Master Thesis

Assessing Public Engagement in Climate Change from the Eye of Twitter

Soha Torki

Supervised by
Prof. Silvia Chiusano
Dr. Hamed Haddadi

Department of Control and Computer Engineering
Polytechnic University of Turin
July 2018

Abstract

Climate change which refers to the shift in weather patterns and global temperatures is one of the main issues across the world today. According to scientists, the extensive amount of greenhouse gases emissions has caused a tremendous increase in global temperatures in recent years in such a way that August 2017 was the second warmest August in 137 years of record-keeping.

Twitter as one of the most popular social networking websites, has become a common platform for climate change conversations. The climate change debates on Twitter is generally categorized into two types of opinion. One group believes that climate change is happening and actions should be taken to fight it and protect the earth. While the second group does not believe in climate change. This group that is known as climate change deniers or skeptics, claims that climate change is not happening or if it is, this is not caused by the human activities and there is no need to take actions.

In this thesis, during a five-month period, a dataset of tweets about climate change is created using Twitter API in order to analyze the public opinion on climate change by means of machine learning techniques as an approach for sentiment analysis. The tweets are collected using popular climate change hashtags and are labeled as positive and negative which correspond to two opinions about climate change discussed earlier. The positive group believes in climate change while the negative group supports the climate change denial. Then different classification algorithms are applied to the dataset to classify the tweets. The experiments show that Support Vector Machine classifier and the Logistic Regression classification, using the unigrams, sentiment lexicons, word embeddings and Twitter-specific features have the best performance.

Analysis on the dataset shows that the majority of the tweets are positive. In addition, most of the negative tweets are from the United States mainly in Georgia, Oklahoma, Kentucky, Texas, Kansas, Alabama and Mississippi. These states are typically from the Republican party, the party of the current president of the United States, Donald Trump, who announced

the withdrawal of the U.S. from the Paris agreement in June 2017. This is while according to the Emission Database for Global Atmospheric Research (EDGAR) and EPA, the U.S. is the second biggest CO_2 emitters in the world and Texas has the most amount of annual CO_2 emission in the U.S. it is obvious that climate change deniers are the biggest CO_2 emitters who want to stop the regulations on their activities.

These analyses could also be performed on other social networks and on a bigger dataset to develop this work and make a wider assessment.

Dedication

This thesis is dedicated to my beloved parents and brother who have always supported me.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Extensive quantities of greenhouse gas emissions into the Earth's atmosphere has led to the increased average global temperature, known as global warming, resulting in the climate change. According to scientists, the rate of the warming has been unprecedented over the last 25 years, to the extent that 2016 was the hottest year since modern record-keeping began in 1880 in which the average global temperature was 0.99°C higher than the 20th century mean [12]. The planet is suffering from the extra heat which has caused glaciers and sea ice melting, sea level rising, precipitation shifting patterns and oceans acidity.

According to the researches done by scientists, greenhouse gases play the biggest role in increasing the global temperature and causing the climate change. However greenhouse gases are essential for the human life as they keep the sun's heat, the excessive amount of these gases makes this extra heat to be harmful to the earth and human. The greenhouse gases typically include carbon dioxide (CO_2), water vapor (H_2O), methane (CH_4), nitrous oxide (N_2O) and chlorofluorocarbons (CFCs). Among these gases, the water vapor (H_2O) is the most plentiful [13] but because the amount of water vapor in the atmosphere is not affected directly by the human activity, it is not considered as a greenhouse in some categories. (CO_2) concentration is increased rapidly by fossil fuels burning and other human activities.

There are some international agreements on climate change in order to prevent the worsening of the climate change by stopping the harmful and dangerous human activities. The main objective of these agreements is controlling the concentration of the greenhouse gases in the atmosphere. Paris agreement is the newest agreement which is adopted in Paris in 2015 and has 169 participants. The principal goal of this accord is to limit the global temperature rise well below 2°C above pre-industrial levels to reduce the risks of climate change [14]. In June 2017, Donald Trump declared the withdrawal of the United States from Paris agreement because of its economic effects on his country. However, the U.S. is among the top CO_2 emitters causing the climate change gets worse.

The climate change topic has been widely discussed in social media recently. The Twitter is one of the most used social platforms for this subject. The importance of this issue makes some people really worried about the future of human life and the earth. This causes the creation of different campaigns and groups to make others aware of the matter and to persuade them to take actions for climate change. On the other hand, there is another behavior which conveys a belief that climate change is not happening or if climate change exists, it is not caused by human activities. The presence of these two beliefs caused a polarized discussion which could be followed on Twitter.

In this thesis, the objective is to analyze the climate change debate on Twitter using sentiment analysis techniques. Sentiment analysis is a method to mine the opinions of the people on different subjects [15] and it could be an efficient technique for social networks analysis. Sentiment analysis could be done through sentiment lexicon approaches and machine learning approaches. However, there are also hybrid methods that make use of both former approaches. In this work, different machine learning algorithms are used to make a classification on Twitter data in order to analyze the opinions on climate change subject. Since beside machine learning techniques, sentiment lexicons are also used, it could be considered as a hybrid method for sentiment classification.

Since there is no available dataset of Twitter data specific to climate change subject, The dataset is created using Twitter API which contains almost 2,200,000 tweets. After removing the duplicates, about 120,000 tweets are labeled as positive or negative. Positive indicates a supporting opinion about taking actions for climate change issue while the negative shows climate change skepticism or denial. Then the classifiers are trained using machine learning techniques to make a binary classification on the dataset.

In this thesis, Chapter 2 discusses the climate change issue, its causes and effects and the climate change conversation on social media. in Chapter 3 the sentiment analysis and its techniques are studied, then the related works on sentiment analysis are reviewed. Chapter 4 discusses the data collection and the process of creating the dataset of climate change related tweets. In Chapter 5 the implementation of sentiment classification on the provided dataset is discussed using different machine learning algorithms. Finally the Chapter 6 analyses the results derived from the implementation and discusses the climate change deniers data.

Chapter 2

Climate Change

In this chapter the climate change issue is described generally, then the impact of social media on this issue is discussed. Eventually, climate change conversations on a particular social network which is Twitter, are analyzed.

2.1 Climate Change

Climate change is one of the major issues facing the world nowadays, the change which refers to weather patterns and global temperatures in recent years is believed to be down to human activity. According to scientists at NASA's Goddard Institute for Space Studies (GISS) in New York, August 2017 was the second warmest August in 137 years of record-keeping [1], as can be seen in Figure 2.1.

The Intergovernmental Panel on Climate Change (IPCC) is an international body established by two of the United Nation organizations, United Nations Environment Programme (UNEP) and the World Meteorological Organization (WMO) in 1988 with more than 1300 scientists from the United States and other countries, for the assessment of climate change. Thousands of scientists and experts from all over the world contribute to IPCC work to provide and review reports which assess the scientific, socio-economic and

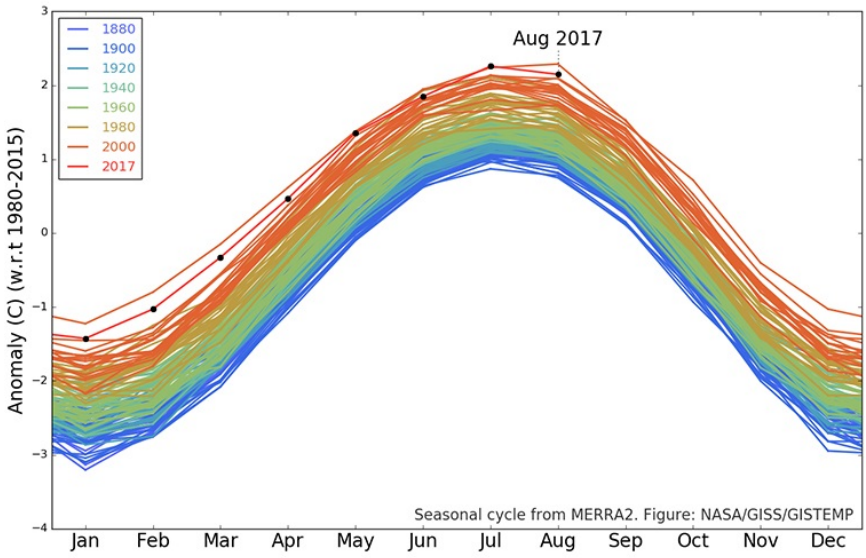

Figure 2.1: GISTEMP seasonal cycle since 1880 [1]

technical basis of climate change. Reports by IPCC contain Summary for policymakers (SPM) which is a summary that approved by all participating governments and could help policymakers. According to IPCCs Fifth Assessment Report (AR5) completed in 2014, Warming of the climate system is unequivocal and since the 1950s, many of the observed changes are unprecedented over decades to millennia. AR5 also states that human influence on the climate system is clear, and recent anthropogenic emissions of greenhouse gases are the highest in history [16].

2.1.1 Causes

Many scientists believe that the main cause of climate change and global warming is the effect of greenhouse gases. As stated by IPCC, anthropogenic greenhouse gas (GHG) emissions have increased since the pre-industrial era, driven largely by economic and population growth, and are now higher than ever. This has led to atmospheric concentrations of carbon dioxide (CO_2), methane (CH_4) and nitrous oxide (N_2O) that are unprecedented in at least

the last 800,000 years [17].

The sunlight entering the earth's atmosphere heats the earth and make it livable, the earth then radiates back the heat upward to space. Greenhouse gases are essential to the survival of the human since they help the earth to keep parts of the sun's heat reaching its atmosphere and to provide a life-supporting temperature. what happened is that the amount of greenhouse gases in the atmosphere has been rapidly increasing over the past several decades, causing the heat radiating from the earth toward space to be trapped in the atmosphere. Water vapor (H_2O), carbon dioxide (CO_2), methane (CH_4), nitrous oxide (N_2O) and chlorofluorocarbons (CFCs) are greenhouse gases that block the heat in the atmosphere from escaping.

Water vapor (H_2O) is the most plentiful greenhouse gas in the atmosphere, but its concentration in the atmosphere acts as a feedback to the climate, in a way that by increasing the temperature, more water is evaporated, leading to more water vapor in the atmosphere [13]. The rise in water vapor causes more of the heat radiated back from the earth to be absorbed and makes the atmosphere warmer.

Carbon dioxide (CO_2) is the minor but most important component of the atmosphere and is the most significant long-lived greenhouse gas. Human activities such as the burning of fossil fuels like coal and oil, lands clearing for agriculture and clear-felling the forests are exacerbating greenhouse gases especially by increasing the concentration of (CO_2). Figure 2.2 shows atmospheric CO2 levels in recent years measured at Mauna Loa Observatory, Hawaii.

Methane (CH_4) concentration is much less abundant than CO_2 but it is a very strong radiation absorber. CH_4 is produced both through natural processes and anthropogenic sources such as rice cultivation, cattle raising and agricultural practices and waste decay in solid waste landfills.

Nitrous oxide (N_2O) is emitted through agricultural activities such as soil cultivation, fertilizers usage, fossil fuel combustion, biomass burning, nylon production and Nitric acid production.

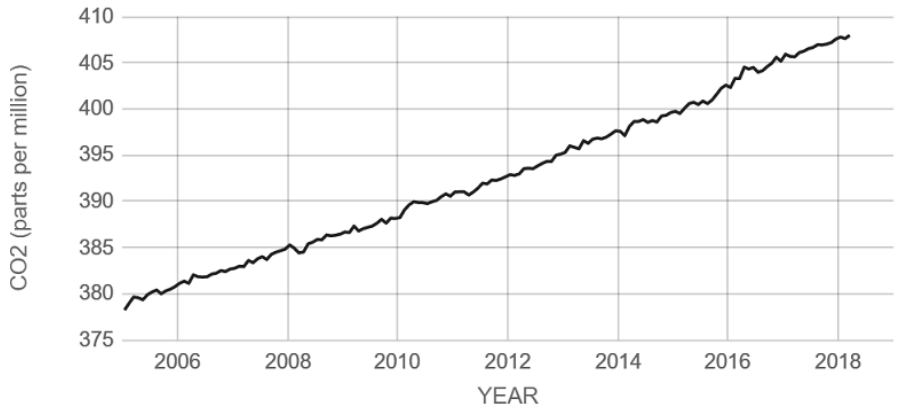

Figure 2.2: CO_2 concentration level since 2005 [2]

Chlorofluorocarbons (CFCs) is a synthetic powerful greenhouse gas which is released through industrial processes. The emission of this gas is in smaller quantities since it was successfully regulated through a global agreement, because of its ability to destroy the ozone layer.

2.1.2 Effects

According to IPCC, In recent decades, changes in climate have caused impacts on natural systems and human systems over all the continents and across the oceans [16]. Since greenhouse gases are largely produced, the global temperature has risen for decades. IPCC forecasts that the global temperature will rise about 2.5 to 10 degrees Fahrenheit over the next decades. Climate change has already affected the world, glaciers have been shrinking and the number of glacial lakes is increased, the sea level has been rising over the past century due to melting ice sheets and seawater expansion, precipitation and flooding has been increased, coral reefs have been destroyed, plants and animal ranges have been shifted over past decades and seasons have been changed in a way that spring arrives earlier and winters are shorter. Figure 2.3 shows the sea level rise between 1993 and 2017.

As global warming has long-term effects there will be also future consequences. There will be more severe weather and the global average tem-

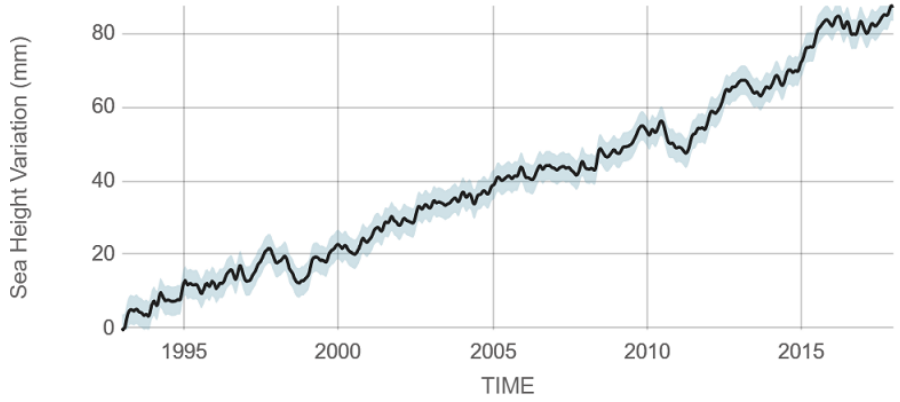

Figure 2.3: Sea level rise since 1993 [3]

perature expected to be increased. The rate of melting of glaciers seems to be increased and will make the sea levels higher. Stronger hurricanes, the higher rate of wildlife extinction, changes in precipitation patterns, extremes of drought and flooding, more acidic oceans and increased threats to human health are expected to be the other main future effects.

