**Personality Prediction from Social Data**

A PROJECT REPORT

Submitted By

Aman Aryan 1651137

Ankit Kumar 1651145

Abhishek 1651155

Anisha Chhetri 1651243

*Under the supervision of*

*Amitabha Acharya < Assistant Professor >*

In partial fulfilment for the award of the degree

Of

Bachelor of Technology < B. Tech >

IN

Computer Science and Engineering

Heritage Institute of Technology

An Autonomy Institute under

Maulana Abul Kalam Azad University of Technology

Formerly Known as

West Bengal University of Technology

July, 2020

**DECLARATION 20/12/2019**

We hereby declare that project report entitled “Personality Prediction from Social Data” submitted is a record of an original work done by us under the guidance of Amitabha Acharya,Assistant Professorof Computer Science and Engineering, HERITAGE INSTITUTE OF TECHNOLOGY, KOLKATA and this project work is submitted in partial fulfilment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering. The result embodied has not been submitted to any other courses or University or Institute of any degree by us.

Aman Aryan

Anisha Chhetri

Ankit Kumar

Abhishek

**ACKNOWLEDGEMENT 20/12/2019**

First and foremost, we would like to thank our mentor Amitabha Acharya who guided us through doing this project, provided us with invaluable advices, helped us in difficult period and provided us the necessary contents and documents that helped us in making our project. His willingness to motivate us contributed tremendously to the success of this project. Beside we would like to thank all CSE teachers who helped us by giving advice and providing equipment which we needed.

Also, we would like to thank our parents for their encouragement and support. At last we would like to thank all those who motivated and helped us.

Aman Aryan

Anisha Chhetri

Ankit Kumar

Abhishek

**CONTENTS**

1. ABSTRACT 5-6

2. DESIGN AND IMPLEMENTATION 7-10

2.1 DATA EXTRACTION

2.2 TEXT PROCESSING

2.3 PERSONALITY CLASSIFICATION

2.4 ANALYSIS AND STORAGE

3. RESOURCES REQUIRED 11

4. WORKING AND PERFORMANCE 12

5. LIMITATIONS 13

6. FUTURE PLANS 13

**1. ABSTRACT**

Personality has been studies extensively in social science and psychology as it reflects the way people behave and react in online social media and in the society. For instance, it correlates with music taste, Extroverted people tend to like popular music, while open to experience people are more likely to enjoy unpopular one. Personality is also related to the formation of social relations, the pages that people like on Facebook, and the language that people use to communicate.

People are increasingly using social media platforms, such as Twitter, Facebook, and Pinterest, to share their thoughts and opinions with their friends or people who are interested. Such scale of social media platforms provides us with an unprecedented opportunity to understanding psychological attributes on a large user base. In this project, we want to analyse and predict personality by constructing a bridge between personality and language in popular social media such as Twitter. Specifically, we aim to find the linguistic features that distinguish people with different personality types and explore how these features can be explain by personality. Further, using the these features we want to understand the degree to which we can predict personality traits from social media language.

However, little research has touch upon understanding personality through social media because of a few reasons. First, language on social media has richer content that makes the typical linguistic analysis tool perform poorly. For example, Twitter, an online social networking service that enables users to send and read short 140-character messages called “tweets”, contains many Twitter specific language such as hashtag (#), at-mention (@), URL, and emoticons. People tend to use shorten version of phrases on Twitter, for example, “iono” means “I don’t know”. Twitter poses additional challenges due to the conversational nature of the text, the lack of conventional orthography, and 140-character limit of each message (tweet).

