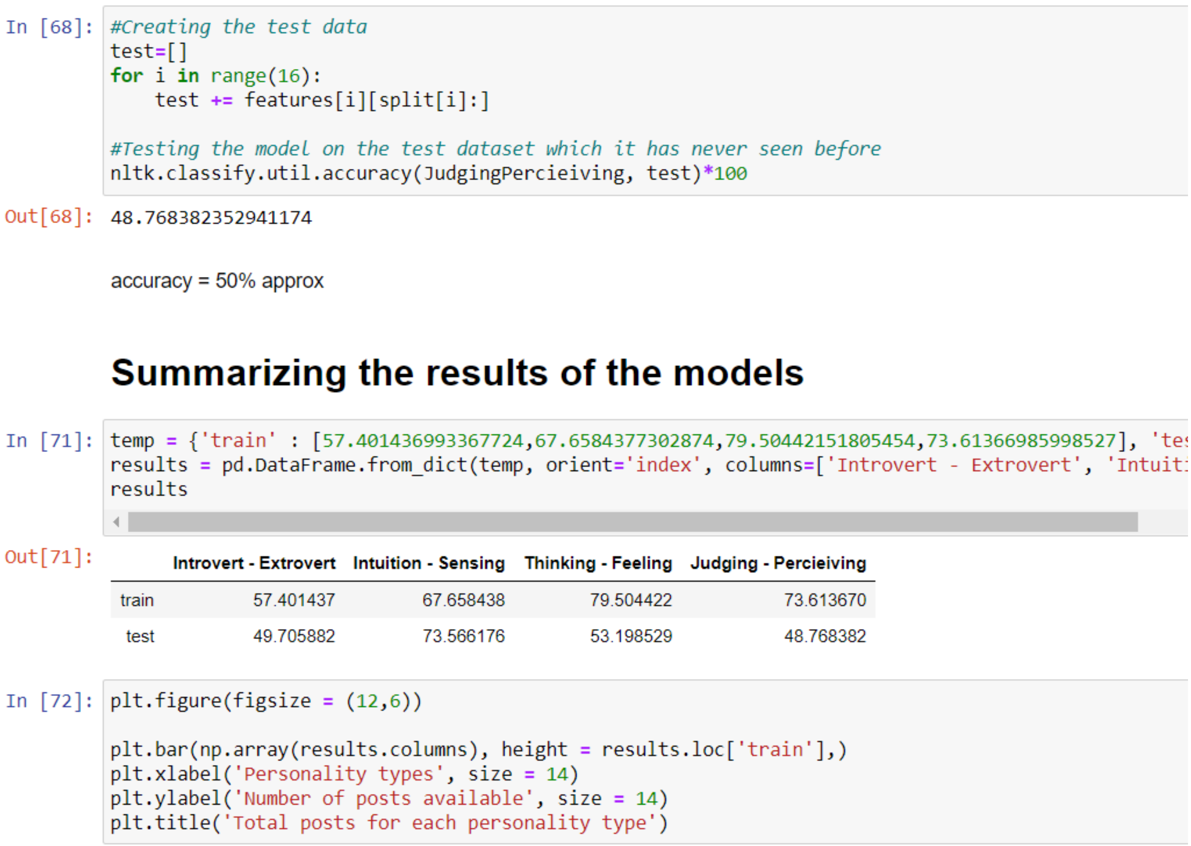


```
Out[65]: 53.1985294117647
```

accuracy is 53%

creating classifier for class 4 : J/P - Judging(J)/Perceiving(P)

```
In [67]: # Features for the bag of words model
         features=[]
         for j in types:
             temp1 = newdataset_copy[j]
             temp1 = temp1.dropna() #not all the personality types have same number of files
             if('J' in j):
                 features += [[(build_bag_of_words_features_filtered(i), 'Judging') \
                 for i in temp1]]
             if('P' in j):
                 features += [[(build_bag_of_words_features_filtered(i), 'Percieving') \
                 for i in temp1]]

         #Data for training
         train=[]
         for i in range(16):
             train += features[i][:split[i]]

         #Training the model
         JudgingPercieiving = NaiveBayesClassifier.train(train)

         #Testing the model on the dataset it was trained for accuracy
         nltk.classify.util.accuracy(JudgingPercieiving, train)*100

Out[67]: 73.61366985998527
```



```
In [68]: #Creating the test data
         test=[]
         for i in range(16):
             test += features[i][split[i]:]

         #Testing the model on the test dataset which it has never seen before
         nltk.classify.util.accuracy(JudgingPercieiving, test)*100

Out[68]: 48.768382352941174
```