2.1.3 International agreements

The main international agreement on climate change is the United Nations Framework Convention on Climate Change (UNFCCC) which has been ratified by 197 countries at Rio Earth Summit in 1992. The objective of UNFCCC is to stabilize the greenhouse gas concentrations in the atmosphere at a level that prevents dangerous human interference with the climate system [18].

2.1.3.1 Kyoto Protocol

Kyoto Protocol is an international agreement which was adopted in 1997, it extends the UNFCCC and commits its parties to reduce the emission of greenhouse gases. Since developed countries are mainly responsible for

high levels of greenhouse gases emissions the agreement requires developed countries to take action.

the second commitment period of the Kyoto Protocol which began in 2013 and is covered by the Doha amendment, states that the participants have to reduce the emissions by at least 18% below 1990 levels. As the United States didn't sign the agreement, Canada left the first period of commitment and Russia, Japan and New Zealand are not participating the second period, this protocol applies to less than 20% of the world emissions.

2.1.3.2 Paris Agreement

Paris Agreement is a new agreement within UNFCCC that was adopted in 2015 in Paris and ratified by 169 parties. The aim of the agreement is to limit the global temperature rise well below 2°C above pre-industrial levels ideally holding the temperature increase to 1.5°C in order to reduce the risks of climate change. The nations signing the agreement were asked to decrease the greenhouse gas emissions and their contribution to cutting emissions will be reviewed every five years. Rich countries are requested to help financially the poorer nations to cope with climate change [14].

In June, 2017, Donald Trump the President of the United States announced that the U.S. will leave the Paris Agreement stating that the accord has a negative economic impact on the U.S. Syria and the U.S. are the only two nations who are not members of the Paris Agreement, this is while the United States has been always among top polluters with high quantities of CO_2 emissions

2.2 Climate Change on Twitter

Social media which are generally online communication channels that provide people interactions by sharing information, has grown tremendously in

recent years. They are dissolving the geographic boundaries and seems to be everywhere with over a billion people on its different platforms.

Twitter is one of the most popular social media platforms and microblogs which broadcasts short messages known as tweets with the size restriction of up to 140 characters. Twitter was created in 2006 and has been grown rapidly with more than 300 million monthly active users. One of the specific features of this social network is trending topic which is a word, phrase or hashtag that is mentioned at a higher rate than others. Trending topics are useful to get informed of what is happening in the world.

Twitter has been a popular medium for climate change issue, which is one of the topics being discussed widely and has been among trending topics many times. Many people are getting involved in this environmental issue through Twitter by making the public aware of the issue, creating new campaigns and encouraging them to take action. Also, politicians, government officials, agencies, and organizations use this social media to express their thoughts about climate change and get a wider audience.

According to some scholars, the climate change debate is generally polarized between those who believe in its occurrence and the human role in it, and those who don't, known as climate change skeptics and deniers. In climate change skepticism/denial some groups claim that global warming and climate change is not taking place at all while others accept that climate change is happening but deny the influences of human activities on it. In this thesis a dataset of tweets about climate change is collected, then the sentiments of the tweets are analyzed.

Chapter 3

State of the Art

In this chapter, The sentiment analysis is discussed, the techniques used for sentiment classification are explained and then the related works done in this subject are studied.

3.1 Sentiment Analysis

Sentiment Analysis which is also known as Opinion Mining, is a method of determining the people's opinions, emotions, and attitudes towards different individuals, events or subjects, computationally.[15].

Sentiment Analysis which refers to the use of Natural Language Processing (NLP) is very useful in social media analysis since it gives a wide overview of the public opinion about different topics. It also helps businesses and companies to gain the insight into social attitudes about their products and brands by means of customers reviews so that they could improve the unsatisfying aspects of their products. Sentiment Analysis is also useful in the field of decision making for consumers as it gives recommendations on choice of the products according to the public opinion [19].

Sentiment Analysis can be used for both subjectivity/objectivity identifica-

tion and feature/aspect identification. In subjectivity/objectivity identification a text will be classified in one of the subjective or objective classes. In feature/aspect identification the goal is to determine whether the sentiment of the opinion on the particular extracted features is positive, negative or neutral.

There are two main techniques for sentiment classification, lexicon-based approaches, and machine learning approaches.

3.1.1 Lexicon-based Approaches

In lexicon-based approach, also known as knowledge-based approach the sentiment is calculated based on the semantic orientation of words in a text. The semantic orientation is the degree of subjectivity and opinion in text [19]. This technique is divided into the dictionary-based approach and corpus-based approach.

In dictionary-based approach opinion word seeds are found and then the dictionary is examined to collect the synonyms and antonyms of the words [20]. The disadvantage of this method is that it cannot find the domain-specific opinion words [21]. In this approach, a dictionary of positive and negative words is required containing the words with their corresponding sentiment values. The sentiment value is a score assigned to each word based on its positivity or negativity. These precompiled and known sentiment words are called lexicons [20]. In this dictionary-based approach, each piece of a text is tokenized and then each token is matched for its lexicon in the dictionary and if the match is found it is translated to its score.

When all the words of the text are assigned to their scores, a combining function is used to combine the scores in order to get a final score which represents the polarity of the text [22]. As the lexicon-based approach is based on an assumption that the overall polarity of a text is the sum of the polarities of all individual words or tokens [23], the combining function to calculate this collective score is the sum or the average.

In corpus-based approach a list of opinion words is created and based on their context-specific orientations, related opinion words are searched in a large corpus [20]. This method does not have the limitation of the dictionary-based approach.

3.1.1.1 WordNet

Wordnet is a lexical semantic database which groups the English words into sets of synonyms called synsets and join them together by means of conceptual-semantic relations. This lexical repository which is used in natural language processing and text analysis contains 155,327 words and 175,979 synsets. [24].

3.1.1.2 SentiWordNet

Sentiwordnet is a lexical resource which is used in Sentiment Analysis. Sentiwordnet provides three numerical sentiment scores for each wordnet synset. These scores correspond to positive, negative and neutral and range from 0.0 to 1.0 and the sum of them is 1 for each synset [24].

3.1.2 Machine learning Approaches

In machine learning technique, the main goal is to improve the performance of the system by training the data. Machine learning approaches are classified into supervised learning and unsupervised learning techniques. In supervised learning, the training data which is a large set of examples, is already labeled, while in unsupervised learning the labeling is not done. In sentiment analysis typically supervised learning methods are used [21].

In this thesis supervised learning approaches which are based on popular machine learning algorithms are utilized. The goal is to build a classifier in order to solve the sentiment classification problem. Two data sets called training data and test data are engaged in machine learning classifiers.

The training data set is used to train the algorithm or model with known outputs for the corresponding inputs. This dataset consists of input vectors and output vectors typically known as targets. Targets correspond to the labels which are already provided in training dataset. Whereas in the test data, the target is unknown and the goal is to train the classifier in a way that it could predict the target or label for the unknown data. The classifier will do this evaluation by interpreting the training dataset.

In general, machine learning approaches to sentiment classification problems consist of two steps:

1. Training the model or algorithm by means of training dataset which contains labeled data.
2. Classifying the unlabeled or unclassified testing data using trained data.

Machine learning algorithms used in sentiment classification are Naive Bayes Classifier, Linear Regression, Logistic Regression, Ridge Regression, Support Vector Machine (SVM), Random Forest and K-Nearest Neighbours (KNN).

3.1.2.1 Naive Bayes Classifier

Naive Bayes classifier is one of the probabilistic classifiers and simple approaches to text classification which is based on Bayes' theorem. This classifier is based on the assumption that the features of a text are independent of each other, It assumes that a text is a set of words or features and the probability of a word in the text is independent of the position of it and the existence of other word [25].

in Bayes' theorem,

$$P(h \mid D) = \frac{P(D \mid h)\, P(h)}{P(D)}$$

Where $P(D) \neq 0$,

$P(h \mid D)$ is the probability of hypothesis h, given the data D. This is called the conditional probability or the posterior probability [26].

$P(D \mid h)$ is the probability of data D knowing that the hypothesis h is true. This is also a conditional probability.

$P(h)$ is the probability of observing the hypothesis h and is independent of D. This is called the prior probability of h.

$P(D)$ is the probability of observing D which is independent of h.

The Naive Bayes classifier makes use of a decision rule in order to find a class that maximizes the posterior. This rule is called *Maximum a posteriori* hypothesis in which the goal is to find the most probable hypothesis given the training data.

Maximum a posteriori hypothesis h_{MAP}:

$$\begin{aligned} h_{MAP} &= \arg\max_{h \in H} P(h|D) \\ &= \arg\max_{h \in H} \frac{P(D|h)P(h)}{P(D)} \\ &= \arg\max_{h \in H} P(D|h)P(h) \end{aligned}$$

As in sentiment classification problem, the goal is to classify the text into labels which correspond to the sentiments and to know if a feature set belongs to a specific label, the Bayes' theorem could be rewritten as,

$$P(label \mid features) = \frac{P(features \mid label)\, P(label)}{P(features)}$$

Where $P(label \mid features)$ is the prior probability that the given features is classified as label [15].

According to the assumption of Naive Bayes classifier, features are independent. Therefore the formula would be shown as,

$$P(label \mid features) = \frac{P(label)\, P(f1 \mid label)\, P(f2 \mid label) \ldots P(fn \mid label)}{P(features)}$$

the *Maximum a posteriori* hypothesis would be rewritten as,

$$label_{MAP} = \arg\max_{y} P(label) \prod_{i=1}^{n} P(fn \mid label)$$

Gaussian Naive Bayes This is an extension of Naive Bayes to real-valued or continuous attributes. In this classifier, continuous data associated to each feature is distributed based on Gaussian distribution. Gaussian or Normal distribution is one of the most popular continuous probability distribution which is specified with two parameters, μ or the mean which is the average value and σ^2 which is the variance of the values.

As the likelihood of the features is Gaussian, the conditional probability will be computed as,

$$P(xi \mid y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}(\frac{xi-\mu}{\sigma})^2}$$

Multinomial Naive Bayes It extends the use of Naive Bayes classifier and implements it for the multinomial distribution of data. In this model, the frequencies of each word are used to represent the data by a multinomial distribution [27].

Bernoulli Naive Bayes This model implements the Naive Bayes and is also used in text classification problems. Features are independent boolean binary variables and in contrast to the Multinomial model, the occurrence of these binary features are used.

3.1.2.2 Linear Regression

Linear Regression is one of the simple methods of supervised learning which is widely used in statistical learning methods. As the name of this model indicates, it assumes a linear relationship between input variables X and the dependent output variable Y. The model is called Simple Linear Regression when the input variable is a single variable X while when there are multiple input variables, it is called Multiple Linear Regression.

The simple Linear Regression is shown as,

$$Y = \beta_0 + \beta_1 X$$

Where,

Y is the single output.

X is the single input variable.

β_0 is the bias coefficient which gives the possibility to move up and down in a two-dimensional plot, the bias coefficient is also known as intercept.

β_1 is the coefficient of the feature X.

In simple Linear Regression, the coefficients could be estimated by calculating the statistical properties such as the means, the standard deviations, correlations and covariance [26].

Ordinary Least Squares When the model is not a simple linear regression and there are multiple input variables, this technique is one of the most common methods used in order to calculate the coefficients. The objective of the Ordinary Least Squares method is to minimize the sum of square residuals [26]. The approach is to calculate the square of the distance from each input point to the given regression line and calculate the sum of all the squared distances. Finally, this value should be minimized.

3.1.2.3 Logistic Regression

Logistic regression is one of the powerful techniques for binary classification problems. These problems have two class values of 0/1 representing True/False, Yes/No, Success/Failure or any other binary values.

Logistic regression makes use of the logistic function which is a function to map any real-valued input to a value between 0 and 1. Logistic function, also called sigmoid function produces an output score which indicates the probability of an event occurrence [4]. This is what the linear regression is not able to do, because it can output a result out of the range 0 to 1. Figure 3.1 shows the difference between linear regression and logistic regression.

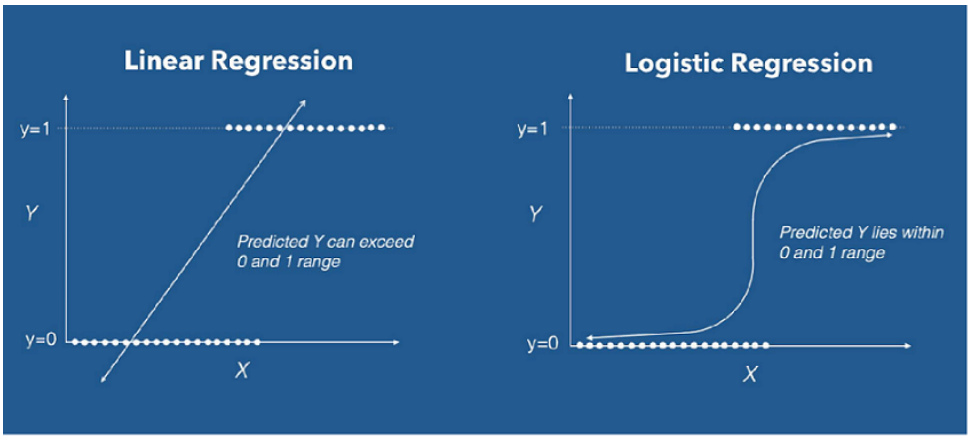

Figure 3.1: Difference between linear and logistic regression [4]

As shown above, the logistic function is an S-shaped curve and does not map the input to a value out of the range 0 to 1. It is represented as;

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Where,

σ is the output between 0 and 1.

e is the Euler's number.

z is the real-valued input.

Logistic regression is represented by a logistic equation as,

$$y = \frac{e^{\beta_0+\beta_1^x}}{1 + e^{\beta_0+\beta_1^x}}$$

Where,

y is the predicted output.

β_0 is the bias coefficient known as intercept.

β_1 is the coefficient of the input value x.

