**Predicting Corporate Climate Impact Using Twitter Activity**

**Jules Cazaubiel & Nicholas Tourtillott**

**Abstract**

Anthropogenic climate change is one of, if not the most pressing issues humanity has ever faced. According to the Carbon Majors Report, published in 2017, just 100 companies were responsible for roughly 71% of all global emissions. With this in mind it remains an imperative that we hold corporations and their greed accountable for the climate crisis they are driving and the danger they are putting humanity in. To do so we have created a methodology which can predict a corporation’s environmental impact grade with ~80% accuracy taking only the corporation’s twitter feed as an input. This first of its kind method will allow journalists, scientists, and others to hold corporations accountable for the environmental damage they cause regardless if the company wishes to disclose these behaviors themselves.

**Introduction**

Anthropogenic climate change is one of, if not the most pressing issues humanity has ever faced. The current climate crisis has already been tied to increased instances of extreme weather events and unprecedented storms, which has led to previously unseen levels of damage as we are unable to adapt to the natural consequences we as a society have caused. According to the Carbon Majors Report, published in 2017, just 100 companies were responsible for roughly 71% of all global emissions (Griffin, 2017). In fact this report emphasizes the danger our environment faces from corporate greed, indicating that if these trends remain unchanged global average temperatures may rise five degrees Celsius by the end of the century. Such a change would cause catastrophic damages to our environment, including species extinction and unsustainable food production (Dahlman et al., 2021). With this in mind it remains an imperative that we hold corporations and their greed accountable for the climate crisis they are driving and the danger they are putting humanity in.

Increasingly, social media platforms are becoming a core medium for the larger social discourse. To this point corporate activity on these platforms, like twitter, is becoming more and more common as they try to reach new audiences. Corporate activity on social media may attempt to advertise jobs, push products, respond to customer complaints, preach their social efforts, and generally try to proselytize everyday people to become their supporters and consumers (Stohl et al., 2015). Basically, many corporations are attempting to be relatable to the average person through their social media activity, however this prevents them from truly owning the damage to the environment they are causing. In fact, the opposite is often true where corporations are quick to push their social efforts on social media as an attempt to garner more public favor. It is our intent to determine if this activity is truly genuine, in regards to the current climate crisis, or if their platitudes and commitments are truly duplicitous as we suspect (Stohl et al., 2015).

Our project aims to classify corporate social media presences, relating it to environmental impact and carbon emissions. This should allow us to determine common trends, and differentiate between companies that are truly mindful of their impact versus companies that simply advertise it-- creating a way to hold corporations accountable for their actions. Ideally, this method would even allow us to estimate if a corporation’s social media presence was duplicitous, indicating a damaging environmental impact, even if such environmental data has not been disclosed publicly. However, our team and methods have their limitations. Mainly, the data we have available to use is a 2013 report on corporate climate emissions from the Carbon Disclosure Project. As such our social media search is also limited to that year, which may limit our results as this was a time when social media was less pervasive in our culture, however, we believe that this project could stand as a framework of investigation given robust modern datasets.

**Background**

Given we are attempting to classify corporate environmental impact from their social media presence, the backbone of our project is natural language processing (NLP). NLP is a branch of artificial intelligence which aims to allow computers to process written or spoken language much the same way that humans do. In our case we employed current NLP methods to process and vectorize a corporation's Twitter feed into a set of features which we then fed into our model.

To move in chronological order, we began by scraping the feed of corporate Twitter accounts we compiled from the S&P 500. Web scraping is a type of methodology that allows scientists to gather mass amounts of information from a website. In our case that website was twitter and the information we were scraping into a dataset were the tweets of these major corporations. Tweepy and snscrape stand as two of the current standard APIs utilized for Twitter web scraping. Both are implementable through python in relatively easy to use and freely available packages. However, snscrape has certain advantages over tweepy (Roesslein, 2020). Mainly, snscrape despite having admittedly poorer documentation allows the user to circumvent some of the limitations and restrictions upon web scraping set by tweepy (JustAnotherArchivist, 2018). Of those limitations is a lack of a need for registration with the Twitter Developer Platform, which at times may not approve or even acknowledge your request to access their platform. So with that in mind and given their relatively comparable functionalities otherwise, we decided to move forward with snscrape as our Twitter scraping API.