accuracy = 50% approx

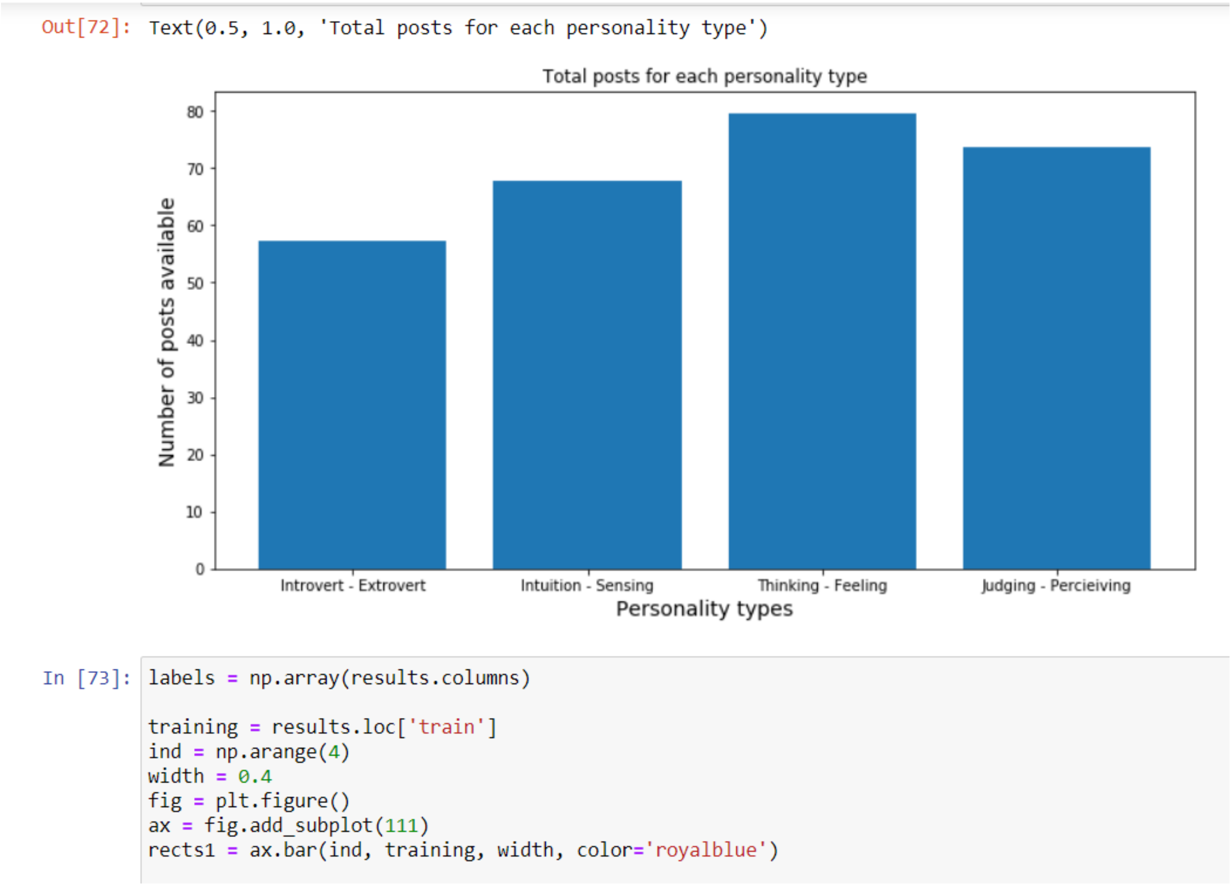
Summarizing the results of the models

```
In [71]: temp = {'train' : [57.401436993367724,67.6584377302874,79.50442151805454,73.61366985998527], 'tes
         results = pd.DataFrame.from_dict(temp, orient='index', columns=['Introvert - Extrovert', 'Intuit
         results
```