Coefficients of the logistic equation can be predicted by maximum likelihood estimation from training data, while in linear regression ordinary least squares are used for prediction of coefficients.

Maximum Likelihood Estimation It is a method to estimate the values of parameters in statistical models for given data. In logistic regression, maximum likelihood estimation is used to predict the coefficients of the logistic equation [26]. Its goal is to find some values for coefficients in a way that the errors in predicted probabilities are minimized and therefore the likelihood function is maximized.

3.1.2.4 Ridge Regression

Ridge regression is a method to analyze multiple regression data with the multicollinearity problem [28]. Multicollinearity describes a situation in a multiple regression model when predictors are correlated with each other. It indicates that a non-linear relationship between variables exists.

Ridge regression is a regularized linear regression model. knowing this fundamental concept that samples from a specific class lie on a linear subspace, it is possible to represent new test data as a linear combination of training data of a specific class. This assumption can be formulated as a linear model in terms of ridge regression [28]. The ridge regression tries to minimize the impact of irrelevant features on trained model by lowering the coefficients.

3.1.2.5 Support Vector Machine

Support Vector Machines are one of the most popular machine learning techniques. In this algorithm, the objective is to find a separator which separates the classes in the search space with maximum distance [20]. This separator is a hyperplane which is a line that splits the input variable input variable space. In a two-dimensional space, the hyperplane is visualized as a line that separates the input variables into class 0 and class 1. The margin is the distance between this line and the closest data point. The line with the largest margin is considered to be the optimal line. This large margin makes the support vector machine to be highly effective because rather than classifying the input variables into classes, it also provides the largest distance in separation [29].

Letting the class $c_j \in \{-1, 1\}$ be the correct class (positive or negative) of document denoted by d_j , the solution can be given by vector w [29].

$$\vec{w} := \sum_j \alpha_j c_j \vec{d_j}, \alpha_j \geq 0$$

Where,

α_js are obtained by solving a dual optimization problem.

$\vec{d_j}$ such that α_j is greater than zero are called support vectors since they are the only document vectors contributing to $\vec{w}$ [29].

As in Figure 3.2 is shown support vectors are the closest data points to the hyperplane.

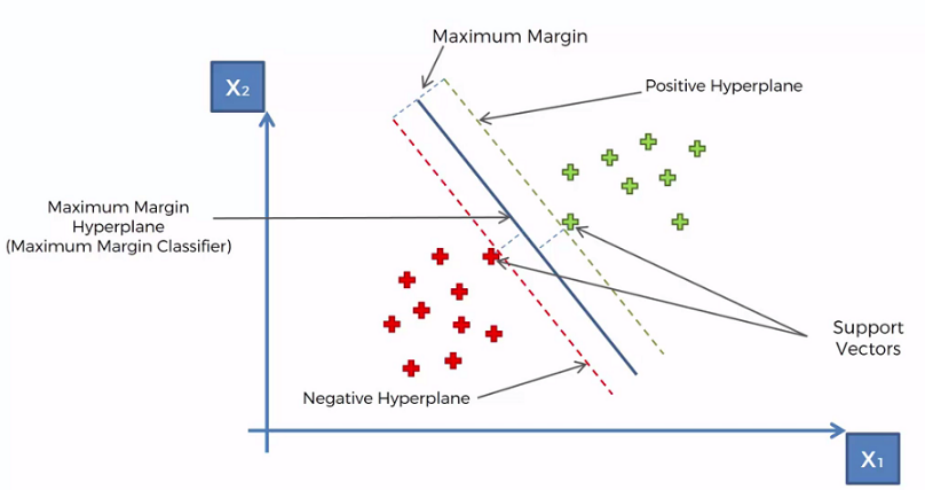

Figure 3.2: Support Vector Machine [5]

3.1.2.6 Random Forest

Random forest is one of the powerful supervised learning algorithms which is tree-based. It consists of the ensemble of decision trees used for predicting the label of the class in which data points belong to, based on categorical dependent variables [20]. These decision trees are usually trained by the bagging method.

Bootstrap Aggregation (Bagging) is a powerful ensemble method that makes more accurate predictions by combining the predictions from multiple machine learning techniques. This method reduces the variance in high variance algorithms such as decision trees.

In random forest, each tree determines a class label for an input by voting for a particular label. The class label with the maximum number of votes is considered to be the class label of the given input [20]. The classification is performed on the root initially and goes in a downward direction to the leaf node. If the predictions from the trees are weakly correlated the error rate of this classifier will be decreased. For this reason, trees should be as less associative as possible to minimize the error rate. In a decision tree, the nodes are represented as the features while outgoing edges are shown as tests on weights of features and the class categories represent the leaves [20]. As the name indicates, the process of splitting the feature nodes in the random forest is done randomly and it searches for the best feature in a random subset of features.

3.1.2.7 K-Nearest Neighbours

K-nearest-neighbour (KNN) is one the simplest classification algorithms which is very useful when there is no prior knowledge about the distribution of the data [30]. As it does not make any assumption on data distribution it is called a non-parametric technique.

In KNN the training data set is searched for K most similar samples which are called nearest neighbours and the output of these K nearest neighbours is summarized which is typically the most common value. The similarity of the K instances is determined based on a distance measure. The Euclidean distance is the most common distance measure used which is shown as,

$$Euclidean - Distance(x, x_i) = \sqrt{\sum (x_j - x_{ij})^2}$$

Where,

x is a new point.

x_i is an existing point across all input attributes j [30].

3.1.2.8 Neural Network (NN)

Neural networks are one of the linear classifiers and supervised learning techniques used in machine learning, however, the learning process in neural networks could be also unsupervised. Artificial neural networks (ANNs) are modeled on biological neural networks and try to solve the problems in a way that human brain does. The basic unit of the neural network is the neuron, therefore, neural network consists of many artificial neurons which are correlated to each other via synapses [15].

Neurons take the input data and after performing some calculations on them, pass the output to another neuron. As synapses which connect the neuron are weighted values, each input is multiplied by a weight. The sum of all input values multiplied by their weights plus a bias value are inputs for an activation function which defines the output of a neuron.

Activation Function controls whether a node should be active or inactive and decides whether to fire a neuron or not. The most popular activation functions are Sigmoid function, TanH function and ReLU function. Activation function performs a non-linear transformation on the input in order to perform more complex tasks. Typically the output of a neuron could be represented as [31],

$$y(k) = F(\sum_{i=0}^{m}(w_i(k) * x_i(k)) + b)$$

Where,

$x_i(k)$ is the input value in discrete time k in which $0 \leq i \leq m$.

$w_i(k)$ is the weight value in discrete time k in which $0 \leq i \leq m$.

b is the bias.

F is the activation function which is also known as the transfer function.

$y(k)$ is the output value in discrete time k.

The step function is a binary function which depending on whether the input value meets a particular threshold or not, gives two outputs. As shown in Figure 3.3, if the threshold is reached the output is 1 otherwise the output will be 0 [31].

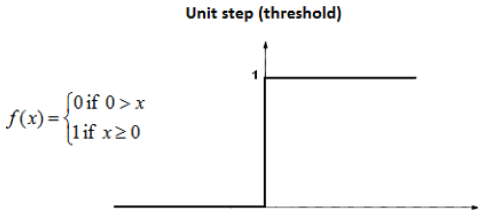

Figure 3.3: Step function [6]

Sigmoid function on the other hand, can output the results also between 0 and 1. This S-shaped curve that is also called logistic function is commonly used. TanH function or Hyperbolic Tangent function like sigmoid function is S-shaped but outputs the values from -1 to 1. This function is widely used in binary classifications [32]. The ReLU or Rectified Linear Unit activation function is one of the most common functions used nowadays since it is widely used in convolutional neural networks. ReLU outputs the result as 0 when the input is less than 0 while for values equal or greater than 0 the output value for an input value x would be $max(0, x)$ which is a linear function [33]. Sigmoid, tanH and ReLU are shown in Figure 3.4

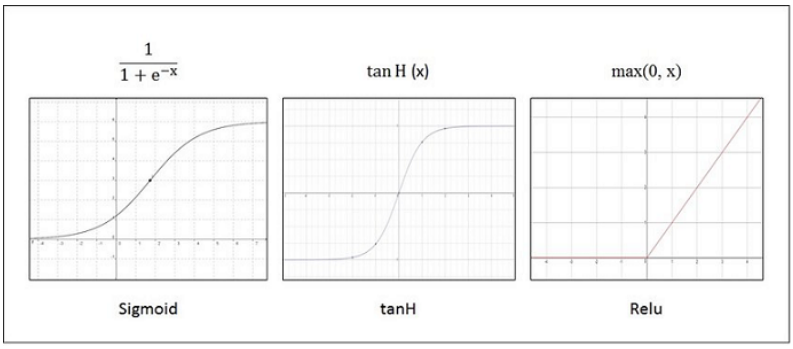

Figure 3.4: Different activation functions and their equations [7]

As the Figure 3.5 indicates an artificial neural network consists of layers which are divided into three types of the input layer, hidden layers and output layer [34].

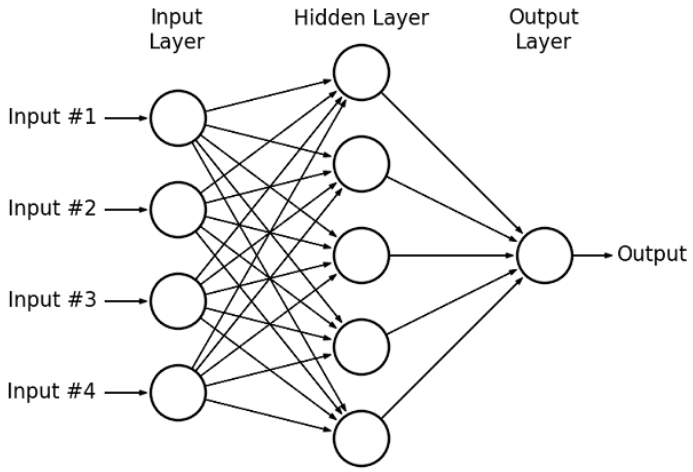

Figure 3.5: Basic model of an artificial neural network [8]

The input layer which is the leftmost layer gets the input data from external environment while the output layer is the rightmost layer and send the output to the external environment. The layers between input and output layers are called hidden layers or intermediate layers. These layers perform most of the internal processing [34]. They receive the input from input layer or previous hidden layers and send the output to the output layer or next hidden layers. These layers are called hidden because they could not be visible from the outside and they are not directly connected to the external environment [35].

Feed-forward artificial neural network is a type of neural network in which the flow of information is only in one direction which is from input towards output [36]. In the feed-forward neural network, there is no feedback from outputs to the inputs [37]. These type of neural networks are divided into two categories of single-layer and multi-layer. The simplest form of a feed-forward neural network is called perceptron which consists of a single neuron and is used for two-classification problems.

Recurrent artificial neural network In contrast to feed-forward neural network, in this neural network the flow of information is not only in one direction and in addition to the input to output, it could be also from outputs to inputs direction. In these neural networks, there is a feedback from outputs towards the inputs [37]. Recurrent neural network has an internal memory which makes it able to remember what it has learned recently, therefore for each input it will consider also the previous inputs. This feature which is not available in feed-forward neural networks could help recurrent neural networks to make better predictions. Figure 3.6 indicates the information flow in these two networks.

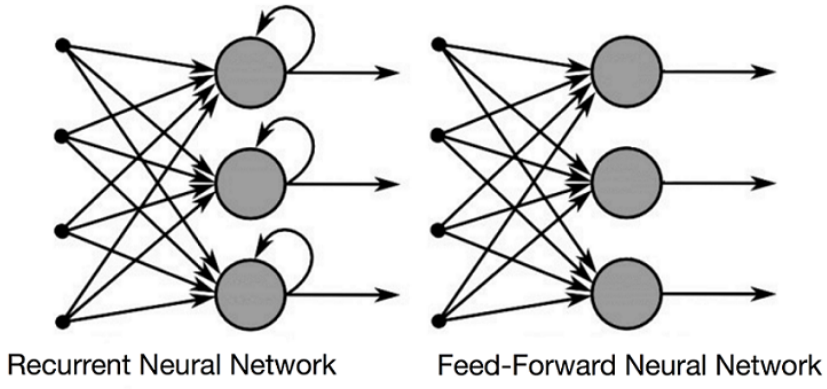

Figure 3.6: Information flow in feed-forward vs recurrent neural network [9]

Learning in artificial neural network One of the most important features of neural networks is the learning ability. In this process, a set of steps are performed for tuning the weighted values in order to map the inputs to the correct outputs [34]. In supervised learning, a set of inputs and their corresponding outputs are given in a training data set and the objective is to set the parameter values for any given pair of input and output [31]. In neural networks the learning process could be also unsupervised in which there is no given output but the neural network should search for some patterns on the given input

3.1.3 Hybrid Approaches

In these techniques, the combination of machine learning approaches and lexicon based approaches is used. According to some researches, in hybrid methods the performance of the classification is improved and reaches to better predictions [21].

3.1.4 Features Extraction

In order to help the machine learning algorithm to learn, text documents could be represented as features which are the properties that describe the data and show its pattern [38] [27]. This is the feature-based data representation in which different features can be used such as n-grams, Part-of-Speech (POS) tags, term frequency and semantic features [27]. Set of all the features can be described as a feature vector which is represented typically as numeric values.

3.1.4.1 Bag-of-Words

Bag-of-Words is a way of feature extraction from documents in which the words of a text and their number of occurrence in the text are represented. In this simple model, the number of occurrence of each word is considered as a feature [39]. In Bag-of-Words a vocabulary of unique words of the document is created and from this vocabulary, the document vectors are designed. for each text in the document, document vector contains 0s and 1s representing presence or absence of the vocabulary words in it.

By increasing the size of vocabulary, the size of vectors get bigger too and this may cause a large amount of 0s in each vector as each vector may only contain few numbers of vocabulary words. These vectors which are called sparse vectors increase the memory usage and may need more computations that make this model more challenging. In order to solve this issue, some techniques of text cleaning can be used. These techniques which are also known as feature selection include stop words removal, punctuation removal, misspellings correction, case normalizing and stemming.

Stop words removal is a pre-processing technique used in Natural Language Processing [40]. Stop words are usually most frequent short words that do not affect the overall meaning of the phrase and do not have any special information [41]. "The", "a", "is", "of" and "to" are some examples of stop words.

