# CSCM35: Big Data and Data Mining
# Coursework 2

Andy Gray
445348

01/05/20

# 1 Introduction

We have presented upon us a challenge to complete a practical data mining task, which involves a technical report and a software solution. We have a dataset provided that has been collect by Johns Hopkins CSSE in real-time, with values related to the current pandemic the coronavirus (COVID-19). However, we need to use additional datasets to complete complement the provided data. Once identifying a research question, we need to develop a prototype and evaluate our method.

With the COVID-19 first declaring over 800 cases on the 23rd of January 2020, and figures still rising, which at point of writing was 3.12m cases and 217K deaths worldwide [**?**], the virus has reached over 199 different countries. We wanted to see if between the time scale of the 22nd of January and the 25th of April, what is the general public sentiment is. We want to know, has the general public sentiment changed, or if it has stayed the same? While over the COVID-19 pandemic.

We will first achieve this by using Twitters API to gain all the tweets that have been tweeted, over this time, using the hashtag 'coronavirus'. We will then complete a sentiment analysis on the data, to gain an overall view of the general feel of the 'coronavirus' tweets. Gaining the sentiment value will allow us to have insights into the public view, and if these tweets are positive, neutral or negative and how they change as the pandemic plays out over time.

[Overview of the results here]

[Overview of what to expect in the document]

# 2 Proposed Solution

We will be using twarc, a Python library for archiving Twitter JSON data. The library accesses the Twitter API to extract a JSON object as Twitter stores tweets as line-oriented JSON. The twarc library handles the Twitters's API, as well, as its rate limits. This library also allows us to use hydrate tweet ids. It is collecting the relevant tweet's details that match the criteria of having hashtags of 'coronavirus', 'virus', 'covid'.

We will then use another python library called hydrate. Hydrate [explain it]. Once the tweets get collected for all of the Tweet IDs, we will then convert the JSON data into a CSV file. We will then do sentiment analysis on the tweets to gauge the over feel of these tweets. To see if they are all negative, neutral or if they overall a positive vibe over each day.

[Need to explain the stages after this as this is where I am in the practicle]

We will then use NLTK to create a sentiment analysis on the tweets text body. [Explain what we did in stages]

We will then use the ARIMA algorithm to predict where we think the general mood about the COVID-19 on twitter is heading, more into a negative direction, or is it neutral or moving into a positive direction.

## 2.1 Packages

We will be using the programming language Python 3 [10], as this allows us to use all the required additional packages needed. The additional packages we will be using are NumPy [13], OpenCV [2], Pptk, Matplotlib [4], SKImages [12], SKLearn [3] and Tensorflow version 2 [1].

## 2.2 Dataset

The presented dataset provided, which is maintained by Johns Hopkins CSSE, consists of [decsribe data here]. The data had many missing values, which correlate to the country not having any COVID-19 cases, so these features got filled in with zeros.

With using Twarc, we were able to gain 15million tweets.

## 2.3 Sentiment Analysis

Sentiment analysis is a type of data mining. It aims to measure what the inclination of people's opinions is, through using natural language processing (NLP). NLP is a computational linguistics and text analysis. It gets used to extract and analyse information from the Web, which is mostly social media and other similar sources. The analysed data quantifies the general public's sentiments or reactions toward certain situations, products, people or ideas and reveal the contextual polarity of the information. Sentiment analysis is also known as opinion mining [**?**].

Sentiment analysis can fall into two categories, pure statistics or a mix of statistics and linguistics [**?**]. Pure statistics use algorithms like the Bag of Words (BOW). This kind of algorithms filters the text down to only the words that the algorithm believes to have sentiment, taking into account no context to the sentence at all. Such models do not aim to understand the language, only analyse the statistical measures to classify the text. [Will I use this approach?]

The mix of statistics and linguistics approach uses an array of Natural Language Processing (NLP) techniques, along with statistics to allow the machine to understand the language truly. The algorithms achieve this by incorporating languages grammar principles into analyst of the text.

There are broadly two main outputs to sentiment analysis. One type of sentiment analysis output gets referred to as Categorical/Polarity. What this means is that the text will get classed as either positive, negative or neutral overall. While the other is Scalar/Degree. What this means is that a score is given based on a predefined scale, that ranges from highly positive to highly negative. This type of sentiment analysis output has been used on tweets to see the views on various USA election candidates.

### 2.3.1 NLTK

### 2.3.2 ARIMA

## 2.4 Analysis Tools

# 3 Results

# 4 Discussion

# 5 Conclusion

# References

[1] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] Bradski, G. The OpenCV Library. *Dr. Dobb's Journal of Software Tools* (2000).

[3] Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., VanderPlas, J., Joly, A., Holt, B., and Varoquaux, G. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning* (2013), pp. 108–122.

[4] Hunter, J. D. Matplotlib: A 2d graphics environment. *Computing in science & engineering 9*, 3 (2007), 90–95.

[5] Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (2012), pp. 1097–1105.

[6] LeCun, Y., Bengio, Y., and Hinton, G. Deep learning. *nature 521*, 7553 (2015), 436–444.

[7] LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. Backpropagation applied to handwritten zip code recognition. *Neural computation 1*, 4 (1989), 541–551.

[8] LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE 86*, 11 (1998), 2278–2324.

[9] LeCun, Y., et al. Generalization and network design strategies. *Connectionism in perspective 19* (1989), 143–155.

[10] Python Core Team. *Python: A dynamic, open source programming language.* Python Software Foundation, Vienna, Austria, 2020.

[11] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. Learning internal representations by error propagation. Tech. rep., California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[12] Van der Walt, S., Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., Gouillart, E., and Yu, T. scikit-image: image processing in python. *PeerJ 2* (2014), e453.

[13] Walt, S. v. d., Colbert, S. C., and Varoquaux, G. The numpy array: a structure for efficient numerical computation. *Computing in Science & Engineering 13*, 2 (2011), 22–30.