

Natural Language Processing (NLP - Final Project)

Date: 21st June, 2021

Submitted by:

ZOHA JALEEL 18i-0722

MUHID ABID 18i-0560

Table of Contents

**LOCATION BASED POLITICAL SUPPORT MONITORING3**

Abstract3

Introduction3

Related Work3

Goal4

Sentiment Analysis4

Prerequisites4

Datasets4

Process Description4

Getting Authentication Credentials4

Authenticating5

Creating Functions to Fetch Tweets5

Pre-processing of Dataset6

Clean6

Lemmatize6

Location7

Sentiment Analysis Function7

Positive8

Negative8

Neutral8

Saving Sentiment Count9

Visualization9

Evaluation10

Conclusion and Future Implementations10

# LOCATION BASED POLITICAL SUPPORT MONITORING

## Abstract

This paper describes how a model will use NLP techniques to analyze sentiments of people talking about a particular topic and categorize the results based on location.

We will see how this analysis will help political parties to get to know the opinions and sentiments of general public about them and how they change from time to time. A twitter based analysis will be used in gathering the data. This will basically include text and Social Media analysis by going through different posts and reviews uploaded on twitter by people.

## Introduction

Imran Khan, given his monumental rise in popularity, especially after his brief ‘dharna’ and becoming the Prime Minister of Pakistan. We thought it would be interesting to take a look at all that general public of Pakistan from different provinces have been saying about him and analyses their sentiments about Imran Khan and his workings after becoming Prime Minister.

Through this one would be able to make an educated guess about how much Imran Khan has managed to achieve during the past few years and how it has affected people’s opinion about him. Moreover, We also wanted to see what were the biggest and most notable events in the past few years that forced people to talk about him (be it in a positive or negative light).

Sentiment Analysis technique will be used to calculate the percentage of polarity in different provinces in accordance to their population ratio.

## Related work

“Brand24 is a [media monitoring](https://brand24.com/blog/what-is-media-monitoring-and-analysis/) tool (‘A comprehensive guide to media monitoring + TOOLS | Brand24 Blog’, 2020) with powerful analytics. It monitors the length and breadth of the Internet in search for the keywords you provide.” (‘What is Brand24? - FAQ | Brand24’, no date)

The original work ‘analyzes several surveys on consumer confidence and political opinion over the 2008 to 2009 period, and finds they correlate to sentiment word frequencies in contemporaneous Twitter messages.’ (O’Connor *et al.*, 2010)

## Goal

The original aim was to extract the views of the Pakistani public using data from tweets. We plan to do a real time sentiment analysis of tweets about Imran Khan and to categorize them based on geo tag.

## Sentiment Analysis

It is a process that predicts whether a certain sentiment has a positive, negative or neutral sentiment to it (*Everything There Is to Know about Sentiment Analysis*, no date). It is the most common text classification tool that analyses an incoming message or text and returns the polarity and sentiment about content.

Deployment in the social media space to discover how people feel about certain topics, particularly through users’ word-of-mouth in textual posts, or in the context of Twitter, their tweets is a common use of this technology.

## Prerequisites

The programming language we used in implementing the algorithm is **python.**

## Datasets

To generate the datasets automatically from twitter, we will use ‘Tweepy’ which is a Python Library that will allow us to get data from the Twitter API. We plan to stream tweets and plot Graphs in real time since twitter does not allow scraping of historic data without commercial Tools. If we fail to extract data through Tweepy, an alternative would be to get the URLs, clean them and save the downloadable Twitter Analytics data into ‘Panda data frames’. This method can be done with R’s package but we still have to look into this method and hope we won’t have to resort to it.