Stemming is another pre-processing method in which each word is reduced to its stem or its root form [42]. The objective of this process is to concentrate on the sentiment of a text rather than its meaning. For instance, the words "fisher", "fishing" and "fished" are reduced to their root which is "fish". This is done by stemming algorithms like Porter stemming algorithm.

3.1.4.2 N-grams

An n-gram is a language model which is probabilistic and given a sequence of words, its task is to predict the next word by means of Markov models [43].

N-gram model is the approach in which the vocabulary is consists of grouped words. Unigram is the n-gram with the size of one while n-gram with the size of two or two-words sequence of words is called bigram and n-gram with the size of three or three-word sequence of the word is called trigram [44]. Often using bigrams provides better performance than Bag-of-Words [45].

3.1.4.3 Term Frequency and TF-IDF

To score the word occurrence in documents, in addition to word presence which used in Bag-of-words, the word frequency and word can also be used. However, using word frequency may cause some high-frequency words to influence the scores while these words do not contribute to better sentiment extraction. To solve this problem, another method of scoring could be used which is Term Frequency-Inverse Document Frequency (TF-IDF). This technique is a statistical scoring to the importance of the words and to how rare they are [46]. The Inverse Document Frequency is high for rare words while for frequent words it is low.

3.1.4.4 Part-of-Speech tagging

Part-of-Speech (POS) tagging is one of the Natural Language Processing techniques in which a morphological tag such as verb, noun, adjective, and adverb is assigned to each word [47]. This model helps to reduce ambiguity in sentiment analysis as some tags like adjectives and adverbs are sentiment indicators [38].

3.2 Related Work

There are numbers of works done on sentiment analysis of Twitter data which in most of them the objective is to classify the tweets into positive and negative classes. Also in some classifications in addition to positive and negative, there is a neutral class.

Pang, Lee and Vaithyanathan [29] in 2002, had a survey to classify documents by sentiment instead of topics. They applied machine learning algorithms like Naive Bayes, Maximum Entropy and Support Vector Machine using different features to classify the sentiments of a movie reviews dataset. The dataset which consists of 1301 positive and 752 negative reviews is made using IMDB archive of movie reviews. They realized that traditional topic-based classifications output better result than sentiment-based classifications [29] and that Support Vector Machine provides the best result. Using the unigrams presence gave them the best performance among other features like part-of-speech and bigrams [29]. Pang et al. discussed that in contrast to standard text classifications, the term presence is better than term frequency in sentiment analysis [38].

In 2009, Go et al. [48] collected the Twitter data by querying sad and happy emoticons. They used smiley faces such as ":)", ":-)", ":)", ":D", "=)", ":(", ":-(", ":(" to get the tweets. Unigrams, bigrams, unigrams and bigrams, and Part of Speech were the different features they have used to train their classifier. The machine learning-based algorithms used were Naive Bayes, Maximum Entropy and Support Vector Machine [48]. They realized that as the feature space in bigrams is sparse, using only bigrams as features is not useful and the combination of unigrams and bigrams gives the better result. They also found out that using part-of-speech tags as features were not helpful [48].

In 2010, Pak and Paroubek [49] collected a corpus from the Twitter which is categorized into three sets of positive, negative and objective texts. Also in this work positivity indicates joy and happiness while negativity shows sadness. Objective texts are those texts without sentiments. Their approach

to collect positive and negative texts was the same as Go et al. [48] by using emoticons however for objective texts, they queried the account of famous newspapers [49]. They used multinomial Naive Bayes and Support Vector Machine algorithms to make their classifiers on features like n-grams and part-of-speech tags. Using bigrams provided the best performance [49].

In 2010 also Barbosa and Feng [50] suggested a sentiment analysis approach on Twitter data which consists of two steps. In the first phase, the tweets are classified as subjective and objective. If the text is objective it does not have any sentiment. In the second phase, the subjective tweets are classified into positive and negative classes [50]. They used part-of-speech, prior subjectivity and polarity as meta-features and using Support Vector Machine they get the best result. Davidov et al. [51] also in 2010 used K-nearest-neighbours algorithm to classify the Twitter data utilizing smileys and hashtags, however, their classification was to sentiment and non-sentiment classes [52].

Saif et al. [53] in 2012 discussed a semantic-based approach for sentiment analysis on tweets collected from three datasets. In this approach, the semantic concepts of each entity were added as an additional feature [53]. The Twitter datasets used in this work are Stanford Twitter Sentiment Corpus, Health Care Reform, and Obama-McCain Debate. In this work, Naive Bayes classifier is used to identify positive and negative sentiments. classification is done by using unigrams, part-of-speech and semantic features. Semantic features outperform the unigrams and part-of-speech in all three datasets [53].

Chapter 4

Data Collection

In this chapter, the process of collecting the dataset from the Twitter API is discussed, then the collected dataset is considered and the data labeling is done on the dataset.

4.1 Twitter API

API stands for Application Programming Interface which is a tool that provides an easy interaction with web services. Twitter provides its Streaming API to developers to interact with its service and access public data in real-time, programmatically. Twitter also supports REST API in which there is rate limitation and it is not possible to download more than a specific amount of data [38]. REST API provides only short-lived connections while Streaming supports connections with long intervals. In this work, as a large amount of data is needed to be collected without rate limitation and in long-lived connections, Twitter Streaming API is used.

4.1.1 Obtaining the Twitter API Credentials

In order to have access to Twitter data, an application is created to interact with Twitter API. This application enables the access to the API keys which are tokens that can be used to make a request to the API. consumer key, consumer secret, access token and access token secret are tokens that can be used to make a request to the API. Consumer key and consumer secret are used for authenticating the application while access token and access token secret are used to authenticate the user.

4.1.2 Connecting to the Twitter API

The open source Python library called Tweepy is used to access Twitter Streaming API, Tweepy is one of the Python libraries that simplifies the interaction between Python and Twitter API. By connecting to the API through Tweepy, the Information of the tweets are extracted. As the Figure 4.1 shows, Tweet Id which is a unique Id assigned by Twitter to the tweet , the text of the tweet, the language of the tweet, retweet count of the tweet, the time in which tweet is sent, user Id, user name, screen name of the user, user's description, user's followers count, user's following count, user's location, list of the hashtags in the tweet and the URL entities in the tweet are some of the main important fields were extracted.

4.2 Dataset

In this work, the dataset is created as there was no available public dataset about tweets on Climate Change to be used. The process consists of two phases in which the first phase is collecting the tweets as the dataset and the second phase is labeling the collected dataset in order to be divided into training and testing datasets.

Field	Value
Id	859933953121288000
Text	RT @LeoDiCaprio: #ClimateChange is real. Scientists agree. Get the facts. #climatefacts https://t.co/OsSxAJ6cdV
Language	en
CreatedAt	04/05/2017 02:53:25
RetweetCount	11804
UserId	2278849278
UserName	Lady_Cross
UserScreenName	Dlc40458
UserDescription	Mother and Grandmother, advocate for saving the environment and animal lover.
UserFollowersCount	764
UserFollowingCount	1620
UserLocation	Ohio, USA
HashtagEntities	[ClimateChange, climatefacts]
URLEntities	[http://bit.ly/2kuiN5j]

Figure 4.1: Properties extracted from a tweet

4.2.1 Dataset Creation

The total number of 2,200,000 Climate Change related English tweets were extracted from May 2017 to September 2017. These tweets are extracted by using the most popular climate change hashtags in the mentioned period, including:

- **#climatechange**
- **#climate**
- **#globalwarming**
- **#climateaction**
- **#actonclimate**
- **#keepitintheground**
- **#environment**
- **#climatehoax**

- **#parisagreement**
- **#climatechangeisreal**

The dataset is collected based on each of the hashtags mentioned above, therefore it may contain duplicates. For instance when tweets are collected based on **#climatechange**, a tweet may contains multiple hashtags such as **#climatechange**, **#globalwarming** and **#climateaction**. For this reason, this tweet is gotten also two more times when the tweets are being collected by **#globalwarming** and **#climateaction**. As duplicate tweets are not useful in the analysis performed in this work, they are removed. After removing the duplicates the dataset contains 1,500,000 tweets. The extracted tweets are stored as CSV files and ready for labeling.

4.2.2 Data Labeling

In sentiment analysis, data should be labeled so that machine learning algorithms can apply their models on these labeled data and make predictions for unlabeled data. In this step, a sentiment should be assigned to each tweet of the dataset. The assigned sentiment is called label and represents whether the sentiment of the tweet is positive, negative or neutral.

In this thesis, The sentiments which are considered are positive and negative. Positive represents that the tweet is about believing in climate change and that the person who tweeted believes that climate change is happening and it exists. While negative indicates that tweet is about climate change denial or climate change skepticism and that the person who tweeted does not believe in climate change and denies its existence. It may also show that the user does not think that human activity causes climate change and that actions should be taken to stop it. Neutral is not considered in this work as it does not help in these analyses and does not make an important role in the classifications considered in the climate change context. To sum up, the classification in this work is a two-class classification in which classes are:

- **Positive:** If the tweet indicates an opinion which supports climate change action and shows a belief in this phenomenon and that human role in this fact is very effective.
- **Negative:** If the tweet indicates an opinion against the existence of climate change, supports climate change denial or climate change skepticism, and shows an opposition to the belief about impacts of human activity on climate change.

Labeling the most part of the dataset created is done manually. As the dataset is large, to speed up the labeling process, the positive and negative influencers are considered first. This is done by looking at the users with the most number of tweets and retweets and considering their Twitter profile to get their opinion on climate change. In this way by querying the database for tweets of these top influencers using their User Id, a part of the dataset is labeled more quickly. The total number of 120,000 tweets are labeled in this thesis.

Figure 4.2 shows a few samples of the labeled tweets.

Text	Sentiment
Today, our planet suffered. It's more important than ever to take action. #ParisAgreement https://t.co/FSVYRDcGUH	Positive
How Trump's decision to leave the #ParisAgreement hurts America, in 5 graphics. #ActOnClimate https://t.co/s5uoLdnJS3	Positive
#MakeOurPlanetGreatAgain ??Researchers, Teachers, Entrepreneurs: Join France in its fight against #GlobalWarming >... https://t.co/AW3W9QUYPS	Positive
#GreatBarrierReef is the jewel we need to protect - not the #coal industry #ActOnClimate @billshortenmp https://t.co/hjaL7ezYwu	Positive
When we protect our lands and waters, it helps us protect our climate for future generations. #ActOnClimate https://t.co/8tygZI0TDR	Positive
Addressing climate change is one of my top priorities. We need to take action for our kids. #ActOnClimate	Positive
Little people cause Global Warming #FakeGlobalWarmingFacts https://t.co/f1QSPSgGnc	Negative
Climate change? You mean like the weather?- Donald Trump #FakeGlobalWarmingFacts https://t.co/dabf16BA7v	Negative
There's no reason to #ActOnClimate change, it is a hoax! #climatechange #climatehoax https://t.co/5cwYJ7YVka	Negative
#ClimateChange is a hoax #ParisAccord #FridayFeeling https://t.co/1ekTuUvrT1	Negative
#FakeGlobalWarmingFacts You can cool down the earth by pouring ice all over it.	Negative
#ClimateChange is a hoax that costs America BILLIONS of $ per day! #climatehoax #ParisClimateDeal	Negative

Figure 4.2: Labeled tweets sample

Chapter 5

Implementation and Results

In this chapter, supervised machine learning approaches are used for sentiment classification. In machine learning techniques the training data is used for learning and the model is applied to the testing data, therefore the dataset should be split into the training dataset and testing dataset.

Before splitting the dataset, the retweets are omitted because they do not affect the training process. The dataset consists of retweets which are reposts of the tweets by other users. This is not the same as duplicates removal discussed in the subsection 4.2.1 because the duplicate is the same tweet posted by a particular user and it is completely useless in this study while retweet is a forwarded tweet by another user. Retweets are not helpful in the process of sentiment classification because having retweets in the training dataset, the classifier would be trained by repeating data and will not have a good performance as the training dataset would be very small. However, retweets are helpful in some cases such as analysis on the location of the users and therefore they are not removed at initial steps like data duplicates but for creating the training and testing dataset the retweets are not considered.

120,000 tweets are labeled that include also the retweets. After removing the retweets for classification 53,468 tweets are considered as the dataset to be used in classification. 80% of the dataset is considered as the training dataset which consists of 43,996 tweets while the rest is for the testing dataset. 30%

of the training dataset is used as validation set. The objective is to compare the performance of different algorithms on validation set and keep the testing dataset unseen until the final assessment of the performance.

5.1 Data Preprocessing

The tweets available in the dataset are not suitable for feature extraction since they may contain irrelevant information which should be cleaned [54]. This process which is a text normalization approach is called data preprocessing and helps in improving the performance of the classification by decreasing the noises of the text and size of the feature sets of the tweets. Data preprocessing consists of:

- **Lowercasing**
- **Replacing Elongated Words**
- **Replacing URLs**
- **Replacing Mentions**
- **Replacing Hashtags**
- **Removing non-alphanumeric**
- **Removing stop words**
- **Stemming**
- **Replacing numbers**
- **Tokenization**

5.1.1 Lowercasing

In this step, all the uppercase words are converted into lowercase. This is one of the important techniques in data preprocessing as it reduces the dimensionality of the problem and provides a consistent form of the tweets [55]. The method **lower()** which is a built-in method of Python for string modification is used for lowercasing.

5.1.2 Replacing Elongated Words

This is the process by which elongated words that are the words that contain characters repeating more than 2 times, are reduced to their standard form. Elongated words are used to express more feelings and emphasize on that word and they do not have a different meaning, therefore using them makes classifier think they are different words and it may decrease the performance [56].

5.1.3 Replacing URLs, Mentions and Hashtags

Tweets may contain URLs which are links to other pages or websites. The URL itself does not contain useful information for sentiment classification, for this reason, they could be replaced. The **sub()** method of the regular expression module of Python called **re** is used as **re.sub()** to replace URLs with the tag < **url** >.

There may be typically a number of mentions in a tweet. Mentions which are in the form of **@username** are used to refer or reply to a user. As mentions also do not have sentiment, they are replaced with the tag < **user** >. Hashtags may contain information that is related to the subject, for this reason they are used for feature extraction that will be explained in the next sections. However, for preprocessing the symbol of the hashtag can be removed and the word is replaced as the tag < **hashtag** >.

