ANKARA YILDIRIM BEYAZIT UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

ELECTION PREDICTION WITH MACHINE LEARNING

M.Sc. Thesis by

Eyyüp YETKİN

Department of Computer Engineering

February, 2021

ANKARA

M.Sc. THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"ELECTION PREDICTION WITH MACHINE LEARNING"** completed by **EYYÜP YETKİN** under supervision of **Prof. Dr. Fatih V. ÇELEBİ** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Dr. Fatih Vehbi ÇELEBİ

Supervisor

Prof. Dr. Şahin EMRAH

Jury Member

Dr. Mustafa YENİAD

Jury Member

Prof. Dr. Ergün ERASLAN

Director

Graduate School of Natural and Applied Sciences

ETHICAL DECLARATION

I hereby declare that, in this thesis which has been prepared in accordance with the Thesis Writing Manual of Graduate School of Natural and Applied Sciences,

- All data, information and documents are obtained in the framework of academic and ethical rules,
- All information, documents and assessments are presented in accordance with scientific ethics and morals,
- All the materials that have been utilized are fully cited and referenced,
- No change has been made on the utilized materials,
- All the works presented are original,

and in any contrary case of above statements, I accept to renounce all my legal rights.

Date: 2021, 24 February **Signature: ……..**

Name & Surname: Eyyüp YETKİN

ACKNOWLEDGMENTS

In conducting this thesis, my thesis advisor and my dear supervisor, Ankara Yıldırım Beyazıt University, Institute of Science, Head of Computer Engineering Department and Dean of the Faculty of Engineering and Natural Sciences, Prof. Dr. Fatih Vehbi ÇELEBİ, I would like to express my endless gratitude and respect to himself., I would like to thank Ankara Yıldırım Beyazıt University Computer Engineering Department, my lecturers who did not spare their help and my mother who has helped me reach today, who is with me in every moment of my life, and who does not spare me their financial and moral support

2021, 24 February **Eyyüp YETKİN**

ELECTION PREDICTION WITH MACHINE LEARNING

ABSTRACT

Political elections are carried out according to election systems such as parliamentary elections and presidential elections and voting procedures are carried out at the polls because of the democracy system is implemented in most countries nowadays. The majority of the survey companies apply the surveys among the public by selecting the participants before the elections and predicting elections based on the responses of the participants. However, Twitter is one of the platforms where people express the most opinions about elections in today's technology age. Election predictions can be made by taking the opinions of millions of people from this platform instead of choosing participants from among the public and even with developing machine learning and artificial intelligence technologies, emotion analysis can be made from the data on Twitter using various classification algorithms together with these data and sentiment analysis libraries and it can contribute all employees, companies and institutions working on election predictions substantially. It can be easily determined who will win the elections according to the tweets written by people using emotion analysis libraries and machine learning technology. Besides, even voting can be made without going to the polls according to a comment on Twitter. This study is a election prediction study using emotion analysis, text classification and machine learning algorithms according to tweets. This study includes sentiment analysis libraries, machine learning algorithms and comparative accuracy results of algorithms, time performance and comparative results of SentiWordNet and TextBlob libraries that perform sentiment analysis. Thus, the conclusion on which algorithm and emotion analysis library to use will more accurately predict election predictions has shed light on our purpose. As a result, TfidfVectorizer classification method, SentiWordNet emotion analysis library with a 10% difference by TextBlob and Passive Agressive Classifier algorithm with 84 % accuracy test score, 99 % accuracy train score, 86 % F1 Test Score and its online learning algorithm has been observed the most efficient algorithm and methods.

Keywords: Twitter, textblob, sentiwordnet, machine learning, sentiment analysis, tfidfvectorizer, passive agressive classifier

MAKİNE ÖĞRENMESİ İLE SEÇİM TAHMİNİ

ÖZ

Günümüzde ülkelerin çoğunda demokrasi sistemi uygulandığından dolayı, siyasi seçimler milletvekili seçimleri, cumhurbaşkanlığı seçimleri gibi seçim sistemlerine göre uygulanmakta, oy kullanma işlemleri sandık başında yapılmaktadır. Anket şirketlerinin büyük çoğunluğu seçimlerden önce, anketleri halk arasında katılımcılar seçerek uygulamakta, katılımcıların verdikleri yanıtlar doğrultusunda seçim tahminleri yapmaktadır. Oysa günümüz teknoloji çağında, Twitter, seçimler konusunda, insanların en çok görüş bildirdiği platformlardan birisidir. Twitter üzerinden halk arasından katılımcı seçmek yerine, milyonlarca kişinin görüşünü bu platformdan alarak seçim tahminleri yürütülebilir, hatta gelişen makine öğrenmesi ve yapay zeka teknolojileri ile Twitter üzerindeki verilerden duygu analizi yapılarak, bu veriler ve duygu analizi ile birlikte çeşitli sınıflandırma algoritmaları kullanılarak seçim tahmini yapılabilir ve seçim tahminleri konusunda çalışan tüm çalışanlara, şirketlere ve kurumlara büyük ölçüde katkıda bulunabilir. Kişilerin yazdığı tweetlere göre duygu analizi ve makine öğrenmesi teknolojisi kullanılarak kolaylıkla seçimleri kimin kazanacağı belirlenebilir. Ayrıca Twitter üzerindeki bir yoruma göre, sandık başına gitmeden seçim oylaması bile yapılabilir. Bu çalışma, tweetlere göre duygu analizi, metin sınıflandırma ve makine öğrenmesi algoritmaları kullanılarak seçim tahmini yapma çalışmasıdır. Çalışmada duygu analizi kütüphaneleri, makine öğrenmesi algoritmaları, algoritmaların karşılaştırmalı doğruluk (accuracy) sonuçları, zaman performansı ve duygu analizi yapan SentiWordNet ve TextBlob kütüphanelerin de karşılaştırmalı sonuçları yer almaktadır. Böylece hangi algoritmayı ve duygu analizi kütüphanesi kullanmanın seçim tahminlerini daha doğru tahmin edeceğine dair sonuç, amacımıza ışık tutmuştur. Sonuç olarak, TfidfVectorizer sınıflandırma yöntemi, %10 luk bir duygu tahmini farkıyla SentiWordNet duygu analizi kütüphanesi ve kendine özgü online öğrenme algoritması ile ve % 84'lük doğruluk test skoru, % 99 'luk doğruluk eğitim skoru ve %86 lık F1 Test Skoru ile Pasif Agresif Sınıflandırıcı algoritmasının en verimli çalışan algoritma ve yöntemler olduğu gözlenmiştir.

