Sentiment Analysis of Twitter Discourse in the Context of UN Climate Conferences

James Hughes, Ameer A Saleem, Vignesh Sridhar, Holly Taylor.
Supervised by Janique Krasnowska and Fernando A Zepeda

17 March 2023

Abstract

The following document entails a student-led end-to-end data science project, the goal of which being to investigate the changes in perceptions towards climate change over time. The project involved a review of the current literature and available data, sentiment analysis, neural network model training and evaluation of the developed model, as well as projections for the future.

Contents

1 Introduction

We discussed many different avenues for research and project focus initially. Below are detailed a few of these ideas and their resolutions:

- Analyse pledges made by political leaders and conduct some data-backed investigation into how much they keep to their word.
  Pros: Plenty of data available, can be investigated in many different scopes (e.g. national vs global, a few select politics vs advocates for a particular party, etc.).
  Cons: The language used by such politicians is constrained, which could make it difficult to train a model. Also, it would be difficult to associate a metric with keeping promises, as this could be subjective and it is unlikely that a database encapsulating this with objective labels exists. The scope of individuals to investigate is also somewhat constrained, regardless of the aforementioned scope.

- Investigate the changes of attitudes of the public towards climate change following different climate summits.
  Pros: A genuinely interesting question, the implications of which could be used to forecast how attention on climate change could evolve through the years.
  Cons: Very ambitious scope for a student-led project- the title needs refined in order for the problem to be well-defined.

After considering the benefits and drawbacks of each of the project ideas we eventually settled on the key questions outlined below.

Twitter is a widely-used social media platform which can be used to sample the public's reaction to international events. Our study used historical data of Tweets occurring across the courses of multiple COP conferences from COP24 to COP26. We used a human-labelled dataset [2] to construct a model to classify the sentiment of a Tweet, limited to *negative*, *neutral*, or *positive*, which we then used to label our original dataset. Additionally, we manually categorised the Twitter accounts in our original data - limited only to the top 100 users ordered by the maximum likes of any of their tweets in the dataset. Our analysis was then centered around the following key questions:

1. Who are the most influential stakeholders in the Twitter discourse that surrounds the COP events?

2. Is there a relationship between the sentiment of a tweet and how much exposure it receives?

3. How does this relationship vary across types of users and over time?

From here, we conducted a literature review to ascertain which of these avenues would be the most fruitful to pursue.

2 Literature review and pre-requisites

2.1 Related Work

[Insert information about the literature on the NLP Teams Channel]

2.2 Scraping Tweets

The literature suggested that online tweets would prove useful as an abundant data source for the project. While Twitter has an API for scraping tweets, we settled on the use of snscrape, which appeared easier to implement.
Twitter is a prevalent platform with immense usage, meaning that scraping within an arbitrary timeframe will yield a lot of irrelevant data. It makes reasonable sense to focus on time intervals before, during and after COP summits; this is the time when most 'notable figures' would be discussing climate change and voicing their respective opinions on it the most. This also extends to the public, providing us a greater density of climate-related data. Since Twitter does not allow making the text of tweets public, any Twitter-related dataset must undergo a process called hydration.

2.3 COP Conferences

COP, standing for 'Conference of the Parties', is the main decision-making body regarding any major decisions regarding climate change. COP meetings are organised by the United Nations and involve almost every country in the world. The goal of such summits is to discuss what major governmental authorities can achieve in the fight against climate change. COP meetings usually take place every year and aim to review the global progress made in mitigating the impact of climate change. The first COP summit took place in Berlin in March 1995. At the time of writing, the most recent COP summit, COP 27, took place in Egypt in November 2022.