Next we moved onto the actual NLP portion of the project where we faced a variety of choices. Current NLP methods tend to use lemmatization and tokenization as means to process a piece of text and vectorize it for input into standard machine learning (ML) algorithms. For this project we used a python package called NLTK, otherwise known as the Natural Language Toolkit (Bird et al., 2019). NLTK is the premiere package for natural language processing in python and appears to be the current gold standard of the field. NLTK offers simple methods in text pre-processing such as: cleaning for punctuation, word removal, and lemmatization. Cleaning for punctuation and the removal of common and truthfully meaningless words such as “the”, “a”, “it”, etc. increases the later efficiency of text processing later by removing unnecessary bits of information while not losing the overall meaning of the text in question. Lemmatization is the process of reducing a word to its root and then converting that to its base form. For example, the word “informed” would first be reduced to its stem “inform” and then converted to the base word of “information” (Bird et al., 2019). This process allows us to maintain the overall meaning of a text while not having to process each unique word for vectorization. The exact method of vectorization we used was TF-IDF, or term frequency inverse document frequency. Essentially what that means is that the frequency of a word in a body of text increases proportionally to its attributed weight (Bird et al., 2019). This is inversely proportional to the frequency of the word in the corpus, or the given library of structured texts within the package. This method has one drawback in that it cannot properly capture the relation between words in a sentence, which is an inherent strength in another method of NLTK called word2vec. However, word2vec is best applied in settings of shallow neural networks, which is not the type of model we are intending on implementing, and therefore have chosen TF-IDF as our method of vectorization (Bird et al., 2019).

Finally to quickly implement and test a variety of models we chose to rely on the sci-kit learn python package when compared to other equivalents like tensorflow or pytorch. Compared to sci-kit learn, tensorflow is a relatively lower level package which falls short of the comprehensive ML models and metrics offered by its competitor (Abadi et al., 2016). In a separate vein, pytorch, while similarly comprehensive to scikit-learn, is simply better suited for deep learning applications which we found unnecessary for the scope of our project (Paszke et al., 2019). As such, we determined the easy to implement and comprehensive ML models of scikit-learn to be the best approach for the creation and testing of our project (Pedregosa et al., 1970).

As we progress through this paper we will generally discuss our dataset and methods, the results and comparisons of different model implementations, the limitations of our approach, and, finally, the potential insights we have gained along with future next steps for this area of study.

**Methods**

Before we dive into the exact methods of our study we find it beneficial to formulate the problem in a more official context. We are attempting to create a classifier for corporate twitter activity with the purpose of predicting their climate impact grade as defined on the scale provided by the Carbon Disclosure Project. For each company we include in our dataset we will collect 150 of their tweets, process them through NLP methods described below, and feed that processed data into our model for classification. The labels, or grades, that our model can assign assign nine unique grades on an A-E scale (A, A-, B, B-, etc.) as established by the Carbon Disclosure Project.

*Datasets*

As with any ML project our dataset is as important as the methods that we apply to them to generate our models. For this project we generated an entirely new dataset which we then used to test and train all models. Since our aim for this project was to implement a means of classifying and predicting corporate environmental impact from their social media activity we obviously needed a dataset which could relate those two aspects of a company. Unfortunately, such a dataset had not existed prior to our undertaking of the project, so we decided to create our own.