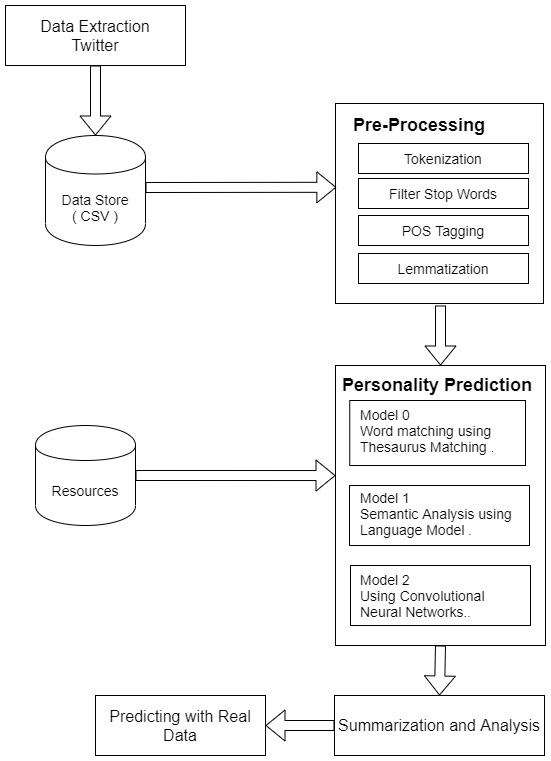
In this project, we try to solve the aforementioned problem by designing new richer linguistic analysis tools which can extract language feature in social media context and introducing a mechanism to automatically extract personality from text in social media. Using Twitter as a case study, we investigate the relationship between language features in tweets and personality traits, which leads to further experiment in predicting personality from language.

The Big Five personality traits is the model to be compared with the relationship between personality and behaviour.  The traits in the big five model are :-

1. **Openness to experience:**  People who are open to experience are intellectually curious, open to emotion, sensitive to beauty and willing to try new things.
2. **Conscientiousness:** Conscientiousness is a tendency to display self-discipline, act dutifully, and strive for achievement against measures or outside expectations.
3. **Extraversion:** Extraversion is characterized by breadth of activities (as opposed to depth), surgency from external activity/situations, and energy creation from external means.
4. **Agreeableness:** The agreeableness trait reflects individual differences in general concern for social harmony.

1. **Neuroticism:** Neuroticism is the tendency to experience negative emotions, such as anger, anxiety, or depression.

**2. DESIGN AND IMPLEMENTATION**

****

Our project is mainly divided into four parts.

Data Extraction, Text Processing, Personality Classification , Analyse and Store

**2.1 DATA EXTRACTION**

We are successfully extracting a persons latest (n number) of tweets.Twitter is a popular social network where users share messages called tweets. Twitter allows us to mine the data of any user using Twitter API or Tweepy. The data will be tweets extracted from the user. The first thing to do is get the consumer key, consumer secret, access key and access secret from twitter developer available easily for each user. These keys will help the API for authentication.

Tweepy is one of the library that should be installed using pip. Now in order to authorize our app to access Twitter on our behalf, we need to use the OAuth Interface. Tweepy provides the convenient Cursor interface to iterate through different types of objects. Twitter allows a maximum of 3200 tweets for extraction.

We had to register our twitter account as developer to retrieve tweets.

To access the Twitter API, we need 4 things from the Twitter App page. These keys are located in our Twitter app settings in the Keys and Tokens tab.

• consumer\_key = '\*\*\*\*\*\*\*\*'

• consumer\_secret = '\*\*\*\*\*\*\*\*'

• access\_token = '\*\*\*\*\*\*\*\*'

• access\_token\_secret = '\*\*\*\*\*\*\*\*'

We used tweepy module to extract the tweets. It return the tweets in form of tweepy.Cursor. We extracted the text from the cursor and stored in form a paragraph equivalent to the person's name in the Data File

**2.2 TEXT PROCESSING**

The data extracted from the Twitter contains too much symbols like @ and # and many additional symbols. Also the tweets contains too much inaccurate english words like laughinggggg etc.

**Tokenization**:

To handle the tweets string. First of all we tokenize the given string. During tokenization we remove punctuations , emojis ,URLs.

Then we filtered out the tokens that are of acceptable length (3) and remove stop words from the string.

**POS Tagging**:

This is used to assign parts of speech to each token in the string.

Ex : NN Noun, VRB Verb , RB Adverb etc.

This is mainly used in this program to perform lemmatization.

Lemmatization uses POS Tags to extract lemmas from the word.

We are using NLTK POS Tagger to perform this operation.

**Lemmatization**:

We needed to extract lemmas of each word to perform prediction. Lemma means the root word of a equivalent words.

Ex: lemma of the word kindness is kind .

We have used WordNetLemmatizer to extract lemmas of each word. It uses the word as well as its POS Tag to find correct lemma corresponding to it.

**2.3 PERSONALITY CLASSIFICATION**

This is the third and most important model of our project. In this we have to predict or calculate the peronality of a person using the the text we recieved.