Out[71]:

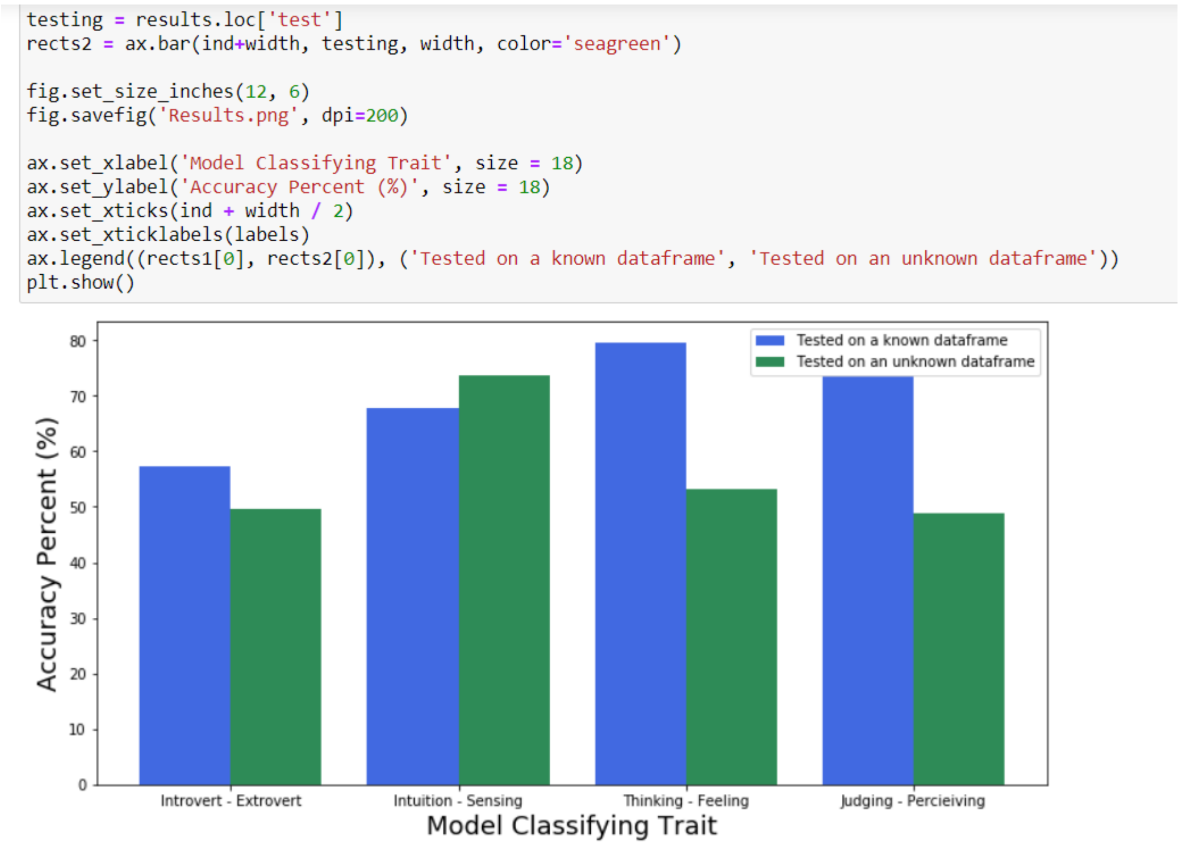
	Introvert - Extrovert	Intuition - Sensing	Thinking - Feeling	Judging - Percieiving
train	57.401437	67.658438	79.504422	73.613670
test	49.705882	73.566176	53.198529	48.768382

```
In [72]: plt.figure(figsize = (12,6))

         plt.bar(np.array(results.columns), height = results.loc['train'],)
         plt.xlabel('Personality types', size = 14)
         plt.ylabel('Number of posts available', size = 14)
         plt.title('Total posts for each personality type')
```