## Process Description

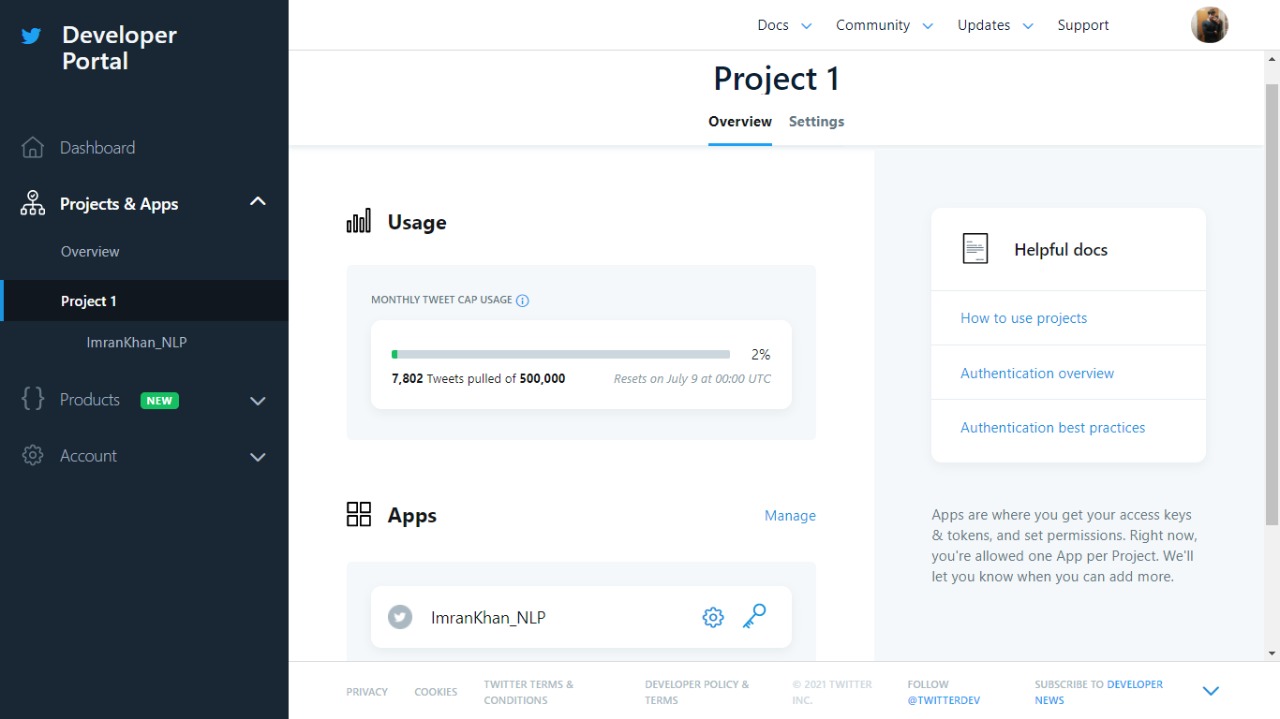
The task of sentiment analysis focuses in textual data which requires a lot of processing. For that we chose to register on twitter (Al-Masri, 2020).

## Getting Authentication credentials

Firstly, we visited twitter developer website and logged into our account. We applied and agreed to all the terms and conditions and after few days the application got approved. From there we got the keys and tokens as well as the API secret key. There were four keys in total. Consumer key, Consumer Secret, Access key, Access token Secret and Bearer token. All the four tweets are shown once but can be regenerated.

## Authenticating

Since we got our Twitter Developers login credentials (i.e. API keys and Access token), we proceeded to authenticating our program. First, we import the Tweepy library, then create an Twitter. API object with the credentials from the “safe” place.



## Creating Functions to fetch tweets

Basically, this is going to be a function that takes a search keyword (i.e. string) as an input, searches for tweets that include this keyword and returns them as twitter. Status objects that we can iterate through. A set rule function takes the value, tag, keyword and language and return the tweets in accordance to them. All the tweets are based on these rules. We specified our tweets to be in English language and related to politics. We used JSON library to fetch the tweet text. We sometimes get **ChunkedEncodingError** in functions because this is a twitter API issue being faced by multiple developers and is currently being worked on.

Keyword and value we entered was ‘Imran Khan with a tag of ‘politics’. By this we only fetched the tweets that were relevant for us.