For instance the tweet:

"#GreatBarrierReef is the jewel we need to protect - not the coal industry #ActOnClimate @billshortenmp https://t.co/hjaL7ezYwu"

After replacing the URLs, mentions and hashtags would be as:

"< hashtag > is the jewel we need to protect - not the coal industry < hashtag > < user > < url >"

5.1.4 Removing Non-alphanumeric

In this step, Non-alphanumeric characters are removed since they do not have sentiment. $=, -, *,), (, \{, \}, \%, \&, \#, @, <>, +, /$ are some of the non-alphanumeric characters. As mentioned earlier, # and @ are used in replacements done before, therefore it is important to perform this step after mentions and hashtags replacement. Also, <> is not considered to be removed since it is used as the tag in previous steps, therefore it is not taken as a non-alphanumeric character to be removed in this thesis.

5.1.5 Removing Stop Words

In stop words removal most frequent short words which do not have any effect on the overall meaning of the phrase and do not convey any particular information, are removed [41]. "The", "a", "an", "is", "of" and "to", "he", "she", "on", etc are examples of stop words. Stop words are removed using **re.sub()** to replace words containing only 1 or 2 characters with empty space.

5.1.6 Stemming

In stemming each word is reduced to its stem or its root form with the purpose of focusing on the sentiment of the word rather than its meaning [42]. This process helps to reduce complex grammatical transformations and

also dimensionality of the text [57]. Porter stemming algorithm is used for stemming.

5.1.7 Replacing Numbers

In this stage, Numbers are replaced with the tag < **number** > since they do not contain sentiment and does not affect the sentiment classification.

5.1.8 Tokenization

In this step, a text is split into smaller word-like pieces called tokens. Tokenization breaks the tweet into the token using **split()** method of regular expression **re** module of the Python. The separation can be done by the whitespace character, comma or other punctuation marks [57].

5.2 Feature Extraction

In this phase, the properties of the data that describe it, are extracted in order to be used in classification. These properties called features make the tweets be represented as characteristics that are discriminative since the whole input data may be too large for classification [38]. The features reduce the dimensionality of the input data and could help in redundancy prevention [58].

The features used in this thesis are:

- **N-grams**
  - Term Frequency
- **Sentiment Lexicons**
  - POS Tag
  - Summations over the sentiment scores
  - Total number of positive and negative words regarding the lexicon
- **Word Embeddings**
  - Summations over the word embedding vectors of d dimensions
- **Twitter features**
  - Tweet Length
  - URLs
  - Mentions
  - Uppercase Words
  - Negation words
  - Elongated Words
  - Hashtags

5.2.1 N-grams

In the n-gram model, given a sequence of words, the next word is predicted. This probabilistic language model uses Markov models to perform predictions [43]. n-gram can be categorized into unigram, bigram, and trigram based on the number of grouped words. In this work, unigrams and bigrams are used for features extraction.

In this thesis, **Scikit-learn** library which is a machine learning library for Python is used.

By means of the module **sklearn.feature_extraction.text.CountVectorizer** the text of the tweets are converted to their tokens count.

A small sample of training dataset and testing dataset are represented in Figure 5.1 and Figure 5.2. Figure 5.1 indicates text of 6 tweets, the hashtags inside each tweet and the sentiment or the label assigned to them as training data. Figure 5.1 consists of 2 tweets considered as testing data, their including hashtags and the predicted labels for them.

Text	Hashtag Entities	Sentiment
The founder of the #WeatherNetwork says there is no evidence of #GlobalWarming and if he thought there was he would... https://t.co/JNNHK7HZ3y	[WeatherNetwork, GlobalWarming]	Negative
There's no reason to #ActOnClimate change, it is a hoax! #climatechange #climatehoax https://t.co/5cwYJ7YVka	[ParisAgreement]	Negative
Today, our planet suffered. It's more important than ever to take action. #ParisAgreement https://t.co/FSVYRDcGUH	[parisagreement]	Positive
#ClimateChangeIsReal & now Trump has backed out of the #ParisAgreement look for real leaders I support the United... https://t.co/SjxOzt6j9P	[ParisAgreement]	Positive
#FakeGlobalWarmingFacts paying your lord and savior big government a tax will prevent the climate from changing.....NOT	[ClimateChangeIsReal, ParisAgreement]	Positive
#ClimateChange is a hoax that costs America BILLIONS of $ per day! #climatehoax #ParisClimateDeal	[FakeGlobalWarmingFacts]	Negative

Figure 5.1: Sample of training data

Text	Hashtag Entities	Predicted Sentiment
CO2 Is Beneficial To Your Skin #FakeGlobalWarmingFacts @HashFakeFacts	[FakeGlobalWarmingFacts]	Negative
Giant Antarctic iceberg 'hanging by a thread', say scientists. #ActOnClimate #ClimateChange #KeepItInTheGround https://t.co/Xxyfo8M1yx	[ActOnClimate, ClimateChange, KeepItInTheGround]	Positive

Figure 5.2: Sample of testing data

CountVectorizer is one of the modules of **Scikit-learn** library which converts the text into the matrix of tokens occurrence count. One of the parameters of the **CountVectorizer** is **ngram_range** which reveals the lower and upper bound of n-grams values.

- $ngram\_range = (1, 1)$, returns the unigrams count
- $ngram\_range = (1, 2)$, returns both unigrams and bigrams count

The vocabulary of the dataset consists of words and their feature indices. The **Vocabulary** attribute of the module **CountVectorizer** is used to create the vocabulary shown in the Figure 5.3.

```
{'the': 62, 'founder': 21, 'of': 42, 'weathernetwork': 70, 'says':
54, 'there': 63, 'is': 32, 'no': 39, 'evidence': 18, 'globalwarming'
: 24, 'and': 4, 'if': 30, 'he': 27, 'thought': 64, 'was': 69, 'would
': 72, 'https': 29, 'co': 14, 'jnnhk7hz3y': 34, 'reason': 52, 'to':
65, 'actonclimate': 2, 'change': 8, 'it': 33, 'hoax': 28, 'climatech
ange': 11, 'climatehoax': 13, '5cwyj7yvka': 0, 'today': 66, 'our':
43, 'planet': 49, 'suffered': 56, 'more': 38, 'important': 31, 'than
': 60, 'ever': 17, 'take': 58, 'action': 1, 'parisagreement': 45, 'f
svyrdcguh': 23, 'climatechangeisreal': 12, 'now': 41, 'trump': 67, '
has': 26, 'backed': 5, 'out': 44, 'look': 36, 'for': 20, 'real': 51,
'leaders': 35, 'support': 57, 'united': 68, 'sjxozt6j9p': 55, 'fakeg
lobalwarmingfacts': 19, 'paying': 47, 'your': 73, 'lord': 37, 'savio
r': 53, 'big': 6, 'government': 25, 'tax': 59, 'will': 71, 'prevent'
: 50, 'climate': 10, 'from': 22, 'changing': 9, 'not': 40, 'that':
61, 'costs': 15, 'america': 3, 'billions': 7, 'per': 48, 'day': 16,
'parisclimatedeal': 46}
```


Figure 5.3: Vocabulary of the dataset

By mean of the method **fit_transform()** of the **CountVectorizer** module, the data will be tokenized and their occurrence count will be represented in the form of a matrix. This method returns the term frequency of the words in the dataset. Figure 5.4 shows this matrix.


```
[[0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 1 0 0 1 0 0 2 0 1 1 0 1 0 1 0
  0 0 0 1 0 0 2 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 2 2 1 0 0 0 0 1 1 0
  1 0]
 [1 0 1 0 0 0 0 0 1 0 0 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 1 0 0
  0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0
  0 0]
 [0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 1 0 0 0 0 0 1 0 1 0 1 0 0
  0 0 1 0 0 0 0 1 0 1 0 0 0 1 0 0 0 0 0 0 1 0 1 0 1 0 0 0 0 1 1 0 0 0 0 0
  0 0]
 [0 0 0 0 0 1 0 0 0 0 0 0 1 0 1 0 0 0 0 0 1 0 0 0 0 0 1 0 0 1 0 0 0 0 0 1
  1 0 0 0 0 1 1 0 1 1 0 0 0 0 0 1 0 0 0 1 0 1 0 0 0 0 2 0 0 0 0 1 1 0 0 0
  0 0]
 [0 0 0 0 1 0 1 0 0 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0
  0 1 0 0 1 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 1
  0 1]
 [0 0 0 1 0 0 0 1 0 0 0 1 0 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0
  0 0 0 0 0 0 1 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
  0 0]]
```


Figure 5.4: Matrix of training data indicating number of tokens of each tweet

The matrix shown in Figure 5.4, is a sparse matrix of 6×74, in which 6 is the number of rows and is equal to the number of tweets in training dataset and 74 is the number of words (features) in the vocabulary created from the dataset. Each element shows the number of occurrence of the corresponding word to the index of the vocabulary, in the tweet. This matrix represents the unigrams count.

The sparse matrix of the testing data mentioned above is shown in Figure 5.5. This is a matrix of 2×74 since there are 2 tweets in the testing dataset and 74 words in the vocabulary. This matrix represents the bigrams count.

```
[[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0
  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
  0 1]
 [0 0 1 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
  0 0]]
```

Figure 5.5: Matrix of testing data indicating number of tokens of each tweet

5.2.2 Sentiment Lexicons

SentiWordNet that is a lexical resource used for sentiment and opinion mining, assigns to each word of the tweet three scores, the positivity score, negativity score, and objectivity score. Using the **SentiWordNet** interface of the **nltk.corpus** package and its **senti_synsets()** method, the scores of each token in the tweet is calculated. Also the sentiment scores of the POS tags are calculated by means of **senti_synsets**.

By means of **NLTK** which is a leading library for Python programs to play with natural language, and using the **pos_tag()** method the Part-of-Speech tags of the words are returned. The Part-of-Speech (POS) tag is morphological tag such as verb, noun, adjective, and adverb that is assigned to each word in order to reduce ambiguity in the processing of the natural language [47].

In next step, the summation of the positivity score for each word and the positivity score of its POS tag is calculated to generate an overall positivity score for each tweet. In the same way, the overall negativity score is also derived.

A list of sentiment lexicon containing 6800 positive and negative words, compiled by Hu and Liu [59] is used to prepare two sets of positive and negative words. This opinion lexicon is used to get the number of positive and negative words. In general in this step, the summation of the sentiment scores and the total number of positive and negative words are used as features.

5.2.3 Word Embeddings

Word embedding is a kind of word representation in the form of numeric vectors. This method which is a feature learning technique, is used because typically machine learning techniques are not capable of processing text and the raw form of natural language, but they need numeric values to be able to do the processing [60]. It can be done by means of different approaches such

as CountVectorizing (one-hot encoding), TF-IDF transforming, Word2Vec, and GloVe.

In general in word embeddings, each word of the tweet is mapped to a vector. In this work GloVe (Global Vectors for Word Representation) is used to perform this mapping. Glove tries to combine global matrix factorization techniques [61] and local context window [62]. The training of the GloVe is on the counts of the global co-occurrence of the words by creating the word co-occurrence matrix. The co-occurrence of two words is the number of times they appeared with one another. Considering the two words w_i and w_j, each element of the co-occurrence matrix is number of times the word w_i is occurred in the context of the word w_j. The intuition of the GloVe training is that ratio of the co-occurrence probabilities of the words helps in capturing the word meanings rather than the probabilities[63].

The co-occurrence of probability of a pair of words is calculated as [63],

$$P_{co}(w_j \mid w_i) = \frac{C(w_i, w_j)}{w_i}$$

Where,

w_i and w_j are the words.

$C(w_i, w_j)$ is the co-occurrence of the words w_i and w_j.

While the ratio of the co-occurrence probabilities depends on the three words and can be represented as [63],

$$F(w_i, w_j, w_k) = \frac{P_{co}(\widetilde{w_k} \mid w_i)}{P_{co}(\widetilde{w_k} \mid w_j)}$$

Where,

w_i and w_j are the words that output word vector w.

w_k is the word in separate context, for example it is related to w_i but not related to w_j.

The GloVe pre-trained word vector file for Twitter which consists of 1.2 million word embeddings of 25, 50, 100 and 200 dimension vectors, is used in this work. First, an empty embedding of 200 dimensions is created and then for each word in the tweet if the word is in the GloVe, its embedding is added to the empty embedding created. When all of the words in a tweet are processed, the summation of all the embeddings are calculated for each tweet. In this phase, summations over D dimensions generate D features for each tweet.

5.2.4 Twitter Features

There are other features that are Twitter-specific. The length of the tweets, the presence of the URL in the tweet, User mention, negation, number of emoticons, number of exclamation words, number of uppercase words, the number of elongated words, presence and the number of hashtags are the Twitter features considered in this study.

Binary vectors are used to extract feature for the presence of URLs. if there is a URL in the tweet, 1 is appended to the feature vector otherwise 0 is appended. The same behavior is treated for mentions, negation words such as 'do not', 'does not', 'is not' and 'are not'. However, for the uppercase words, emoticons, exclamation and elongated words, the number of the words are considered not the presence of them.

The hashtag handling in this work is not just by the presence of the hashtags but also by both presence and number of the hashtags. A list of positive and negative hashtags from the dataset is created and for each tweet, the number of positive and negative hashtags are calculated. The labeling of the hashtags is an important step since some of the hashtags may not contain a specific sentiment in this context. For example, the **#climatechange** that is one of the most frequent hashtags in the dataset collected, does not reveal a certain sentiment as it could be used by both climate change believers and climate change deniers and therefore can be labeled as both positive and negative in different tweets. Also the frequent hashtags such as **#globalwarming**,

#climate and **#parisAgreement** could not be labeled specifically as positive or negative in a tweet, because it depends on the other words of the tweet.

However, there are some hashtags that could be considered as positive or negative since they reveal more sentiment and they are typically used by a specific group of users. These hashtags are used in the feature vector. The hashtags that are usually used for expressing the positive opinion in the classification of this work are:

- **#ClimateChangeIsReal**
- **#ActOnClimate**
- **#KeepItInTheGround**
- **#MakeOurPlanetGreatAgain**

while the hashtags that are usually posted by climate change deniers and reveal a negative opinion about the context of this work are:

- **#ClimateChangeHoax**
- **#ClimateHoax**
- **#FakeGlobalWarmingFacts**

5.3 Classification

The sentiment classification in this thesis is a two-class classification which categorizes the tweets into positive and negative classes. As described in the section 4.2.2, the positive class consists of tweets supporting that actions should be taken for climate change and that climate change is happening and does exist while negative class includes tweets that support climate change skepticism or denial.