As mentioned previously, our dataset has two main parts: the corporate twitter activity and an environmental impact grade awarded by the Carbon Disclosure Project (CDP), a major nonprofit organization which monitors environmental impact across the globe. Particularly, we are utilizing the grades for S&P 500 corporations within the 2013 release of the Carbon Disclosure Project’s report due to funding limitations (Cdp, 2016). Despite the relative age of the dataset, we find that the creation of such a model and the methods behind it could still hold true for modern data. Such a limitation could easily be remedied so that a future team could implement the model with a similar, yet more up to date, dataset which would provide predictions which reflect the current state of corporate environmental impact in the world.

The other large focus of our dataset is the twitter activity of all corporations on the S&P 500 which had passed filtering-- those in the CDP dataset who also had a twitter account. The collection of this data in particular was done through web scraping using the snscrape API and we will elaborate on that specific methodology later in this section. Our collection of twitter data ranged from all tweets by the corporations in question starting in 2011 all the way to present day. This ensured that our dataset would be robust enough to properly train the model, while also giving this portion of our dataset applicability with both dated and contemporary releases of the CDP corporate grades. In the context of our model, the tweets of each company acted as our features for training, testing, and prediction while the actual CDP grades were the labels. The exact mechanics of how we vectorized the tweets will be further discussed below, however, NLP methodologies were utilized so that we could properly input this data into our model.

*Web Scraping*

In order to web scrape, we first needed to obtain a twitter username for the companies we selected. This was done by going through all the companies in our list and manually finding their social media handles. As most companies link to their social media accounts on their main website, this process was tedious but ultimately not complicated. Usernames were recorded in a company information csv file containing company names, twitter usernames, and the grades awarded by the CDP. Then, we developed a python script to run the scraping itself using Snscrape. This script iterated through company usernames to run Snscrape commands in the command line, with each command specifying the username, the maximum number of tweets to get (750), and the lower date limit for the scraped tweets (JustAnotherArchivist, 2018). This resulted in the production of an individual JSON file for each company, containing all the scraped tweet information. These were iterated through and the content of each tweet was saved in a common csv file along with its date of creation and the username connected to it. This resulted in an easily readable tweet file that could be used in conjunction with the company information file for the next part of our project.

*Natural Language Processing*

In order to actually use machine learning techniques on the tweets themselves, they first had to be converted to vectors. We used the Natural Language Tool Kit (NLTK) python library. First URLS, and other characters were removed so that only words remained and put into all lowercase. This left only words remaining and prevented the other characters and capitalizations from making the same words appear different. Next stop words were removed from each of the tweets. Stopwords are common words that carry little meaning, but appear frequently like and, a, and the. We used NLTK's built in English stopword list which contains 40 stop words. Stop words were removed to reduce the number of dimensions and prevent the abundance from interfering with the bag of words. Lastly, in order to reduce the number of words more, we lemmatized all of the words using NLTK's built-in lemmatizer. Lemmatization is the process by which words are stripped down to their root. For instance quick, quicken, quickly, quickest, and quicker, would all be converted into quick.

*Tweet Vectorization*

Once the tweets were preprocessed, they were converted into vectors to be used in the machine learning algorithms. We decided to go with a bag of words approach rather than a syntactical approach since building meaning tables was either expensive or time costly. In a bag of words approach, each word in the document is counted and a vector is formed by these counts. Zeroes are also included for words that don't show up in the document, but do show up in the entire collection of documents also known as the corpus. Because many of the words do not show up in a given tweet, many of the elements are zeros which makes the data sparse. Despite the removal of stopwords, we wanted to also bring out words that had more meaning. We did this by employing a term frequency-inverse document frequency (TFIDF) to each of the tweets. TFIDF works by first calculating the frequency of a word in a document and then multiplying that value by the logarithm of the total number of documents divided by the number of documents the term appears in. This means that even if a word appears many times in a document, if the same word appears in many of the documents, then the TFIDF score will be lower. The Scikit-Learn TFIDFvectorizer was used to perform TFIDF. This performs TFIDF on each tweet based on the corpus of all tweets in the sample (Bird et al., 2019).