2.4 Word Embeddings

In order to get a computer to understand and interpret the words in each tweet is to first convert them into vectors in higher dimensional space. The idea is to give words with similar meaning/context a similar vector value- for example, we may want to give the words 'lion' and 'leopard' similar vector representations in a bank of words containing the names of all the animals in the world. There are a few choices as to how this can be achieved, such as Word2Vec or GloVe's pre-trained word embeddings. We went with the latter for this project. The two methods of word embedding would most likely yield very similar results for us, though they work in different ways. Word2Vec utilises a feedforward neural network. GloVe uses a matrix-based approach: it starts by constructing a matrix whose rows are words and columns are contexts it has identified in the input word bank. This matrix can then be decomposed into smaller matrices that are easier to work with (e.g. from word x context to word x feature and feature x context). We opted with GloVe for this project, although the choice didn't matter much.

This begs the question of this technique's utility. Once the words of a dataset have been word embedded, we can implement the clustering algorithm to have the computer try and identify patterns in the words for us. For the sake of this project, we investigated clustering with different numbers of clusters in the EDA portion of the project.

We encountered another potential limitation when manually categorising the data in our scraped tweets. A lot of tweets were indeed related to climate change, but sometimes they discussed an individual rather than the topic of climate change itself, for example praising Greta Thunberg for her efforts in this space. This is not directly related to our goal of attributing sentiment labels to these tweets *with regards to stance on climate change.*

2.5 Clustering techniques

We briefly outline a few different clustering techniques below:

1. Centroid-based clustering: these types of algorithm work by assigning the optimal location for the center of each cluster, known as the *cluster centroid*). The position of each centroid is updated by finding the average of the vectors that belong to that cluster, terminating once a pre-specified tolerance has been reached. The K-Means algorithm is the most popular centroid-based clustering technique.

2. Hierarchical clustering: this algorithm creates a tree of clusters, with a larger number of clusters corresponding to additional branches in these trees. As the name suggests, this method of clustering is best suited to hierarchical data.

3. Density-based clustering: this algorithm creates clusters based on the density of points, allowing for varying shapes and sizes of clusters. However, this form of clustering does not generalise well into higher dimensions.

4. Distribution-based clustering: this approach assumes that the data follows a probability distribution. The closer the data points are to the centre of the cluster, the higher the probability they belong to that individual cluster. This method of clustering ought not to be used if the underlying data distribution is unknown (or doesn't exist).

Search Term	Date Range	Number of Tweets
COP24	15-Nov-2018 to 30-Dec-2018	3551
COP25	15-Nov-2019 to 27-Dec-2019	2560
COP26	14-Oct-2021 to 27-Nov-2021	3489

Table 1: The nine user categories and their descriptions.

3 Data

3.1 Main Twitter Datasets

We used the python library *snscrape* to access historical tweet data, with the queries detailed in Table 1. For the datasets relating to each conference we limited the query to English Tweets containing the corresponding string of the form "COP–", which is case insensitive and includes tweets with the substring "#COP24", for instance. Each tweet was collected along with important meta-data such as information about the author account, and metrics such as number of likes and replies.

Next we compiled the users authoring tweets among the most liked tweets in the dataset, and manually categorised them according to nine stakeholder categories show in Table 2. We then filtered each of the original datasets to contain only tweets from these categorised users. Lastly, we added the sentiment predictions from our trained model.

3.2 Implementation

We began by focusing our attention on dates close to/during COP summits. We scraped tweets containing the string 'COP–' within time frames of a few weeks at a time during periods of high COP discussion and activity, yielding tens of thousands of tweets in each scrape. Our goal at this stage was to explore patterns at both tweet and word level.

We started by accumulating all the tweet data in each scrape and clustering at the word level, with both K-means clustering and agglomerative clustering. The most interesting insights were as follows:

- The most frequently used words in each scrape were what are called *stop words*. In the context of NLP, these are words that carry very little information. Some examples are shown in Figures 1a and 1b. This was something we would have to take care of during data preprocessing. Note that we didn't remove phrases such as "#COP..." since their abundance in the data was a consequence of our filtering criteria.