## Pre-Processing of Data Set

Many tweets that we get had words that were irrelevant including emoji, URLs, hyperlinks, mentions and punctuation etc. These does not contribute in sentiment analysis so it was better to get rid of them. We just need the data that will help us in calculating the polarity of the tweets. Our next step was to pre-process our data which involves:

* Cleaning the tweets
* Lemmatization

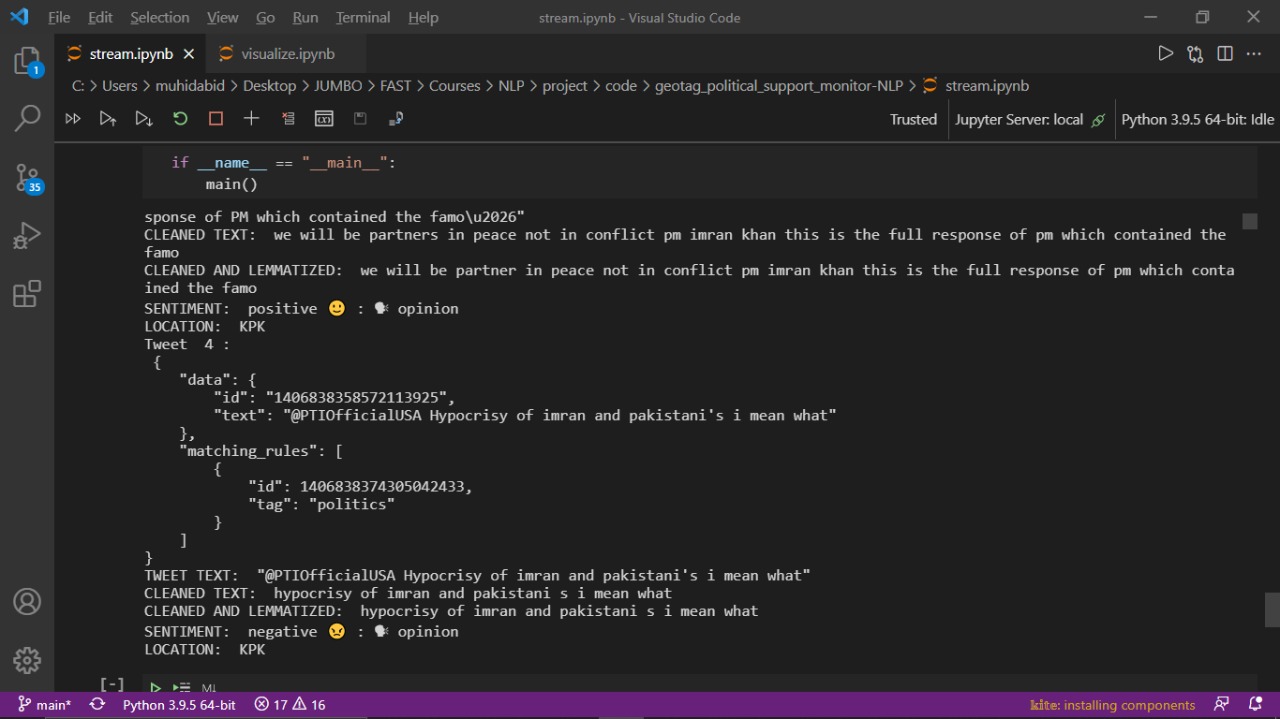
## Clean

A function was created which takes the tweet text and remove all kinds of emojis, hyperlinks, URLs, Unicode, \n and punctuation. For this we specifically used **regex**. In case of hashtags, the hash is removed and the text part is divided into component to make it a part of sentence. This was also very useful while measuring polarity. Other than that all the upper case letters were change to lower case.

The function snippet is displayed below:

## Lemmatize

It was necessary to get the root form of the inflected words. For that we did lemmatization on our fetched tweets. NLTK library was imported for this purpose. The reason we didn’t use stemming is because stemming sometimes change the actual word which has no meaning or any kind of context. While lemmatization on the other hand make sure that text is still in its context (*Stemming and Lemmatization in Python*, 2018).