We are going to use multiple methods to achieve best possible results.

**Model 0** : Thesaurus Based Equivalence

This is out first attempt in predicting the personality. This is totally thesauras based approach. In this we have created/generated dictionary based on each personality trait .Now we are using the processed lemmas to match and find the number of the matches In positives and negatives of each traits.

We are taking each token of each sentence and matching it in the dictionary of all the traits .We are predicting the polarity of each sentence toward the trait using this number. Now the polarity of each sentence computes the overall polarity of the person.

This approach is not suitable for texts not present in our dictionary.

We are getting 45% accuracy and 51% precision using this method.

**Model 1** : Similarity using Semantic Analysis ( SPACY )

This is our second attempt or improvement in personality prediction.

In this model , instead of matching the occurences we are computing the similarity of two words using the spacy similarity function.

Spacy similarity uses context sensitive tensors and word vectors to compute the similarity between two words.

The language model we used is "en\_core\_web\_lg" and "en\_vectors\_web\_lg"

Here lg means large dataset.

**Model 2** : Using Convolutional Neural Network ( CNN )

This is our third attempt for personality prediction.

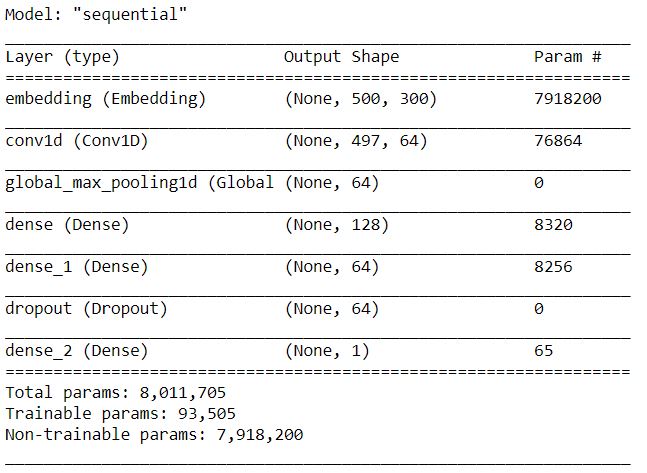
In this model we training the CNN model using 80 % of the essays data.

We are using pretrained embeddings weights from Google for embedding layer in Neural Network Model.

Embedding used from Google is " GoogleNews-vectors-negative300".

We trained five models one for each trait. Each of the model consist of five layers.

The summary of one of the network is given below:



All the 5 models have same layer configuration.

**2.4 ANALYSIS AND STORAGE:**

This is the final and non functional component of our project.

Here we will calculate the performace metrics like accuracy , precision etc of the document. Also we will store the results in csv file.

**2.5 USING MODEL TO PREDICT TEXTS:**

Using the models trained using essays dataset to predict manually given texts.

**3. RESOURCES REQUIRED**

Programming Language : **Python**

Tools / Libraries:

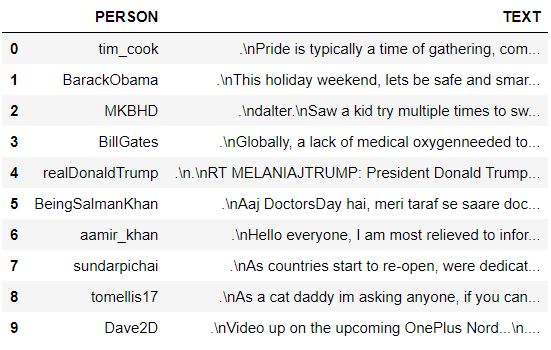
* Pandas : Conversion between CSV and DataFrame
* Re : Regular Expression Library
* TweePy : For Data Extraction from twitter
* NLTK : NLP Preprocessing Library
* spaCy : **spaCy** is a free open-source library for Natural Language Processing in Python. It features NER, POS tagging, dependency parsing, word vectors and more.
* Language Model : en\_core\_web\_lg , en\_vector\_web\_lg
* WordNet : It is a part of NLTK . It acts a dictionary for NLP.
* Gensim : It is used to access Google Word Embeddings word2vec. " GoogleNews-vectors-negative300"
* Keras and TensorFlow : It is used for creating, training and predicting neural network models.
* Sklearn : We used only one function of this library to generate classification report.