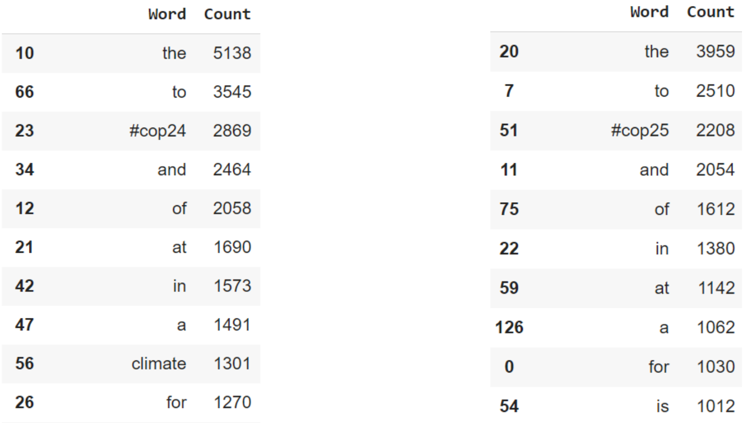
- For K=__, the K-Means clustering algorithm was able to independently group together tweets pertaining to different categories of *people*, i.e.

Category	Description	Examples
Activist	Accounts belonging to activists or charitable organisations working to help climate change mitigation or adaption.	@GretaThunberg @SumakHelena
Business	Accounts related to a business or personal accounts of business leaders.	@BNPParibas @BoschGlobal
Celebrity	Personal accounts for which no other category applies and with more than 10,000 followers.	@LeoDiCaprio
International Organisation	Official accounts for non-commercial international organisations.	@GreenpeaceNZ @UNFCCC
Journalist	Personal accounts of news reporters and commentators.	@katieworth @LeoHickman
News	Official accounts for large news and media outlets, including digital news websites.	@Reuters @AJEnglish
Politician	Official accounts for individuals in (or formerly in) government positions or international political organisations.	@NicolaSturgeon @Justin Trudeau
Scientist	Accounts related to research organisations or personal accounts for people whose primary occupation is research or public science discourse.	@Astro_Alex @Peters_Glen
Misc	None of the categories above apply.	@Damo_Mullen @mamaloe66

Table 2: The nine user categories and their descriptions.

separate clusters related to politics, climate change activism and environmental science. This gave us the idea to categorise the tweets by the occupation of the user (or their relation to the topic of climate change).

- The dendrograms from the agglomerative clustering showed that...



	Word	Count
10	the	5138
66	to	3545
23	#cop24	2869
34	and	2464
12	of	2058
21	at	1690
42	in	1573
47	a	1491
56	climate	1301
26	for	1270

(a) COP24

	Word	Count
20	the	3959
7	to	2510
51	#cop25	2208
11	and	2054
75	of	1612
22	in	1380
59	at	1142
126	a	1062
0	for	1030
54	is	1012

(b) COP25

Figure 1: The top 10 most common words in the COP24 and COP25 tweet scrapes

3.3 Sentiment Analysis

We implemented a neural network model in Tensorflow to learn the patterns of label associations with the tweets.

In order to train the model, we required a dataset with sentiment labels attributed to each tweet. This could be done by a computer with clustering, but defeats the purpose of using the dataset for training, since we would have no feasible way of validating so many rows of labelled data. There exists a Climate Change Twitter Dataset [1] that, among many other things, attributes to each tweet a value in the interval $[-1, 1]$, with 1 denoting positive sentiment, -1 denoting negative sentiment and 0 indicating a neutral stance on the topic. However, this dataset is machine-labelled and we didn't like the idea of training our neural network on data that was labelled by another machine. Moreover, upon closer inspection of some of the attributed labels, we found some values that we disagreed with, e.g. instances when a tweet was labelled with a 0 when we thought it should've been labelled with a 1.

We did a bit more searching and found the Twitter roBERTa-base for Sentiment Analysis [2] consisting of 1.6 million tweets across 15 different languages. We started with just the tweets written in English, with the project potentially extending to tweets in other languages if time allowed. Note that the tweets in these datasets also require hydration to use. In contrast to the aforementioned dataset, this dataset uses the labels $\{0, 1, 2\}$ for negative, neutral and positive sentiment respectively, which is the methodology we chose when applying our model to the data.