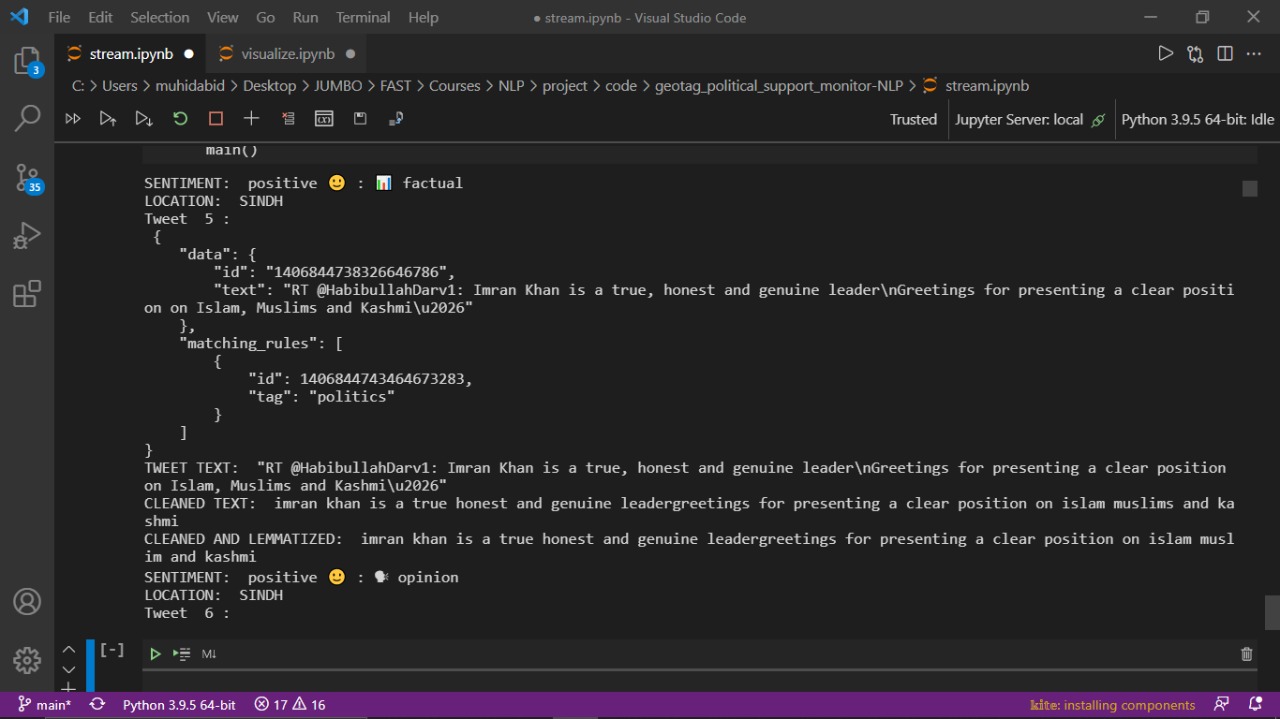
## Location

We couldn’t fetch geo tags because we got approved for a 'standard' twitter developer team account and not for an 'academic research' account. The best alternative we came up with was to create a location function that assigns provinces to tweets based on the probability in accordance to population ratio. The population data was taken from Pakistan Bureau of Statistics (*Final Results Census-2017 | Pakistan Bureau of Statistics*, no date)