*Model Implementations & Evaluation*

We started by implementing a support vector machine learning (SVM) model and a random forest model to predict the grade of the company writing each tweet, as these models seemed to be good fits for this type of classification problem. Support vector machine learning works by finding a hyperplane that best splits the data. Random forest works by randomly splitting the data with decision trees. Naive bayes and Gaussian Process algorithms both rely on a foundation of Bayesian statistics, though given the structure of our datasets neither were a good fit which we will discuss later in our results. For neural networks we used a multilayer perceptron, though our topology was not a good fit for this particular dataset others could provide a higher performance. In order to assess the performance of our models, we decided to use K-fold cross validation. The standard scikit-learn method was used, shuffling the data before splitting it into 10 groups and running the model 10 times with different training and testing sets. For each iteration of the model, we recorded 2 metrics (training accuracy and test accuracy) before plotting them and calculating their average for each model (Pedregosa et al., 1970).

*Cluster Visualization*

To visualize clusters, we used Sklearn's implementation of TSNE. T-SNE works by creating a heatmap of distances in the total dimensions, then projecting that data into a lower dimensionsional representation by using the T-distribution to find distances between points. T-SNE was used to project the data into two dimensions. Sklearn's implementation of TSNE can only project up to 3 dimensions, though we use only a 2D projection.

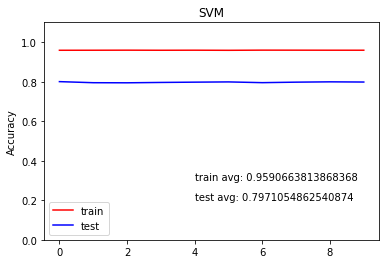
*Case Study*

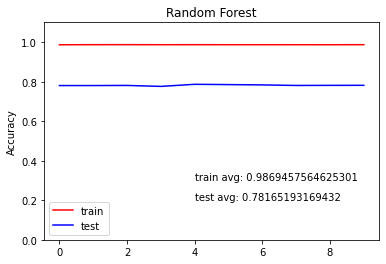
Finally, we decided to take a brief look at a collection of large companies not included within our original dataset, and predicted grades for them using our highest scoring random forest classifier. Our choice of companies was motivated by a few factors: size, industry, and willingness to disclose their impacts. Firstly, given that major corporations seem to be largely responsible for the current emissions trends and the climate crisis we are facing, we decided to delve into some of the largest companies in the world by collecting them from the S&P 500. In this process we tried to reach out into different industries of the private sector like tech, energy, food, entertainment, etc. Our reasoning being that we were curious if our model favored any particular industry and to see if any industry was deemed particularly harmful based on the CDP grading standards we trained our model on. Finally, we decided on this collection of companies in particular because as of the 2013 iteration of this CDP report none had decided to work with the non-profit and submit to their scoring, meaning they have a precedent of being not particularly forthcoming in terms of their environmental impact.

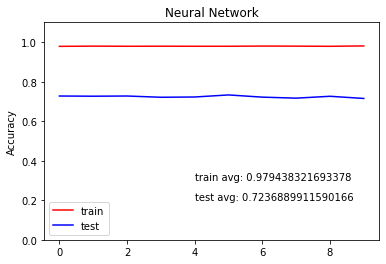
**Results**

After collecting, cleaning, and engineering our data we were ready to train our models. As a brief overview we attempted to construct a SVM classifier, a random forest classifier, a naive bayes classifier, a gaussian process classifier, and a neural network classifier for environmental impact grades. As we move forward in this section we will discuss the accuracy and performance of each model as well as attempts to improve these metrics. It is worth noting that all models were consistent in that they exhibited exceedingly high train accuracy scores while performing consistently lower during testing.

First, the SVM model performed well with an average test accuracy of about 0.797. The behavior of the SVM model can be seen below in figure 1A, where it is evident that the performance of the model remains relatively consistent across the process of cross validation. Next, looking at the random forest classifier we saw a slight decrease in average test accuracy compared to the SVM classifier at 0.781. Similarly, the performance of the random forest classifier remains consistent as well in figure 1B. In figure 1C you will see the performance of our neural network model, which performed significantly lower than either the SVM and random forest models after 10-fold cross validation with a test accuracy of 0.723.