In the context of climate change, the classification may be complex. In order to make it more clear, consider classifying a dataset such as movie reviews. For classifying the reviews on a movie, positive and negative words generally make an important role, therefore, it could be simpler to predict the label for a review as positive when there are some positive words in it, or as negative when there are some negative words in it. However in the climate change context, there could be a tweet which contains some negative words but the whole meaning of the sentence is against the climate change denial, so it is a positive tweet. For instance the tweet:

"Removing the United States from the #ParisAgreement is a reckless and indefensible action. https://t.co/gYaOAANgWa"

Contains the words **reckless** and **indefensible** which are negative words according to the opinion lexicon provided by Hu and Liu [59], however, the tweet is classified as positive since it is supporting the action that should be taken for climate change.

Therefore different features should be taken into account in this context to make a good classifier. In this section, various classification techniques are applied to the dataset, using different features. Then the performance metrics for each classifier are calculated in order to evaluate the classification. Performance metrics which are considered in this work include:

- **Accuracy** Accuracy which is one of the most common performance metrics, is the ratio of correct predictions to the total number of predictions made. It is important to notice that accuracy alone is not a good evaluation metric for performance since it works well when there is an equal number of observations in each class. Therefore other performance metrics should be also taken into account in order to have a good evaluation of the classifier. Accuracy could be calculated as,

$$Accuracy = \frac{T_p + T_n}{T_p + T_n + F_t + F_p}$$

  Where,

  T_p is the number of true positives (tweets that classifier correctly classified as positive) [64].

  F_p is the number of false positives (tweets that classifier incorrectly classified as positive).

  T_n is the number of true negatives (tweets that classifier correctly classified as negative).

  F_n is the number of false negatives (tweets that classifier incorrectly classified as negative).

- **Precision** Precision is defined as the number of true positives divided by the total number of predicted positives which includes true positives and false positives. Precision which shows the exactness of the classifier is calculated as,

$$Precision = \frac{T_p}{T_p + F_p}$$

Where,

T_p is the number of true positives (tweets that classifier correctly classified as positive).

F_p is the number of false positives (tweets that classifier incorrectly classified as positive).

- **Recall** Recall is number of the true positives divided by number of true positives and false negatives. This metric which shows the completeness of the classification is calculated as,

$$Recall = \frac{T_p}{T_p + F_n}$$

Where,

T_p is the number of true positives (tweets that classifier correctly classified as positive).

F_n is the number of false negatives (tweets that classifier incorrectly classified as negative).

- **F1-Score**

F1-score is a metric which combines precision and recall in order to show the performance of the classifier. F1-score which is the harmonic mean or weighted average of the precision and recall is calculated as,

$$F1 - Score = 2 \times \frac{P \times R}{P + R}$$

Where,

P is the precision.

R is the recall.

- **Confusion Matrix**

  Confusion matrix is another performance metric which represents the accuracy in form of a matrix. In this thesis that the classification is binary, the matrix consists of two columns and two rows in which rows represent the actual class values and the columns represent the predicted labels. Figure 5.6 indicates what confusion matrix reveals [65].

		Predicted	
		Negative	Positive
Actual	Negative	TN	FP
	Positive	FN	TP

TN = True Negative
FP = False Positive
FN = False Negative
TP =True Positive

Figure 5.6: Confusion matrix

Different machine learning algorithms used for classification, are:

- **Naive Bayes classifier**
  - Bernoulli Naive Bayes
  - Gaussian Naive Bayes
- **Logistic Regression**
- **Ridge Classifier**
- **Support Vector Machines**
  - SVC
  - Linear SVC
- **K-Nearest Neighbours (KNN)**
- **Decision Tree**
- **Neural Networks**
  - Multilayer Perceptron
- **Random Forest**

5.3.1 Naive Bayes Classifier

Naive Bayes classifier performs a supervised learning classification. This probabilistic model makes use of Bayes Theorem and as described in section 3.1.2.1 it could be shown as,

$$P(label \mid features) = \frac{P(features \mid label)\, P(label)}{P(features)}$$

Having the feature sets of the tweets, the goal is to find the probability that whether the tweet belongs to the positive class or negative class.

Using the *Maximum a posteriori* hypothesis as,

$$label_{MAP} = \arg\max_{y} P(label) \prod_{i=1}^{n} P(fn \mid label)$$

The sentiment or label of the tweet will be predicted by calculating the $P(label)$ called class probability and $P(fn \mid label)$ called conditional probability. Depending on the distribution of the $P(fn \mid label)$, Naive Bayes classifiers may differ.

5.3.1.1 Gaussian Naive Bayes

As explained in 3.1.2.1, this classifier extends the Naive Bayes to continuous data.

Using **Scikit-learn** library and the combination of the sentiment lexicons and word embeddings the accuracy would be **0.59** and the classifier results are as shown in Table 5.1

While considering the Twitter features in addition to the previous feature sets, the classifier would perform better as indicated in Table 5.2. The accuracy of would be **0.67** in this situation.

	Precision	**Recall**	**F1-Score**	**Support**
Negative (Class 0)	0.07	0.65	0.12	412
Positive (Class 1)	0.97	0.60	0.74	9060
Average/Total	0.93	0.60	0.71	9472

Table 5.1: Performance metrics of Gaussian Naive Bayes using sentiment lexicon + word embeddings

	Precision	**Recall**	**F1-Score**	**Support**
Negative (Class 0)	0.09	0.71	0.16	412
Positive (Class 1)	0.98	0.68	0.80	9060
Average/Total	0.94	0.68	0.77	9472

Table 5.2: Performance metrics of Gaussian Naive Bayes using sentiment lexicon + word embeddings + Twitter features

It is obvious that n-grams features play an important role in this classification, since using the combination of unigrams, sentiment lexicons, word embeddings and Twitter features as features the classifier would have better performance results. Gaussian Naive Bayes classifier produces its best results as shown in Table 5.3. This result is gotten

	Precision	**Recall**	**F1-Score**	**Support**
Negative (Class 0)	0.70	0.39	0.50	412
Positive (Class 1)	0.97	0.99	0.98	9060
Average/Total	0.96	0.97	0.96	9472

Table 5.3: Performance metrics evaluation for Gaussian Naive Bayes classifier

This classifier outputs the following accuracy:

Accuracy score = 0.96

The confusion matrix of Gaussian Naive Bayes classifier is indicated in Table 5.4.

	Negative	**Positive**
Negative	162	250
Positive	68	8992

Table 5.4: Confusion Matrix for Gaussian Naive Bayes classifier

As the confusion matrix shows, there are:

- 8992 true positive
- 250 false positive
- 162 true negative
- 68 false negative

5.3.1.2 Bernoulli Naive Bayes

The Bernoulli Naive Bayes classifier is performed on binary or boolean feature vectors. By using **Scikit-learn** package and training the dataset using n-grams plus sentiment lexicon and Twitter features as feature set, the performance metrics and confusion matrix of this classifier would be as shown in Tables 5.5 and 5.6.

	Precision	**Recall**	**F1-Score**	**Support**
Negative (Class 0)	0.77	0.46	0.58	412
Positive (Class 1)	0.98	0.99	0.98	9060
Average/Total	0.97	0.97	0.97	9472

Table 5.5: Performance metrics of Bernoulli Naive Bayes classifier for unigrams + sentiment lexicon + Twitter features

The accuracy of the classifier is:

Accuracy score = 0.97

	Negative	**Positive**
Negative	190	222
Positive	56	9004

Table 5.6: Confusion Matrix for Bernoulli Naive Bayes classifier using unigrams + sentiment lexicon + Twitter features

As the confusion matrix shows, there are:

- 9004 true positive
- 222 false positive
- 190 true negative
- 56 false negative

While concatenating all the feature sets and using them the classifier evaluation metrics will be as indicated in Table 5.7. In this the state the accuracy is **0.86**.

	Precision	**Recall**	**F1-Score**	**Support**
Negative (Class 0)	0.15	0.43	0.22	412
Positive (Class 1)	0.97	0.89	0.93	9060
Average/Total	0.94	0.87	0.90	9472

Table 5.7: Performance metrics evaluation for Bernoulli Naive Bayes classifier

The Bernoulli Naive Bayes classifier confusion matrix in this situation is indicated in Table 5.8.

	Negative	**Positive**
Negative	176	236
Positive	1024	8036

Table 5.8: Confusion Matrix for Bernoulli Naive Bayes classifier

5.3.2 Logistic Regression

This classifier that uses logistic function and map input values to an output between 0 and 1, using **Scikit-learn** library and sentiment lexicon, word embeddings and Twitter features as the feature sets gives an accuracy of **0.95** and performance results as shown in Table 5.9.

	Precision	**Recall**	**F1-Score**	**Support**
Negative (Class 0)	0.49	0.50	0.50	412
Positive (Class 1)	0.98	0.98	0.98	9060
Average/Total	0.96	0.96	0.96	9472

Table 5.9: Performance metrics of Logistic Regression classifier for sentiment lexicons + word embeddings + Twitter features

Adding the n-grams to the feature sets and including all the features, the classifier would have the best performance as shown in Table 5.10. This classifier works very well in this thesis and has a high performance.

	Precision	**Recall**	**F1-Score**	**Support**
Negative (Class 0)	0.65	0.71	0.68	412
Positive (Class 1)	0.99	0.98	0.98	9060
Average/Total	0.97	0.97	0.97	9472

Table 5.10: Performance metrics evaluation for Logistic Regression classifier

The accuracy of the Logistic Regression classifier is:

Accuracy score = 0.97

The Logistic Regression classifier confusion matrix is indicated in Table 5.11.

	Negative	**Positive**
Negative	292	120
Positive	160	8900

Table 5.11: Confusion Matrix for Logistic Regression classifier

As the confusion matrix shows, there are:

- 8900 true positive
- 120 false positive
- 292 true negative
- 160 false negative

5.3.3 Ridge Classifier

The Ridge classifier uses the Ridge Regression model and tries to minimize the effect of features which are not relevant. The Ridge classifier gives the performance which is shown in Table 5.12 by mean of **Scikit-learn** library and using n-grams plus Twitter features as the feature sets. The accuracy of the classifier in this situation is **0.95**.

	Precision	**Recall**	**F1-Score**	**Support**
Negative (Class 0)	0.50	0.55	0.52	412
Positive (Class 1)	0.98	0.98	0.98	9060
Average/Total	0.96	0.96	0.96	9472

Table 5.12: Performance metrics of Ridge Regression classifier for unigrams + Twitter features

However, the Ridge classifier gives its best performance when sentiment lexicon and word embeddings are appended to the previous features. The result is shown in Table 5.13

	Precision	**Recall**	**F1-Score**	**Support**
Negative (Class 0)	0.62	0.54	0.58	412
Positive (Class 1)	0.98	0.98	0.98	9060
Average/Total	0.96	0.97	0.96	9472

Table 5.13: Performance metrics evaluation for Ridge Regression classifier

The accuracy of the Ridge Regression classifier is:

Accuracy score = 0.96

Table 5.14 shows the confusion matrix of the Ridge Regression classifier.

	Negative	**Positive**
Negative	224	188
Positive	140	8920

Table 5.14: Confusion Matrix for Ridge Regression classifier

The confusion matrix indicates that there are:

- 8920 true positive
- 188 false positive
- 224 true negative
- 140 false negative

5.3.4 Support Vector Machines

As described in section 3.1.2.5, in a support vector machine for the two-class classification, the hyperplane is a line and the optimal line is the one with the largest distance from the closest data points in two classes and itself. Using **Scikit-learn** library, the SVC and Linear SVC classifications are applied to the dataset.

5.3.4.1 Support Vector Classifier (SVC)

In this classifier, The SVC class of **Scikit-learn** with the linear kernel is used to train the data and perform the classification. The combination of the sentiment lexicon and word embeddings as feature sets gives the accuracy of **0.95** and the performance metrics as shown in Table 5.15.

	Precision	**Recall**	**F1-Score**	**Support**
Negative (Class 0)	0.60	0.09	0.15	412
Positive (Class 1)	0.96	1.00	0.98	9060
Average/Total	0.94	0.96	0.94	9472

Table 5.15: Performance metrics of Support Vector Classifier for sentiment lexicon + word embeddings

By considering all the feature sets the performance result of the Support Vector Classifier is improved. As the Table 5.16 shows, the SVC gives very high-performance results in this state.

	Precision	**Recall**	**F1-Score**	**Support**
Negative (Class 0)	0.88	0.59	0.71	412
Positive (Class 1)	0.98	1.00	0.99	9060
Average/Total	0.98	0.98	0.98	9472

Table 5.16: Performance metrics evaluation for Support Vector Classifier

The accuracy of the Support Vector Classifier is:

Accuracy score = 0.97

The confusion matrix of the Support Vector Classifier is indicated in Table 5.17.

	Negative	**Positive**
Negative	244	168
Positive	32	9028

Table 5.17: Confusion Matrix for Support Vector Classifier with linear kernel

The confusion matrix indicates that there are:

- 9028 true positive
- 168 false positive
- 244 true negative
- 32 false negative

5.3.4.2 Linear Support Vector Classifier

The Linear Support Vector Classifier provides an almost similar result to Support Vector Classifier with linear kernel performed in section 5.3.4.1.

Considering sentiment lexicon plus word embeddings as features, the accuracy of Linear Support Vector Classifier is **0.95** . The performance of the classifier is indicated in Table 5.18.

	Precision	**Recall**	**F1-Score**	**Support**
Negative (Class 0)	0.52	0.12	0.20	412
Positive (Class 1)	0.96	0.99	0.98	9060
Average/Total	0.94	0.96	0.94	9472

Table 5.18: Performance metrics of Linear Support Vector Classifier for sentiment lexicon + word embedding

While Table 5.19 contains the evaluation metrics for the performance of the classifier having the feature sets as the combination of all the features. These features make this classifier have the best performance.

	Precision	**Recall**	**F1-Score**	**Support**
Negative (Class 0)	0.88	0.48	0.62	412
Positive (Class 1)	0.98	1.00	0.99	9060
Average/Total	0.97	0.97	0.97	9472

Table 5.19: Performance metrics evaluation for Linear Support Vector Classifier

The accuracy of the Linear Support Vector Classifier is:

Accuracy score = 0.97

Table 5.20 contains the confusion matrix of the Linear Support Vector Classifier.