We constructed and trained five neural network models using TensorFlow on the English tweets sentiment dataset. The testing and training accuracies are shown in Figure 3. From this, we opted with our first model, consisting of architecture shown in Figure 4. Our final model, the simplest in the set of five, is comprised of the following layers:

- **Embedding layer**: similar to the GloVe package, this Keras layer takes our set of words and embeds them as vectors in n-space. We have specified the input dimension size (the total number of words in our dataset) and the output dimension size (the dimension of space we want to embed our word vectors into). After some experimentation, we found the output space dimension of 16 to be a good fit.

- **Pooling layer**: pooling methods aim to reduce the size of the input space by replacing groups of points with single points formed by aggregating the groups' values. The two most common types of pooling methods: max pooling, in which the maximum value of a group of points is taken as the value of the new aggregated point; and average pooling, where the average value is taken instead. In general, max pooling is able to highlight stark contrasts in a dataset, whereas average pooling is better for 'smooothening out' the values in the entirety of the dataset. We opted with average pooling for our chosen model.

The other four models consisted of the above layers as well as the following:

- **Convolution layer**: a convolution is a filter applied to an input, with multiple filters constituting a feature map. Convolution layers are often used in tandem with pooling layers: convolutions create multiple feature channels in parallel; and the pooling layer reduces the dimension of each of these channels.

- **Dropout layer**: this layer randomly sets activation values to 0, temporarily 'dropping out' these nodes from the network. The main purpose of dropout layers is to help prevent overfitting. The utilisation of this kind of layer is especially useful when training a large neural network on a relatively small dataset.

Our validation dataset consisted of 41.0% positive tweets, 43.5% neutral tweets and 15.6% negative tweets. If the model were to class every single tweet as neutral, it would achieve a validation set accuracy of 43.5% (such

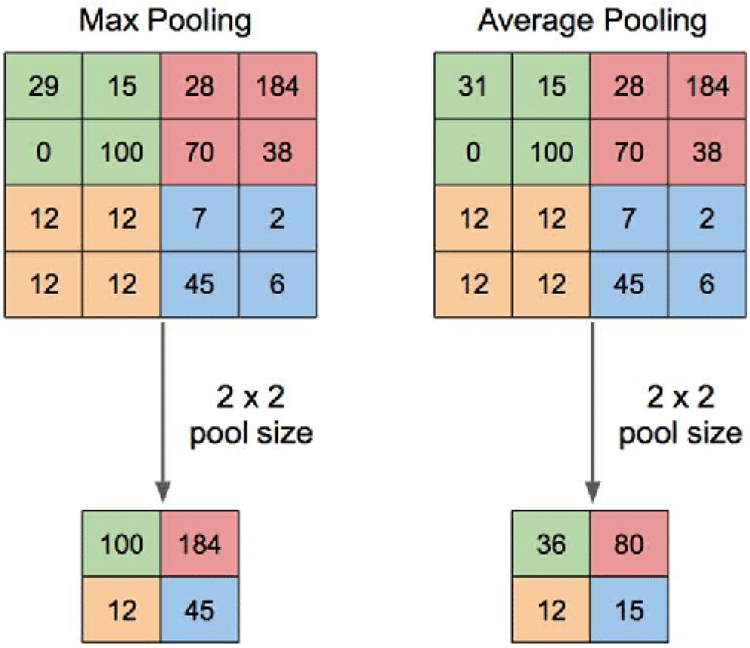


Figure 2: Illustrations of common pooling methods. [Source: ResearchGate.]

pigeon-holing would certainly be possible due to the similar class imbalances in the training dataset). Hence, we decided on a baselines accuracy of 45% for comparison with the accuracy of each of our 5 models.