(A)

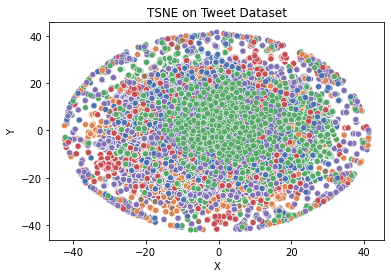
(B)

(C)

**Figure 1.** Accuracy scores of the **A.** SVM, **B.** Random forest, and **C.** Neural Network models over 10 runs, using k-fold cross validation.

You’ll note that test and train accuracies are not presented for our naive bayes or gaussian process models. Given the nature of our datasets neither were capable of properly handling our datasets and given time restrictions we could not troubleshoot or adapt to other methods. Specifically, gaussian process models only perform well with datasets which exist in a lower dimensional feature space, so naturally a natural language dataset where each word is its own feature will not perform well. Similarly, our dataset was too sparse to run with the naive bayes algorithm we had implemented. While it may have been possible to attempt a dimensionality reduction technique to adapt to this setback, we opted not to given time restrictions and given past dimensionality reduction techniques resulted in a significant test accuracy drop in earlier iterations of this project.

Below in Figure 2 is a TSNE projection of our classified tweets. This shows that when reduced to two dimensions there are no clear clusters of each letter grade. This makes sense considering the sparseness of natural language processing data. Each word would have very little impact on the placement of the point.



**Figure 2.** T-SNE plot of tweets. Each point represents a tweet in our dataset and its color corresponds to the environmental impact grade assigned to them by our model.

Moving on to our case study, the results below show that all of the companies investigated had been assigned a predicted grade of B. This could indicate that while these companies neglected to cooperate with the CDP, their climate impact is at the very least on par with other corporations. From this outcome we can draw a couple conclusions. For one, we can see that our model is not discriminatory to one particular industry, which makes sense because that is not considered as a feature during our classification process. To that point we have reason to believe that our model is fairly subjective when grading companies and does not punish some corporations with a lower grade just for being in a particular business. Next, we can draw the conclusion that while these are particularly large corporations they do not seem to have a comparably worse impact than others at least on the scale provided by the CDP. However, it is important to note that this grading scale may quite honestly not be as harsh as it should given our current situation. Yes, it is true that this grading scale assigns these corporations an average performance, but that is compared to other corporations. Essentially, if we are comparing the worst polluters known in the history of humankind many are bound to be average, however, that does not necessarily mean they are causing minimal harm to our environment. These results could also point to the need for stricter standards by which we hold these corporations accountable, especially considering companies like Amazon have only seen significant increases in their net carbon emissions year over year for the past half a decade (Palmer). So, while our model appears to grade corporations fairly on a scale, it is worth noting that the scale is possibly more forgiving than it should be or than we can afford.

| **Company** | **Predicted Grade** |
| --- | --- |
| Dominion Energy | B |
| HP | B |
| Hersheys | B |
| Marathon Petroleum Corporation | B |
| Tyson Foods | B |
| Waste Management | B |
| Amazon | B |
| Netflix | B |

**Table 1.** A table of corporations of interest given their size, industries, and lack of climate impact grade assigned by the CDP. The predicted grades were those assigned by our highest performing random forest classifier and show that these companies are predicted to have an average climate impact based on the CDP grading scale.

**Conclusions**

Firstly, we would like to address some of the limitations of our work. As explained above our results and model are based on outdated climate data from nearly ten years ago, however we believe that the framework of our model should be strong enough in principle to be reproduced with a modern version of the CDP dataset. So while our current climate impact predictions are likely out of date it could be easily retrained with current climate grades and will likely hold similar levels of performance. Additionally, our methods are reliant on corporations being fairly active on social media platforms. As such this method likely fails to capture the correct environmental impact of smaller companies who are less apt to be active online. However, we believe that this method is best used for large corporations since they are likely to cause the most environmental impact. Luckily, these are also the companies who are most likely to participate in online spaces the most through their advertising and PR activities.