	Negative	**Positive**
Negative	198	214
Positive	26	9034

Table 5.20: Confusion Matrix for Linear Support Vector Classifier

The confusion matrix indicates that there are:

- 9034 true positive
- 214 false positive
- 198 true negative
- 26 false negative

5.3.5 K-Nearest Neighbours

The combination of the n-grams, sentiment lexicon, word embeddings and Twitter features make the classifier have the performance metrics as shown below. Table 5.21 contains the performance metrics.

	Precision	**Recall**	**F1-Score**	**Support**
Negative (Class 0)	0.51	0.11	0.18	412
Positive (Class 1)	0.96	1.00	0.98	9060
Average/Total	0.94	0.96	0.94	9472

Table 5.21: Performance metrics of KNN classifier using n-grams + sentiment lexicon + word embeddings

The accuracy of the K-Nearest Neighbours classifier in the best situation is:

Accuracy score = 0.95

Table 5.22 indicates the confusion matrix of the K-Nearest Neighbours classifier.

	Negative	**Positive**
Negative	44	368
Positive	42	9018

Table 5.22: Confusion Matrix for KNN classifier

The confusion matrix indicates that there are:

- 9018 true positive
- 368 false positive
- 44 true negative
- 42 false negative

5.3.6 Decision Tree

Table 5.23 shows the performance result of the Decision Tree classifier applied to the dataset. N-grams, sentiment lexicon, word embeddings and Twitter features are considered as features in this state.

	Precision	**Recall**	**F1-Score**	**Support**
Negative (Class 0)	0.68	0.50	0.57	412
Positive (Class 1)	0.98	0.99	0.98	9060
Average/Total	0.96	0.97	0.97	9472

Table 5.23: Performance metrics evaluation for Decision Tree classifier

The accuracy of the Decision Tree classifier in best situation is:

Accuracy score = 0.96

Confusion matrix of the Decision Tree classifier is shown in the Table 5.24.

	Negative	**Positive**
Negative	204	208
Positive	94	8966

Table 5.24: Confusion Matrix for Decision Tree classifier

The confusion matrix indicates that there are:

- 8966 true positive
- 208 false positive
- 204 true negative
- 94 false negative

5.3.7 Random Forest

Random Forest classifier performance metrics, considering all the features are shown in Table 5.25. The accuracy of the classifier in this state is

	Precision	**Recall**	**F1-Score**	**Support**
Negative (Class 0)	0.99	0.36	0.53	412
Positive (Class 1)	0.97	1.00	0.99	9060
Average/Total	0.97	0.97	0.97	9472

Table 5.25: Performance metrics evaluation for Random Forest classifier

The accuracy of the Random Forest classifier is:

Accuracy score = 0.97

Table 5.26 consists of the confusion matrix of the Random Forest classifier.

	Negative	**Positive**
Negative	148	264
Positive	2	9058

Table 5.26: Confusion Matrix for Random Forest classifier

The confusion matrix indicates that there are:

- 9058 true positive
- 264 false positive
- 148 true negative
- 2 false negative

5.3.8 Neural Networks

The Multilayer Perceptron (MLP) classifier of the **Scikit-learn** library, trains the dataset by mean of propagation algorithm. the performance result of this classifier is shown in Table 5.27.

	Precision	**Recall**	**F1-Score**	**Support**
Negative (Class 0)	0.76	0.45	0.56	412
Positive (Class 1)	0.98	0.99	0.98	9060
Average/Total	0.97	0.97	0.97	9472

Table 5.27: Performance metrics evaluation for MLP classifier

The accuracy of the MLP classifier is:

Accuracy score = 0.96

Table 5.28 indicates the confusion matrix of the K-Nearest Neighbours classifier.

	Negative	**Positive**
Negative	184	228
Positive	58	9002

Table 5.28: Confusion Matrix for MLP classifier

The confusion matrix indicates that there are:

- 9002 true positive
- 228 false positive
- 184 true negative
- 58 false negative

Chapter 6

Analysis and Conclusion

In this thesis different machine learning algorithms are applied to the provided dataset. In Gaussian Naive Bayes addition of the unigrams features to the sentiment lexicon, word embedding and Twitter features, improve the performance of the classifier by increasing the accuracy and f1-score from 0.67 and 0.77 to 0.96 and 0.96 respectively. While in Bernoulli Naive Bayes the combination of the unigrams, bigrams, sentiment lexicon and Twitter features outperforms the result of combining all the features including word embeddings. It increases the accuracy and f1-score of the classifier from 0.86 and 0.90 to 0.97 and 0.97 respectively.

In Logistic Regression the best performance is given using all the features including unigrams, bigrams, sentiment lexicon, word embeddings and Twitter features. The accuracy and f1-score of this classifier are 0.97. The Logistic Regression is one of the strongest classifiers in this work since it provides high performance with any feature set, however, the best scores are those mentioned. Also, the Ridge Classifier gives its highest accuracy when using all the features as 0.96. The f1-score also is 0.96.

Support Vector Classifier is one of the best classifiers in this work since it gives the highest f1-score which is 0.98. This is the case in which unigrams, sentiment lexicon, word embeddings and Twitter features are the features.

The accuracy of the Support Vector Classifier with the linear kernel is 0.97 and the f1-score is 0.98. As shown in Table 5.17 the number of true positives is 9028 which indicates the high precision of the classifier. The Linear Support Vector Classifier also gives the high accuracy and f1-score of 0.97 considering all the features.

The K-Nearest Neighbours classifier using all the features gives the accuracy and f1-score about 0.95 and 0.96. Decision Tree classifier gives almost the same results when using unigrams and Twitter features as feature sets and when combining all the features. However, the accuracy of the classifier using unigrams and Twitter features is 0.97 while concatenating the word embeddings and sentiment lexicons to the previous features give the accuracy of 0.96. The f1-score in both states is 0.97. This indicates that in Decision Tree classifier, sentiment lexicon and word embeddings do not play an important role but unigrams and Twitter features are the most important features.

In Random Forest, the combination of unigrams, sentiment lexicon, word embeddings and Twitter features gives the accuracy and f1-score about 0.97. The number of false negatives in this classifier is only 2 which is very low. Also the number of true positives in this classifier is 9058 which is a high number in comparison with other classifiers. The accuracy of Neural Networks using Multilayer Perceptron classifier is 0.96 and the f1-score is 0.97.

In Table 6.1, the accuracy and other performance evaluation metrics of the machine learning algorithms used are indicated.

Algorithm	**Accuracy**	**Precision**	**Recall**	**F1-Score**
Gaussian NB	0.96	0.96	0.97	0.96
Bernoulli NB	0.86	0.94	0.87	0.90
Logistic Regression	0.97	0.97	0.97	0.97
Ridge Classifier	0.96	0.96	0.97	0.96
SVC	0.97	0.98	0.98	0.98
Linear SVC	0.97	0.97	0.97	0.97
K-Nearest Neighbours	0.95	0.94	0.96	0.94
Decision Tree	0.96	0.96	0.97	0.97
Random Forest	0.97	0.97	0.97	0.97
MLP	0.96	0.97	0.97	0.97

Table 6.1: Comparison of the performance metrics for unigrams + sentiment lexicon + word embeddings + Twitter features

In Table 6.2 the comparison of the number of true positives, false positives, true negatives and false negatives in different algorithms is shown.

Algorithm	**TP**	**FP**	**TN**	**FN**
Gaussian NB	8992	250	162	68
Bernoulli NB	8036	236	176	1024
Logistic Regression	8900	120	292	160
Ridge Classifier	8920	188	224	140
SVC	9028	168	244	32
Linear SVC	9034	214	198	26
K-Nearest Neighbours	9018	368	44	42
Decision Tree	8966	208	204	94
Random Forest	9058	264	148	2
MLP	9002	228	184	58

Table 6.2: Comparison of the number of TP, FP, TN, and FN for unigrams + sentiment lexicon + word embeddings + Twitter features

The performance metrics shown in Table 6.1 are given using unigrams, sentiment lexicon, word embeddings and Twitter features. As it is indicated, the Support Vector Classifier (SVC) has the highest precision, recall, and f1-score in addition to the accuracy. These results are derived by training the classifier on the dataset which includes 50,870 tweets. The training dataset that contains 41,398 tweets is not balanced. Therefore Accuracy alone could not be an effective metric.

One of the techniques to handle the imbalanced data is downsampling which matches the number of samples in minority class that in this work is the negative class to the samples from the majority class which is positive class [66]. The samples from the majority class are chosen randomly. Table 6.3 shows the performance metrics of the classifiers after downsampling.

Algorithm	**Accuracy**	**Precision**	**Recall**	**F1-Score**
Gaussian NB	0.85	0.96	0.85	89
Bernoulli NB	0.84	0.94	0.84	0.88
Logistic Regression	0.95	0.97	0.95	0.96
Ridge Classifier	0.95	0.96	0.96	0.96
SVC	0.93	0.96	0.93	0.94
Linear SVC	0.94	0.97	0.95	0.95
K-Nearest Neighbours	0.80	0.94	0.80	0.86
Decision Tree	0.92	0.96	0.93	0.94
Random Forest	0.92	0.95	0.92	0.93
MLP	0.92	0.96	0.93	0.94

Table 6.3: Comparison of performance metrics for unigrams + sentiment lexicon + word embeddings + Twitter Features after downsampling

As the Table 6.3 indicates, Linear SVC, Logistic Regression, and Ridge classifier performs very well and have the highest performance.

As described earlier, for the classification, the retweets are deleted from the dataset since they are not efficient in classification but they could be used in this stage. The dataset collected contains 1,500,000 tweets after deletion of the duplicates. From this amount, 120,000 tweets are labeled. This data contains also retweets which are helpful in analyzing the location of the users. The objective of this analysis is to find the places in which there is the most number of climate change deniers. According to the tweets classification and the existing dataset, majority of the tweets are positive, therefore most of the users believe in the existence of climate change and support the environment. Among the negative tweets, the majority of them are from the United States. This shows that most of the climate change deniers are in the United States. Georgia, Oklahoma, Kentucky, Texas, Kansas, Alabama and Mississippi are the states with the most number of negative tweets.

The interesting point is that according to the Emission Database for Global Atmospheric Research (EDGAR), the United States is among the top two CO_2 emitters in the world. This is shown in Figure 6.1. According to EPA, Texas has the most amount of annual CO_2 emission [67]. Ohio and Kentucky are also among the top emitters. Comparing these facts with data discussed above, it is obvious that climate change deniers, such as fossil fuel industries who want to stop the regulations on their activities, are the biggest CO_2 emitters and climate polluters. These states are typically from the Republican party, the party of the current president of the United States, Donald Trump. As discussed in section 2.1.3.2, In June 2017, Trump announced the withdrawal of the United States from the Paris agreement.

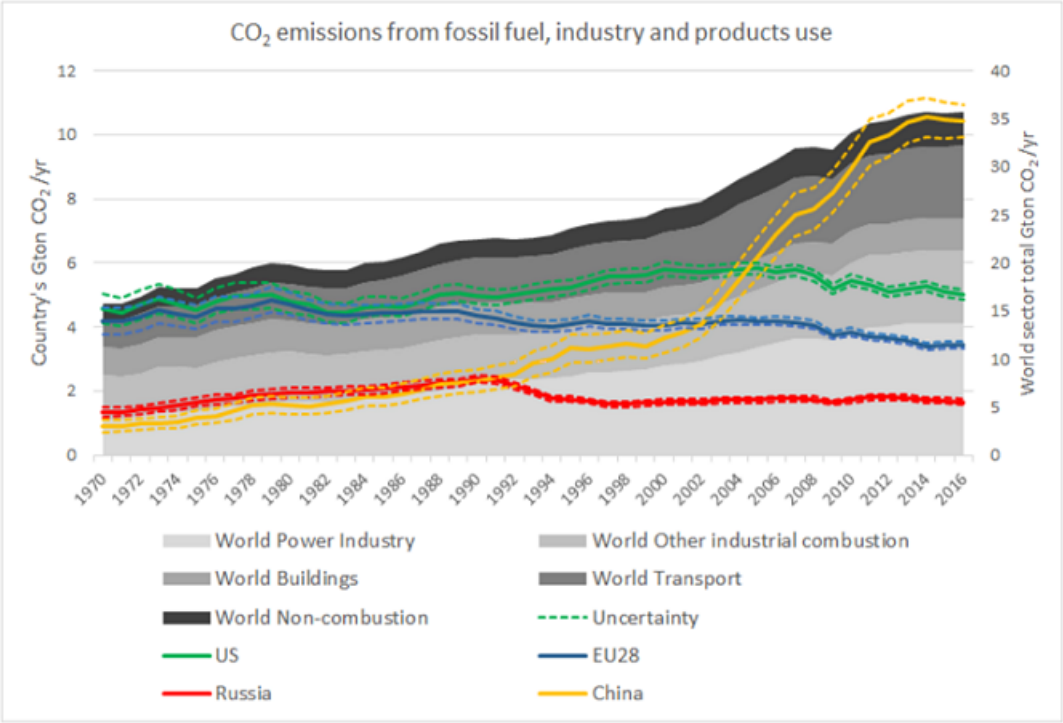

Figure 6.1: Total annual emissions of fossil CO2 in Gton CO2/yr. [10]

Figure 6.2 shows the U.S. 2016 election results by the states in which the blue color represents that the majority of the voters voted for the Democratic candidate, Hillary Clinton, while the red color shows that the majority of the voters voted for the Republican candidate, Donald Trump.