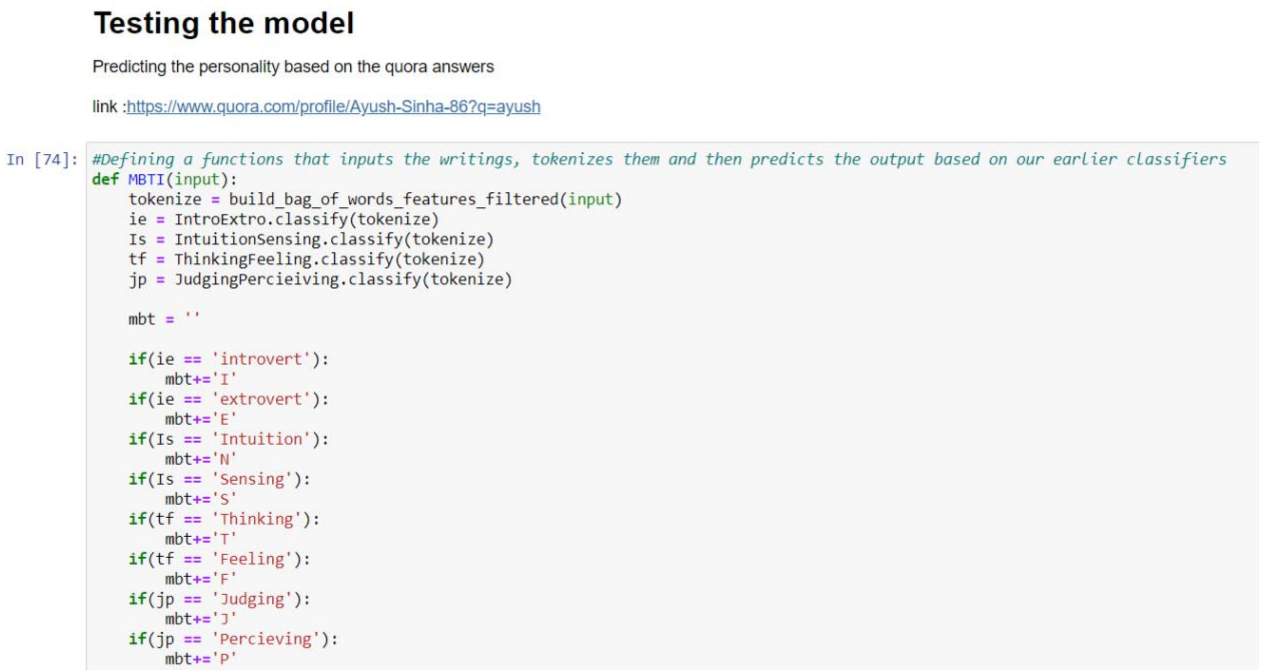


```
Out[72]: Text(0.5, 1.0, 'Total posts for each personality type')
```

```
In [73]: labels = np.array(results.columns)

         training = results.loc['train']
         ind = np.arange(4)
         width = 0.4
         fig = plt.figure()
         ax = fig.add_subplot(111)
         rects1 = ax.bar(ind, training, width, color='royalblue')
```



```
testing = results.loc['test']
rects2 = ax.bar(ind+width, testing, width, color='seagreen')

fig.set_size_inches(12, 6)
fig.savefig('Results.png', dpi=200)

ax.set_xlabel('Model Classifying Trait', size = 18)
ax.set_ylabel('Accuracy Percent (%)', size = 18)
ax.set_xticks(ind + width / 2)
ax.set_xticklabels(labels)
ax.legend((rects1[0], rects2[0]), ('Tested on a known dataframe', 'Tested on an unknown dataframe'))
plt.show()
```