With these limitations being said, we believe that the main goal of the project was still achieved. We still produced a method which can hold corporations accountable for the damage they cause, with a reasonable degree of accuracy, using a model which only requires social media activity meaning we can hold them accountable whether or not they decide to be transparent about this fact. As far as we are aware this is the first attempt to hold corporations accountable in such a way and we recommend that our tool be used for that purpose. In regards to our model this shows some very key insights about this area of research. For one, company social media activity can be used to discern their environmental impact. This is crucial for holding them accountable for their action and opens the door to applying similar methods to delve into other corporate behaviors.

Finally, the next steps of this area of research should be two pronged. Scientists should first attempt to build off of this framework using more robust modern datasets and methods in an attempt to further improve the accuracy of our classification model. Doing so would give us a method which can hold corporations at an even higher degree than the one we provide currently. Secondly, this method should be implemented in real world applications to attempt to hold corporations accountable for the damage to the planet that they are causing and refuse to own up to. This model would be in the best hands of journalists and other researchers who are best fit to delve into these companies and expose them for the true impact they are responsible for. Otherwise this model was produced in vain.

**Bibliography**

Abadi, Mart&#x27;in, Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., … others. (2016). Tensorflow: A system for large-scale machine learning. In *12th $USENIX$ Symposium on Operating Systems Design and Implementation ($OSDI$ 16)* (pp. 265–283).

Bird, S., Klein, E., & Loper, E. (2009). *Natural language processing with Python: analyzing text with the natural language toolkit*. " O&#x27;Reilly Media, Inc."

Cdp. (2016, June 1). *2013 - global 500 emissions and response status: CDP Open Data Portal*. 2013 - Global 500 Emissions and Response Status | CDP Open Data Portal. Retrieved December 1, 2021, from https://data.cdp.net/Companies/2013-Global-500-Emissions-and-Response-Status/marp-zazk.

Dahlman, R. L. and L. A. (2021, March 15). *Climate change: Global temperature*. Climate Change: Global Temperature | NOAA Climate.gov. Retrieved December 1, 2021, from https://www.climate.gov/news-features/understanding-climate/climate-change-global-temperature.

Griffin, P. (2017). Carbon majors report 2017. Retrieved December 1, 2021, from <https://b8f65cb373b1b7b15feb-c70d8ead6ced550b4d987d7c03fcdd1d.ssl.cf3.rackcdn.com/cms/reports/documents/000/002/327/original/Carbon-Majors-Report-2017.pdf?1499691240>.

JustAnotherArchivist. (2018). snscrape: A social networking service scraper in Python. Retrieved December 1, 2021, from https://github.com/JustAnotherArchivist/snscrape.

Palmer, Annie. “Amazon's Carbon Emissions Rose 19% in 2020 Even as Covid-19 Pushed Global Levels Down.” *CNBC*, CNBC, 30 June 2021, https://www.cnbc.com/2021/06/30/amazon-says-carbon-emissions-rose-19percent-in-2020.html.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., … Chintala, S. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32* (pp. 8024–8035). Curran Associates, Inc. Retrieved from http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (1970, January 1). *Scikit-Learn: Machine learning in Python*. Journal of Machine Learning Research. Retrieved December 1, 2021, from https://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html.

Roesslein, J. (2020). Tweepy: Twitter for Python! *URL: Https://Github.Com/Tweepy/Tweepy*.

Stohl, C., Etter, M., Banghart, S., & Woo, D. J. (2015). Social Media Policies: Implications for contemporary notions of corporate social responsibility. *Journal of Business Ethics*, *142*(3), 413–436. https://doi.org/10.1007/s10551-015-2743-9