The unfiltered confusion matrix for our final model is shown in Figure 5. The confusion matrix for a model with 100% accuracy would be a diagonal matrix, for example. Our model outputs a vector $\{p_1, p_2, p_3\}$ with $p_1 + p_2 + p_3 = 1$, analagous to the network's confidence that the input tweet belongs to each of the three classes. We then return the index of the largest p_i, so for example if the output for a particular tweet is $\{0.1, 0.4, 0.5\}$, then we take the sentiment label to be 0 (i.e. negative. This gives us two different ways of evaluating our model: restricting our view to the final decision made by the model, due to selecting the index of the node with largest p_i; or by filtering the results based on the network's confidence in its classification. The confusion matrix for the model's classifications with confidence greater than 80% is shown in Figure 6, i.e. for these 632 tweets, $\max_i p_i > 0.8$. This added filter cuts out 68.4% of the tweets initially in the validation set.

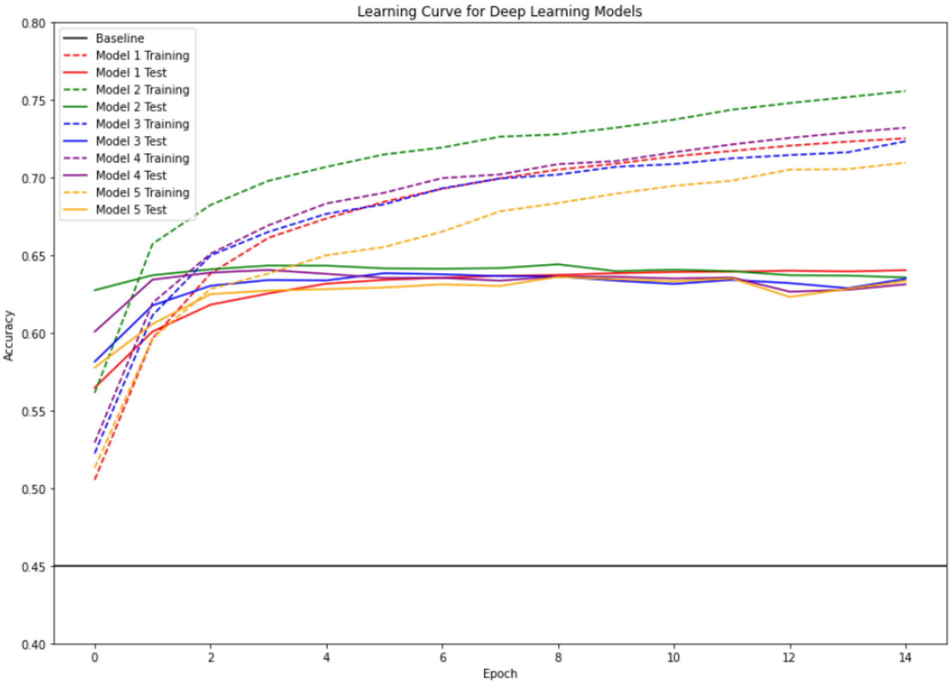
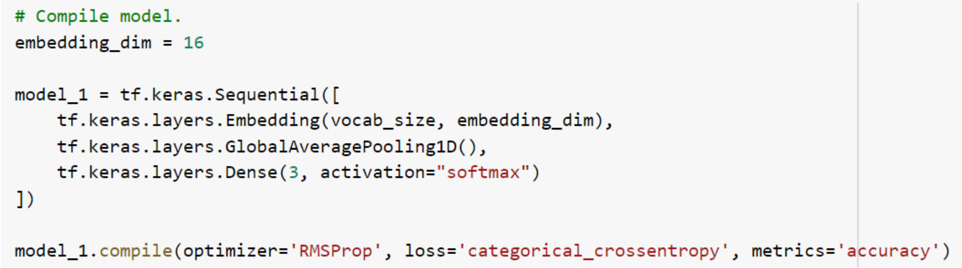


Figure 3: A plot of the training and testing accuracies of the five neural network models.