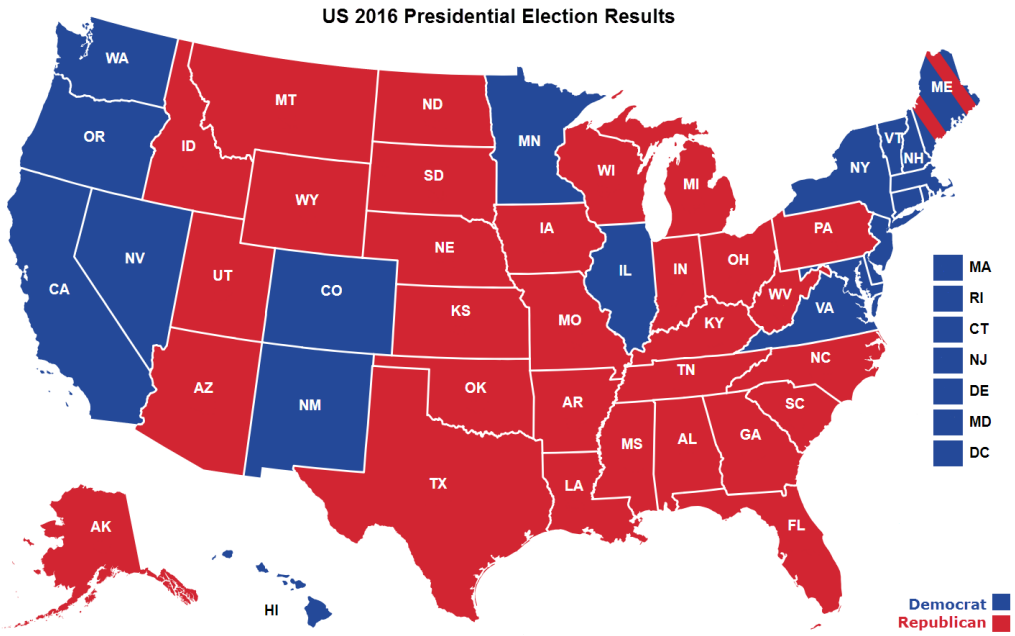

Figure 6.2: U.S. Election 2016 [11]

In Figure 6.3 from the tweets with the location, the states in which more than 50% of them are negative (climate change deniers) are shown by red while the states in which more than 50% of the tweets are positive are shown by blue. Comparing Figure 6.2 and Figure 6.3, it is deduced that the most climate change deniers are from the Republican states who voted for Donald Trump in the U.S. election 2016.

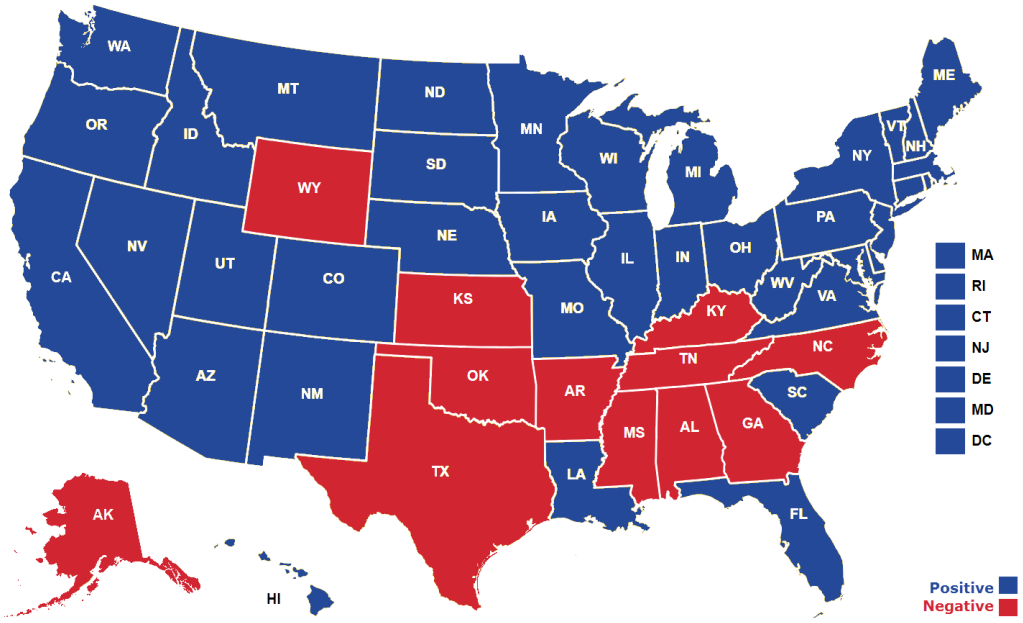

Figure 6.3: U.S. top positive and negative states [11]

As future developments on this work, There could be improvements on using deep learning techniques for classification. In this thesis, the analysis is on public opinion about climate change on Twitter while in future this work could be done on other social networks too in order to perform wider analysis and provide bigger datasets.

Bibliography

[1] "August 2017 was second warmest on record." https://climate.nasa.gov/news/2630/august-2017-was-second-warmest-on-record/. Retrieved on September 18, 2017.

[2] "carbon-dioxide level concentration." https://climate.nasa.gov/vital-signs/carbon-dioxide/. Retrieved on March, 2018.

[3] "sea level rise." https://climate.nasa.gov/sea-level/. Retrieved on December, 2017.

[4] "Linear vs logistic regression." https://www.machinelearningplus.com/machine-learning/attachment/linear_vs_logistic_regression/. Retrieved on September, 2017.

[5] "Support vector machine diagram." http://arun-aiml.blogspot.co.uk/2017/support-vector-machine-svm. Retrieved on July, 2017.

[6] "Step function diagram in neural networks." https://zeolearn.com/magazine/understanding-blocks-of-neural-networks/. Retrieved on April, 2018.

[7] K. N. Haque, M. Yousuf, and R. Rana, "Image denoising and restoration with cnn-lstm encoder decoder with direct attention," 01 2018.

[8] "Neural network diagram." `http://www.astroml.org/book_figures/appendix/fig_neural_network.html`.

[9] "Recurrent nn vs feed-forward." `https://towardsdatascience.com/recurrent-neural-networks`. Retrieved on February, 2018.

[10] C. M. G. D. M. M. S. E. O. J. P. J. S. K. Janssens-Maenhout, G., "Fossil co2 and ghg emissions of all world countries," *Publications Office of the European Union*, 2017.

[11] "U.s. election 2016." ://thefreecities.com/liberal-orthodoxy-is-driving-off-a-cliff-a3d007a58303.

[12] Y. Jianjun, O. Jonathan, P. Cheryl, and S. Ronald, "Big jump of record warm global mean surface temperature in 20142016 related to unusually large oceanic heat releases," *Geophysical Research Letters*, vol. 45, no. 2, pp. 1069–1078.

[13] J. E. Doll and M. Baranski, "Greenhouse gas basics,"

[14] V. UNFCCC, "Adoption of the paris agreement," *I: Proposal by the President (Draft Decision), United Nations Office, Geneva (Switzerland)*, no. s 32, 2015.

[15] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," *Ain Shams Engineering Journal*, vol. 5, no. 4, pp. 1093 – 1113, 2014.

[16] I. P. O. C. CHANGE, "Ipcc second assessment,"

[17] R. Watson, L. Meira Filho, E. Sanhueza, and A. Janetos, "Greenhouse gases: sources and sinks,"

[18] C. UNFCCC, "United nations framework convention on climate change," in *Conference of the Parties at its fourth session*, 1998.

[19] A. Collomb, C. Costea, D. Joyeux, O. Hasan, and L. Brunie, "A Study and Comparison of Sentiment Analysis Methods for Reputation Evaluation," Tech. Rep. RR-LIRIS-2014-002, LIRIS UMR 5205 CNRS/INSA de Lyon/Universit Claude Bernard Lyon 1/Universit Lumire Lyon 2/cole Centrale de Lyon, Mar. 2014.

[20] M. Desai and M. A. Mehta, "Techniques for sentiment analysis of twitter data: A comprehensive survey," *2016 International Conference on Computing, Communication and Automation (ICCCA)*, pp. 149–154, 2016.

[21] S. Vohra and J. Teraiya, "A comparative study of sentiment analysis techniques," *Journal JIKRCE*, vol. 2, no. 2, pp. 313–317, 2013.

[22] A. Jurek, M. D. Mulvenna, and Y. Bi, "Improved lexicon-based sentiment analysis for social media analytics," *Security Informatics*, vol. 4, p. 9, Dec 2015.

[23] C. Kaushik and A. Mishra, "A scalable, lexicon based technique for sentiment analysis," *CoRR*, vol. abs/1410.2265, 2014.

[24] M. Z. Asghar, A. Khan, and S. Ahmad, "Lexicon-based sentiment analysis in the social web," 2014.

[25] S. V. Wawre and S. N. Deshmukh, "Sentiment classification using machine learning techniques," *International Journal of Science and Research (IJSR)*, vol. 5, no. 4, pp. 819–821, 2016.

[26] J. Brownlee, *Master Machine Learning Algorithms: Discover how They Work and Implement Them from Scratch.* 2016.

[27] P. Garg, "Sentiment Analysis of Twitter Data using NLTK in Python," Master's thesis, Thapar University, Patiala, India, 2016.

[28] J. He, L. Ding, L. Jiang, and L. Ma, "Kernel ridge regression classification," in *Neural Networks (IJCNN), 2014 International Joint Conference on*, pp. 2263–2267, IEEE, 2014.

[29] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," in *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pp. 79–86, Association for Computational Linguistics, 2002.

[30] L. E. Peterson, "K-nearest neighbor," *Scholarpedia*, vol. 4, no. 2, p. 1883, 2009.

[31] A. Krenker, J. Bester, and A. Kos, "Introduction to the artificial neural networks," in *Artificial neural networks-methodological advances and biomedical applications*, InTech, 2011.

[32] A. Vehbi Olgac and B. Karlik, "Performance analysis of various activation functions in generalized mlp architectures of neural networks," vol. 1, pp. 111–122, 02 2011.

[33] A. F. Agarap, "Deep learning using rectified linear units (relu)," *arXiv preprint arXiv:1803.08375*, 2018.

[34] R. A. F. L. H. B. L. S. F. d. R. A. Ivan Nunes da Silva, Danilo Hernane Spatti, *Artificial Neural Networks*. Springer International Publishing, 2017.

[35] A. J. Mehta, H. A. Mehta, T. Manjunath, and C. Ardil, "A multi-layer artificial neural network architecture design for load forecasting in power systems," *International Journal of Applied Mathematics and Computer Sciences*, vol. 4, no. 4, pp. 227–240, 2008.

[36] P. Tino, L. Benuskova, and A. Sperduti, "Artificial neural network models," in *Springer Handbook of Computational Intelligence*, pp. 455–471, Springer, 2015.

[37] M. Sazli, "A brief review of feed-forward neural networks," vol. 50, pp. 11–17, 01 2006.

[38] G. Gebremeskel, "Sentiment analysis of twitter posts about news," *Sentiment Analysis. Feb*, 2011.

[39] Y. Goldberg and G. Hirst, *Neural Network Methods in Natural Language Processing*. Morgan & Claypool Publishers, 2017.

[40] J. K and J. Saini, "Stop-word removal algorithm and its implementation for sanskrit language," vol. 150, pp. 15–17, 09 2016.

[41] I. Abu El-Khair, "Effects of stop words elimination for arabic information retrieval: A comparative study," vol. 4, pp. 119–133, 01 2006.

[42] A. G. Jivani *et al.*, "A comparative study of stemming algorithms," *Int. J. Comp. Tech. Appl*, vol. 2, no. 6, pp. 1930–1938, 2011.

[43] M. Trupthi, S. Pabboju, and G. Narasimha, "Improved feature extraction and classificationsentiment analysis," in *Advances in Human Machine Interaction (HMI), 2016 International Conference on*, pp. 1–6, IEEE, 2016.

[44] D. Jurafsky and J. H. Martin, "Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition," 2009.

[45] Y. Goldberg, "Neural network methods for natural language processing," *Synthesis Lectures on Human Language Technologies*, vol. 10, no. 1, pp. 1–309, 2017.

[46] K. Ghag and K. Shah, "Sentitfidf–sentiment classification using relative term frequency inverse document frequency," *International Journal of Advanced Computer Science & Applications*, vol. 5, no. 2, 2014.

[47] M. Z. Asghar, A. Khan, S. Ahmad, and F. M. Kundi, "A review of feature extraction in sentiment analysis," *Journal of Basic and Applied Scientific Research*, vol. 4, no. 3, pp. 181–186, 2014.

[48] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," *CS224N Project Report, Stanford*, vol. 1, no. 12, 2009.

[49] A. Pak and P. Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining.," in *LREc*, vol. 10, 2010.

[50] L. Barbosa and J. Feng, "Robust sentiment detection on twitter from biased and noisy data," in *Proceedings of the 23rd international conference on computational linguistics: posters*, pp. 36–44, Association for Computational Linguistics, 2010.

[51] D. Davidov, O. Tsur, and A. Rappoport, "Enhanced sentiment learning using twitter hashtags and smileys," in *Proceedings of the 23rd international conference on computational linguistics: posters*, pp. 241–249, Association for Computational Linguistics, 2010.

[52] E. Kouloumpis, T. Wilson, and J. D. Moore, "Twitter sentiment analysis: The good the bad and the omg!," *Icwsm*, vol. 11, no. 538-541, p. 164, 2011.

[53] H. Saif, Y. He, and H. Alani, "Semantic sentiment analysis of twitter," in *International semantic web conference*, pp. 508–524, Springer, 2012.

[54] I. Alfina, D. Sigmawaty, F. Nurhidayati, and A. N. Hidayanto, "Utilizing hashtags for sentiment analysis of tweets in the political domain," in *Proceedings of the 9th International Conference on Machine Learning and Computing*, ICMLC 2017, (New York, NY, USA), pp. 43–47, ACM, 2017.

[55] R. Gull, U. Shoaib, S. Rasheed, W. Abid, and B. Zahoor, "Pre processing of twitter's data for opinion mining in political context," vol. 96, pp. 1560–1570, 12 2016.

[56] D. Effrosynidis, S. Symeonidis, and A. Arampatzis, "A comparison of pre-processing techniques for twitter sentiment analysis," 09 2017.

[57] A. I. Baqapuri, "Twitter sentiment analysis," *CoRR*, vol. abs/1509.04219, 2015.

[58] R. d. Groot, "Data mining for tweet sentiment classification," Master's thesis, 2012.

[59] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, (New York, NY, USA), pp. 168–177, ACM, 2004.

[60] R. P. Lebret, "Word embeddings for natural language processing," 2016.

[61] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE*, vol. 41, no. 6, pp. 391–407, 1990.

[62] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *CoRR*, vol. abs/1301.3781, 2013.

[63] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, 2014.

[64] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[65] Y. Liu, *Python Machine Learning by Example*. Packt Publishing, 2017.

[66] Y. Liu, N. V. Chawla, M. P. Harper, E. Shriberg, and A. Stolcke, "A study in machine learning from imbalanced data for sentence boundary detection in speech," *Computer Speech Language*, vol. 20, pp. 468–494, 2006.

[67] "State co_2 emissions." `https://www.epa.gov/statelocalenergy/state-co2-emissions-fossil-fuel-combustion`. Retrieved on 2015.