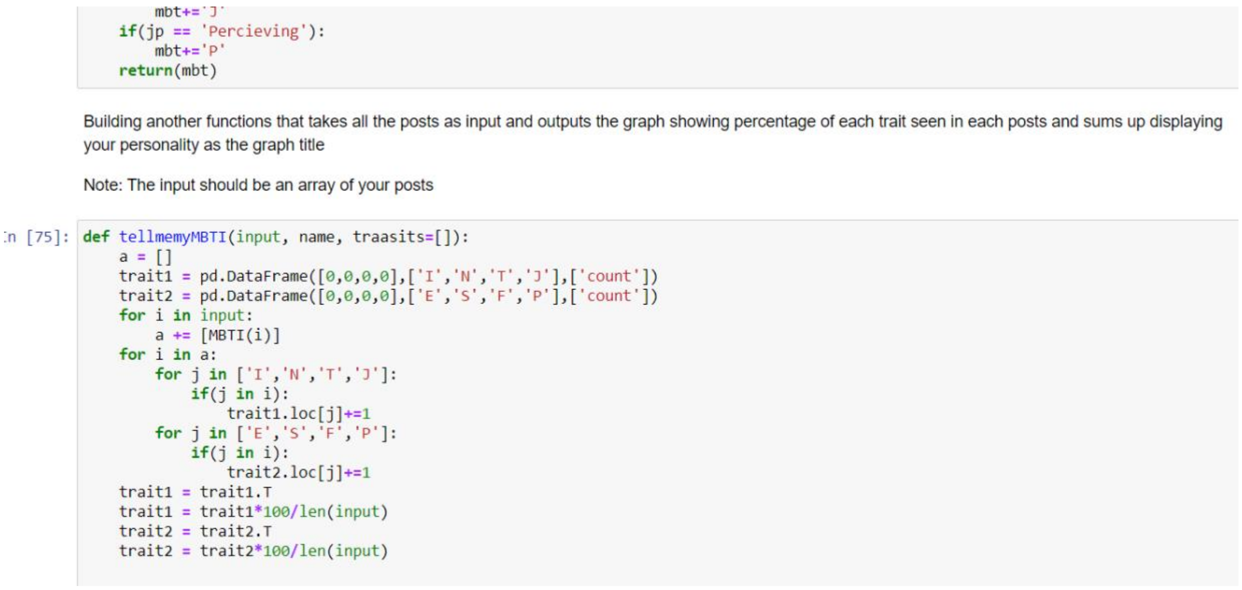


Testing the model

Predicting the personality based on the quora answers

link :https://www.quora.com/profile/Ayush-Sinha-86?q=ayush

In [74]:
```
#Defining a functions that inputs the writings, tokenizes them and then predicts the output based on our earlier classifiers
def MBTI(input):
    tokenize = build_bag_of_words_features_filtered(input)
    ie = IntroExtro.classify(tokenize)
    Is = IntuitionSensing.classify(tokenize)
    tf = ThinkingFeeling.classify(tokenize)
    jp = JudgingPercieiving.classify(tokenize)

    mbt = ''

    if(ie == 'introvert'):
        mbt+='I'
    if(ie == 'extrovert'):
        mbt+='E'
    if(Is == 'Intuition'):
        mbt+='N'
    if(Is == 'Sensing'):
        mbt+='S'
    if(tf == 'Thinking'):
        mbt+='T'
    if(tf == 'Feeling'):
        mbt+='F'
    if(jp == 'Judging'):
        mbt+='J'
    if(jp == 'Percieving'):
        mbt+='P'
```