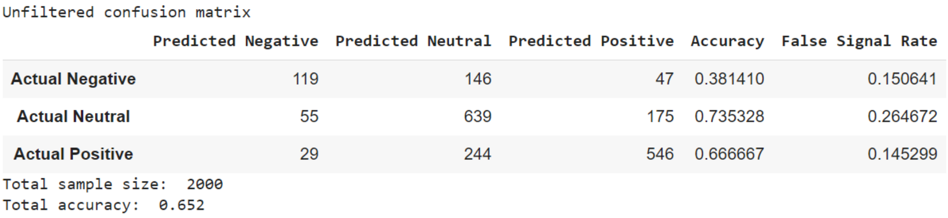


```
# Compile model.
embedding_dim = 16

model_1 = tf.keras.Sequential([
    tf.keras.layers.Embedding(vocab_size, embedding_dim),
    tf.keras.layers.GlobalAveragePooling1D(),
    tf.keras.layers.Dense(3, activation="softmax")
])

model_1.compile(optimizer='RMSProp', loss='categorical_crossentropy', metrics='accuracy')
```

Figure 4: Model 1 architecure.

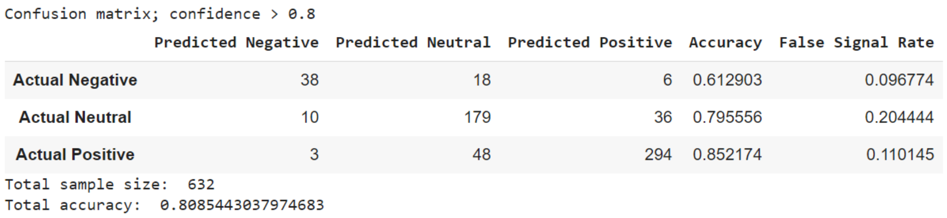


```
Unfiltered confusion matrix
```

	Predicted Negative	Predicted Neutral	Predicted Positive	Accuracy	False Signal Rate
Actual Negative	119	146	47	0.381410	0.150641
Actual Neutral	55	639	175	0.735328	0.264672
Actual Positive	29	244	546	0.666667	0.145299

```
Total sample size:  2000
Total accuracy:  0.652
```

Figure 5: Unfiltered confusion matrix



```
Confusion matrix; confidence > 0.8
```

	Predicted Negative	Predicted Neutral	Predicted Positive	Accuracy	False Signal Rate
Actual Negative	38	18	6	0.612903	0.096774
Actual Neutral	10	179	36	0.795556	0.204444
Actual Positive	3	48	294	0.852174	0.110145

```
Total sample size:  632
Total accuracy:  0.8085443037974683
```

Figure 6: Confusion matrix showing tweets classified with at least 80% accuracy.

4 Conclusion

4.1 Ambiguity of Sentiment

Suppose that one of the tweets in our scrape was the following: “ ‘I’m so happy that everyone is working so hard to solve the issue of climate change’, said no-one ever.”.[1] The tweet indicates a negative sentiment with regards to climate change. However, the use of the word ‘happy’ could potentially cause our model to classify this as a tweet with positive sentiment. Not only would this tweet reduce the accuracy of our network, but the tweet itself could be argued as inherently destructive to the model training process; it may suggest to the network that the word ‘happy’ ought to be synonymised with negative sentiment rather than positive (which it ought to in isolated context). Therein lies an inherent drawback in our pipeline- sarcastic tweets are difficult to deal with, and there’s no feasible way for us to filter them out of the dataset.

4.2 Next steps

References

[1] Dimitrios Effrosynidis, Alexandros I. Karasakalidis, Georgios Sylaios, and Avi Arampatzis. The climate change twitter dataset. *Expert Systems with Applications*, 204, 2022.

[2] Sara Rosenthal, Noura Farra, and Preslav Nakov. SemEval-2017 task 4: Sentiment analysis in Twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 502–518, Vancouver, Canada, August 2017. Association for Computational Linguistics. doi: 10.18653/v1/S17-2088. URL `https://aclanthology.org/S17-2088`.

[1] Note that this made-up sentence does not necessarily reflect the opinions of any of the collaborators of this project.