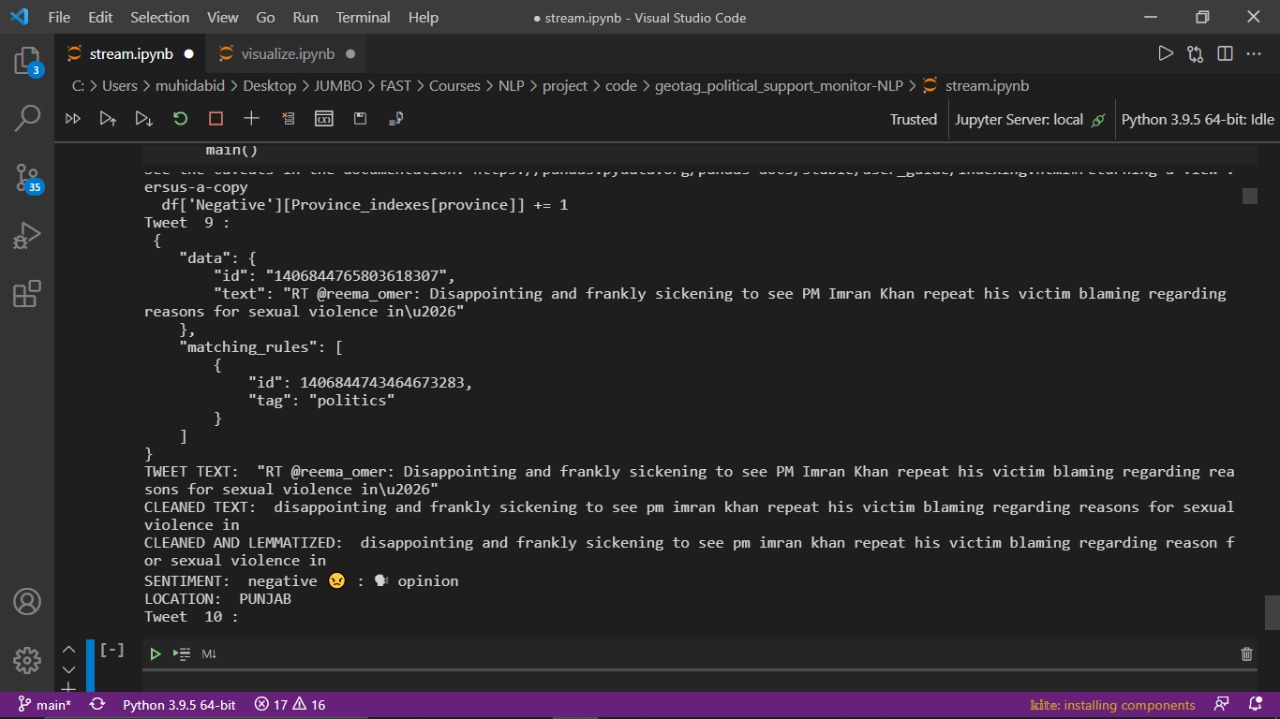
## Sentiment Analysis Function

For analyzing the sentiments we used the library TextBlob. Textblob is an open-source, rule based python library for processing textual data (Sharma, 2020). The function is created in which text and location is passed. It analyses the tweet and calculates the polarity. If the polarity is greater than zero than the sentiment is positive. In case of negative value, it means the sentiment is negative and on equal to zero it returns neutral sentiment. It also returns whether the tweet is an opinion or any factual statement.