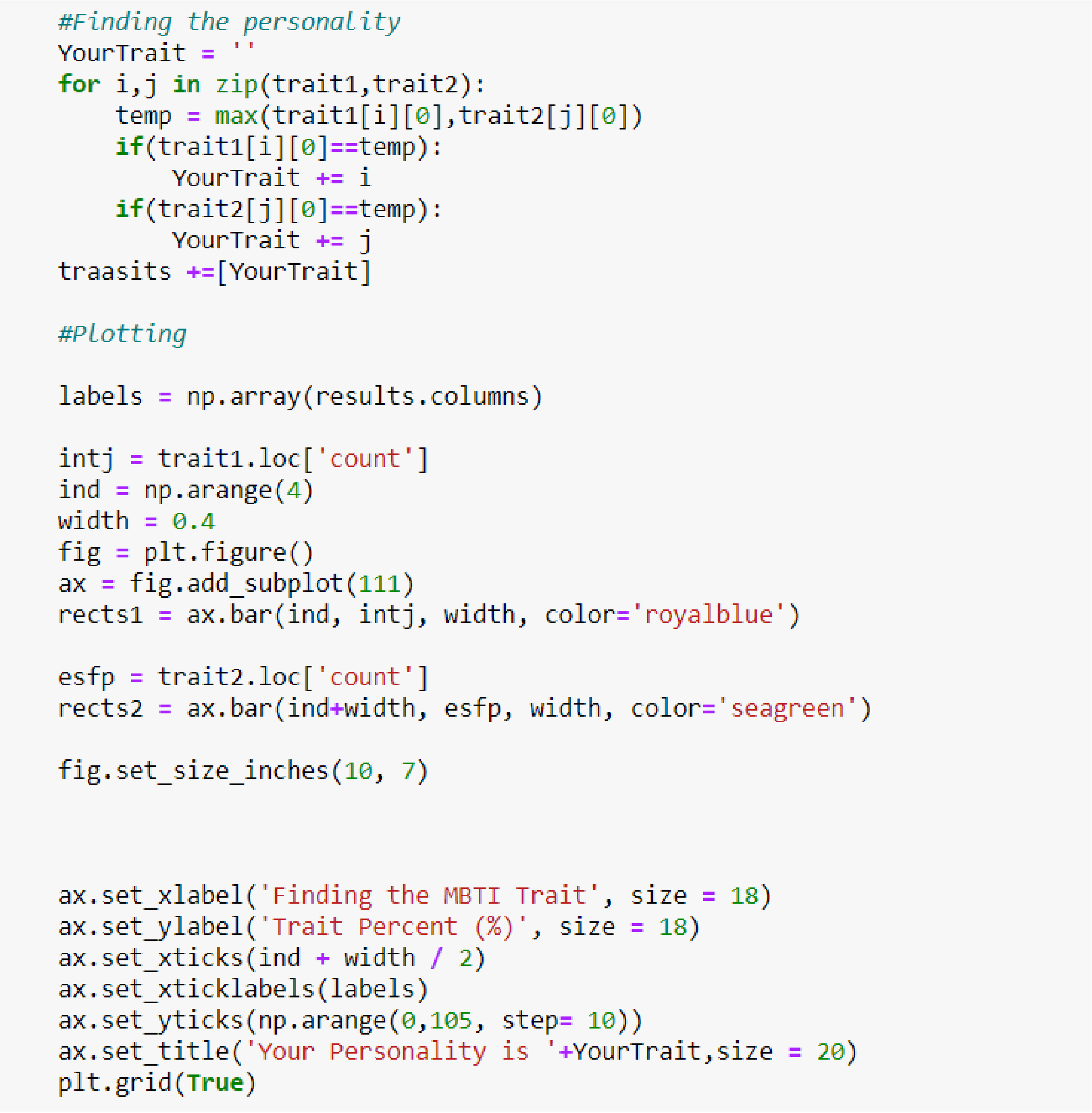
```
        mbt+='J'
    if(jp == 'Percieving'):
        mbt+='P'
```

```
    return(mbt)
```