We tried POS Tagging with various other taggers like StanfordPOSTagger.

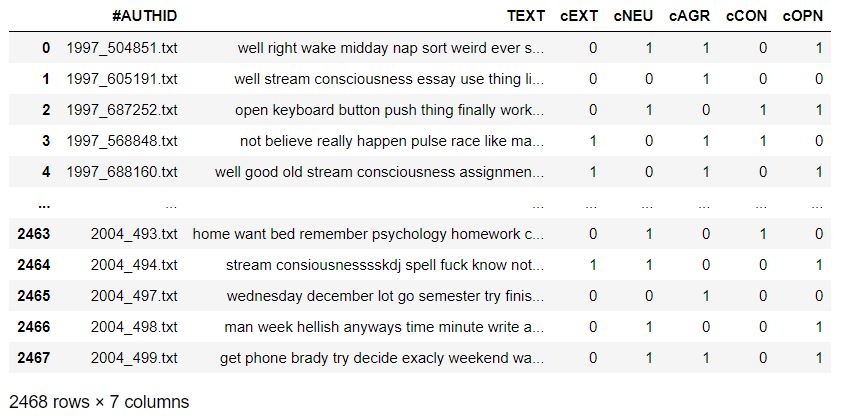
Stanford Tagger failed for large data set ( for token count > 10000). So we switched back to default NLTK POS Tagger which gives average result

**4. WORKING AND PERFORMANCE**

-> Data Extracted from **Twitter**

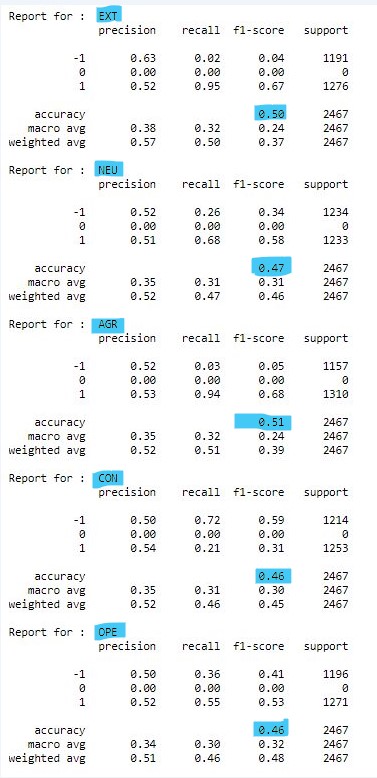
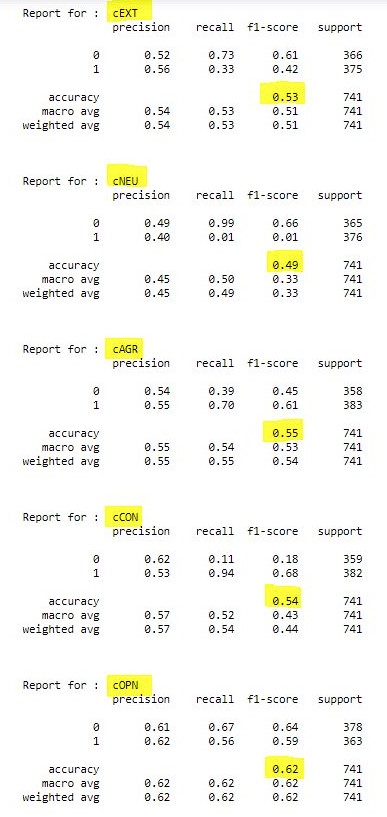


-> **Essays Dataset** for training

****

->> Computation Using **Thesaurus** On **Essays Dataset**.****

->> Performance Analysis

**** **Thesaurus Matching Convolutional Neural Network**

-> Using the trained CNN Model for **Twitter Extracted Data**

****

**5. LIMITATIONS**

* Multiple Languages for single data / Dataset
* Garbage Words (Words having no meaning such as ROFL, LOL)
* In social many words have repetition of letter like wwoooowwwww.
* Sarcasms
* Stanford POS Tagger not working for incorrect sentence. Halts whole program
* Model 1 did not worked well due to hardware limitations and errors is data. It failed for empty vectors.

**6. FUTURE PLANS**

This project can be taken further by using different models like Recurrent Neural Network and LSTM.