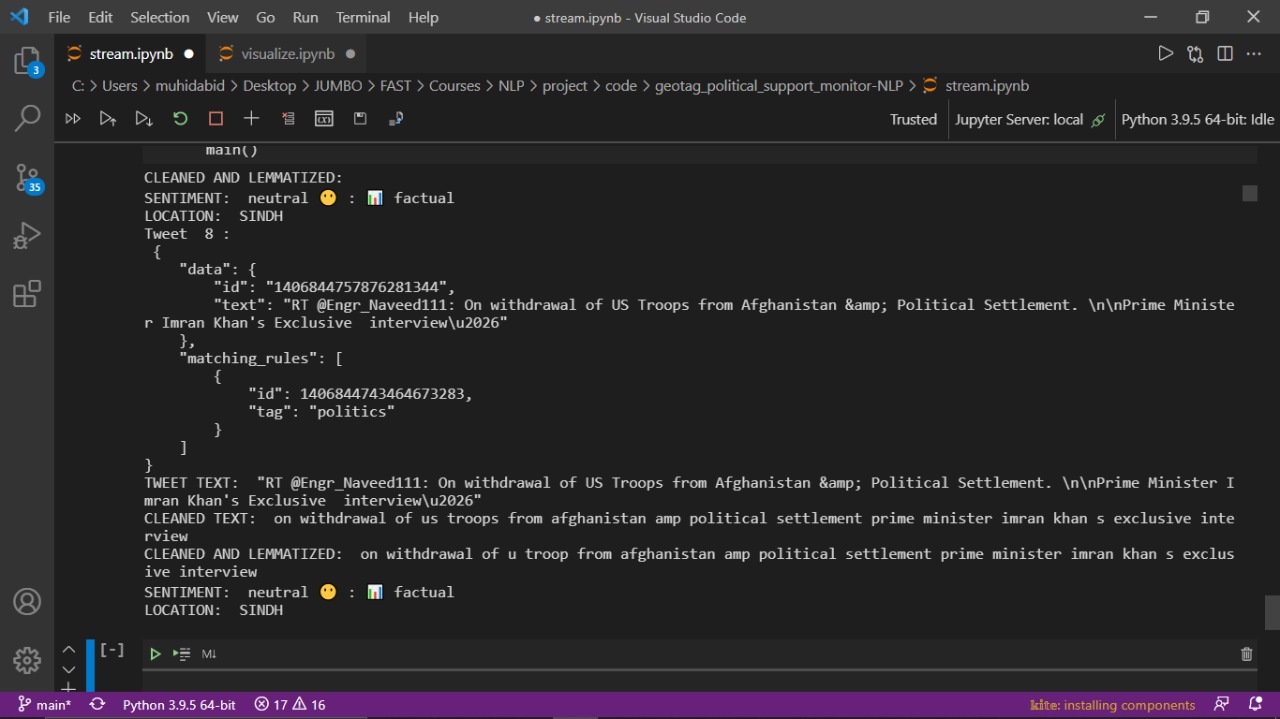
## Positive:

****

## Negative:

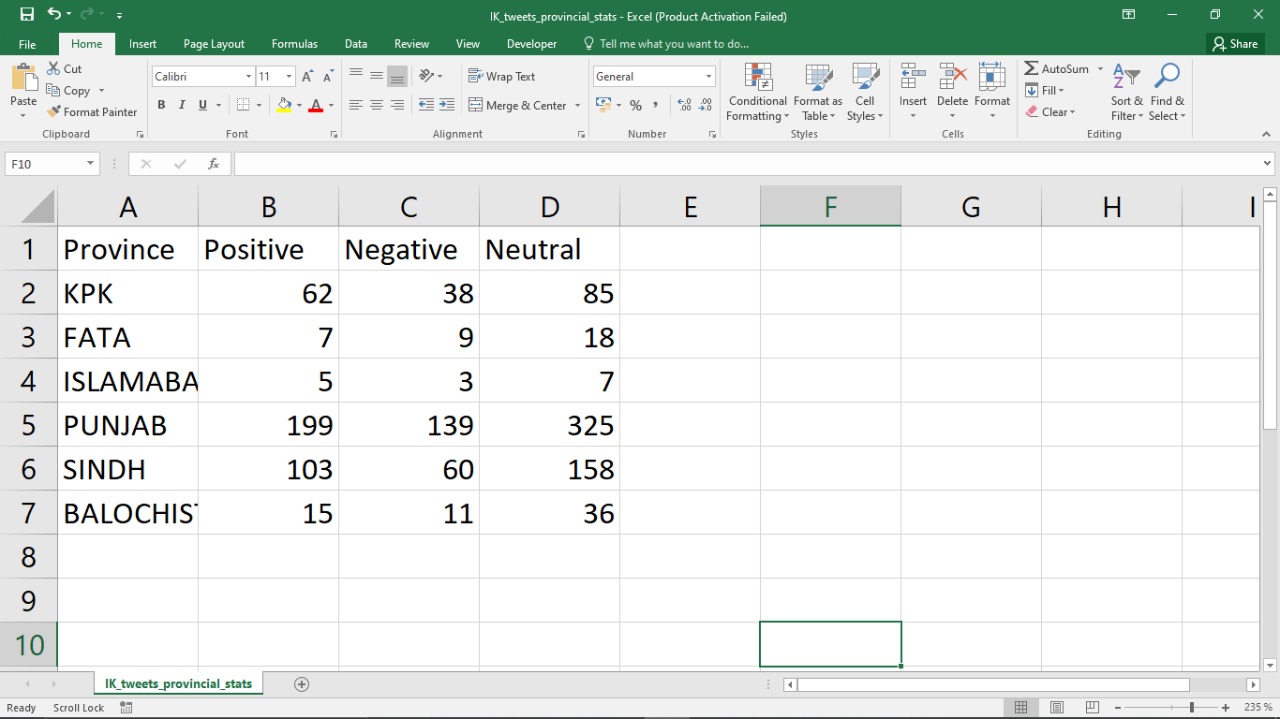


## Neutral:

****

## Saving sentiment count

After fetching all the tweets, cleaning them, assignment of locations sentiment analysis function is applied which gives polarity of every tweet and mark them as positive, negative or neutral. All the data and statistics are saved in a csv file in column every time. As the tweet is fetched, the whole process repeats.



## Visualization

We plotted positive, negative and neutral sentiments for all of the 6 locations. This is in the form of bars which updates as the csv file updates. Tweets are saved in a text file. And are also displayed with the plotted graph.



# Evaluation

Since, we are not using a machine learning approach, we cannot test our model on a test set. Once we have access to real geolocations we can compare our results with the result of last provincial elections (*Pakistan Election 2018: Results, Party Position, Candidates, News, Live Updates*, no date)

# Conclusion and Future Implementations

This model can be very helpful for political parties in Pakistan to monitor support and if a larger historical data is provided, political parties can predict votes in different regions in upcoming elections. Through this model, Imran khan supporters are predicted very well and their sentiments are analyzed really well about his government.

# Bibliography

‘A comprehensive guide to media monitoring + TOOLS | Brand24 Blog’ (2020), 15 April. Available at: https://brand24.com/blog/what-is-media-monitoring-and-analysis/ (Accessed: 21 June 2021).

Al-Masri, A. (2020) *Creating The Twitter Sentiment Analysis Program in Python with Naive Bayes Classification*, *Medium*. Available at: https://towardsdatascience.com/creating-the-twitter-sentiment-analysis-program-in-python-with-naive-bayes-classification-672e5589a7ed (Accessed: 21 June 2021).

*Everything There Is to Know about Sentiment Analysis* (no date) *MonkeyLearn*. Available at: https://monkeylearn.com/sentiment-analysis/ (Accessed: 21 June 2021).

*Final Results Census-2017 | Pakistan Bureau of Statistics* (no date). Available at: https://www.pbs.gov.pk/content/final-results-census-2017 (Accessed: 21 June 2021).

O’Connor, B. *et al.* (2010) ‘From tweets to polls: Linking text sentiment to public opinion time series’, in *Proceedings of the International AAAI Conference on Web and Social Media*.

*Pakistan Election 2018: Results, Party Position, Candidates, News, Live Updates* (no date). Available at: https://www.geo.tv/election (Accessed: 21 June 2021).

Sharma, H. (2020) ‘Let’s Learn TextBlob Quickstart - A Python Library For Processing Textual Data’, *Analytics India Magazine*, 17 August. Available at: https://analyticsindiamag.com/lets-learn-textblob-quickstart-a-python-library-for-processing-textual-data/ (Accessed: 21 June 2021).

*Stemming and Lemmatization in Python* (2018) *DataCamp Community*. Available at: https://www.datacamp.com/community/tutorials/stemming-lemmatization-python (Accessed: 21 June 2021).

‘What is Brand24? - FAQ | Brand24’ (no date). Available at: https://brand24.com/what-is-brand24/ (Accessed: 21 June 2021).