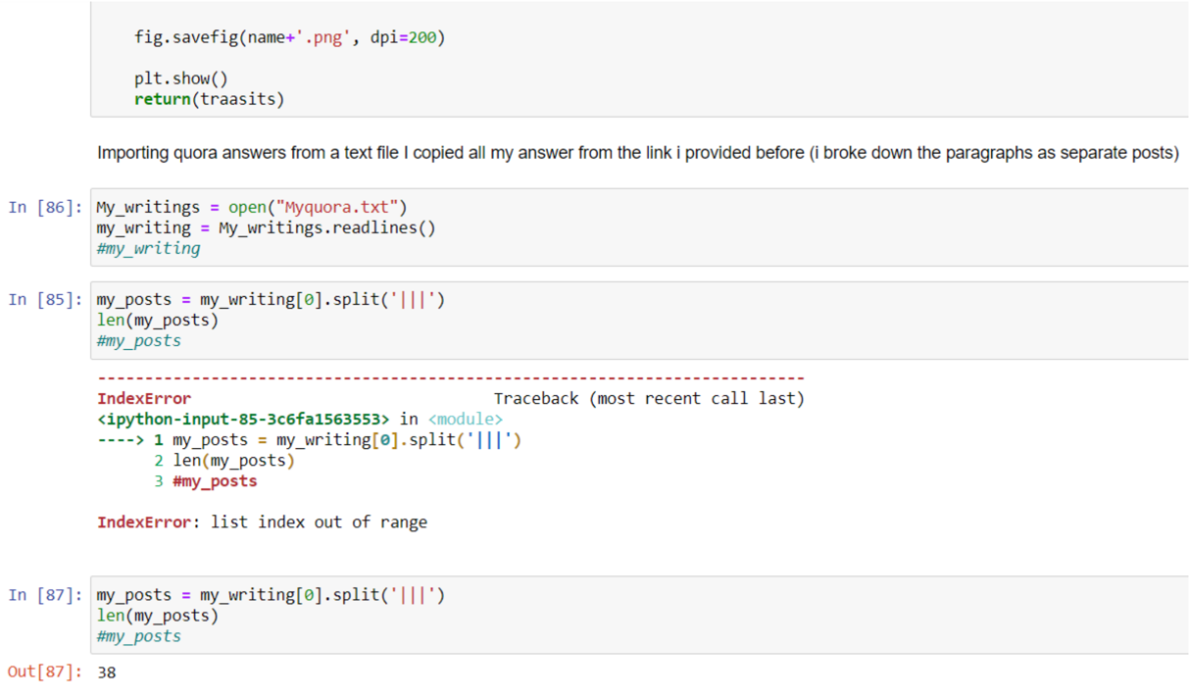
Building another functions that takes all the posts as input and outputs the graph showing percentage of each trait seen in each posts and sums up displaying your personality as the graph title

Note: The input should be an array of your posts

In [75]:
```
def tellmemyMBTI(input, name, traasits=[]):
    a = []
    trait1 = pd.DataFrame([0,0,0,0],['I','N','T','J'],['count'])
    trait2 = pd.DataFrame([0,0,0,0],['E','S','F','P'],['count'])
    for i in input:
        a += [MBTI(i)]
    for i in a:
        for j in ['I','N','T','J']:
            if(j in i):
                trait1.loc[j]+=1
        for j in ['E','S','F','P']:
            if(j in i):
                trait2.loc[j]+=1
    trait1 = trait1.T
    trait1 = trait1*100/len(input)
    trait2 = trait2.T
    trait2 = trait2*100/len(input)
```



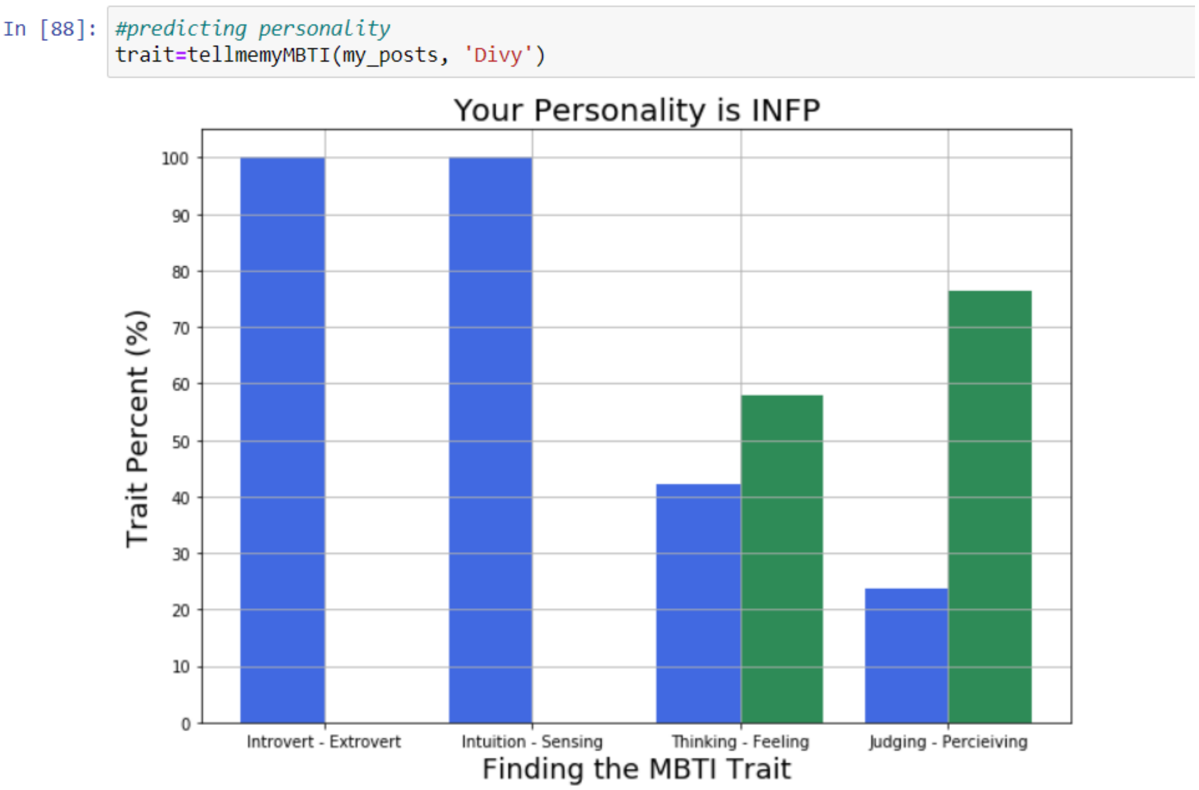
```
#Finding the personality
YourTrait = ''
for i,j in zip(trait1,trait2):
    temp = max(trait1[i][0],trait2[j][0])
    if(trait1[i][0]==temp):
        YourTrait += i
    if(trait2[j][0]==temp):
        YourTrait += j
traasits +=[YourTrait]

#Plotting

labels = np.array(results.columns)

intj = trait1.loc['count']
ind = np.arange(4)
width = 0.4
fig = plt.figure()
ax = fig.add_subplot(111)
rects1 = ax.bar(ind, intj, width, color='royalblue')

esfp = trait2.loc['count']
rects2 = ax.bar(ind+width, esfp, width, color='seagreen')

fig.set_size_inches(10, 7)



ax.set_xlabel('Finding the MBTI Trait', size = 18)
ax.set_ylabel('Trait Percent (%)', size = 18)
ax.set_xticks(ind + width / 2)
ax.set_xticklabels(labels)
ax.set_yticks(np.arange(0,105, step= 10))
ax.set_title('Your Personality is '+YourTrait,size = 20)
plt.grid(True)
```



```
    fig.savefig(name+'.png', dpi=200)

    plt.show()
    return(traasits)
```

Importing quora answers from a text file I copied all my answer from the link i provided before (i broke down the paragraphs as separate posts)

```
In [86]: My_writings = open("Myquora.txt")
         my_writing = My_writings.readlines()
         #my_writing

In [85]: my_posts = my_writing[0].split('|||')
         len(my_posts)
         #my_posts

         ---------------------------------------------------------------------------
         IndexError                                Traceback (most recent call last)
         <ipython-input-85-3c6fa1563553> in <module>
         ----> 1 my_posts = my_writing[0].split('|||')
               2 len(my_posts)
               3 #my_posts

         IndexError: list index out of range

In [87]: my_posts = my_writing[0].split('|||')
         len(my_posts)
         #my_posts

Out[87]: 38
```



```
In [88]: #predicting personality
         trait=tellmemyMBTI(my_posts, 'Divy')
```