Anahtar Kelimeler: Twitter, textblob, sentiwordnet, makine öğrenimi, duygu analizi, tfidfvectorizer, pasif agresif sınıflandırıcı

CONTENTS

NOMENCLATURE

Acronyms

NLP	Natural Language Processing
JSON	Javascript Object Notation
CSV	Comma Separated Values
NLTK	Natural Language Toolkit
TF-IDF	Term Frequency-Inverse Document Frequency
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
SVM	Support Vector Machine

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Overview

Elections were not held when countries were ruled by kingdoms, empires, monarchical and oligarchic forms of government were adopted in the past. Although elections were held in some periods of the Ancient Roman Civilization, the elections were not implemented effectively and the world continued to be governed by empires and dynasties. However, In the 20th century, new states were established, empires and kingdoms collapsed, the types of elections based on popular voting that emerged such as president, parliament and senate elections started to be held and leaders were determined according to these elections. Using of social media and machine learning methods make our outside world think and create different ideas and solution methods nowadays. Especially, Twitter is the very popular social media platform about opinion and sentiment specifying. The million of users write and specify their opinions and sentiments on Twitter. Because of Twitter is more popular social media and many people use this application, the opinions and sentiments on Twitter can be used in survey applications and survey studies can have most approximate prediction results about elections. The elections are usually done at the polls. If the opinions and sentiments can be used at the elections, the election system will not need the polls. Therefore, to study about election prediction based on opinions, sentiments is so important and it can break through in the election systems and the world. It is possible contributing survey studies by receiving tweets from people on Twitter which the social media platform where most ideas and comments are posted using sentiment analysis libraries and even elections can be applied without going to the polls. In addition, with classifying text using various machine algorithms and classification methods according to these tweets, the elections can be predicted according to tweets.

1.2 Aim and Structure of the Thesis

This study was conducted to contribute survey studies, to give more accurate results than survey results, to contribute the studies about election without polls and to can provide election system without polls. The machine learning algorithms and text classification techniques benefit to studies of the prediction and human-computer interaction. Spam detection, credit card fraud detection, election prediction applications, text classifications, sentiment analysis are the examples of applications in the machine learning. The study is the study of the election prediction according to the tweets using sentiment analysis, text classification and machine learning algorithms. This study is consist of taking tweets, preprocessing methods, applying sentiment analysis, text classification and supervised learning algorithms.

CHAPTER 2

LITERATURE REVIEW

The literature researches about sentiment analysis and classification and election prediction were researched and examined. In this section, there are 9 studies about sentiment analysis and classification, 8 studies about election prediction.

2.1 Sentiment Analysis and Classification

The earliest work on sentiment analysis and text classification is Hatzivasilloglou and McKeown's study to predict the semantic aspect of adjectives in 1997 [1]. The aim of the study is to identify and verify the restrictions arising from conjunctions on the semantic orientation of compound adjectives. Log-Linear regression and clustering algorithms were used in the study.

Three million tweets were analyzed that mentioned and retweeted the 13 most influential Twitter users. Firstly, the positive and negative followers of the users were separated, the positive or negative effects of popular users on their followers were discovered and positive-negative measurements were developed for this effect from these 2 findings using Granger causality analysis, the positive-negative mood change based on the time series of the audience users were found to be associated with a view of real world sentiments. Unnecessary characters were removed before sentiment analysis [2]. LIWC (Sentiment analysis was applied using the (Linguistic Inquiry and Word Count) method, Pearson correlation analysis, Spearman rank correlation analysis, and Granger causality analysis were used to determine the sentimental relationship between the users and his followers. Thus, Sentiment analysis, relationship and change were determined.

NLP (Natural Language Processing) and Sentiment analysis and its applications were mentioned [3].

Haddi, Liu and Shi emphasized that feelings and thoughts are mostly in feedback, but sentiment analysis is difficult to derive because of very large data. The data consists of 2 data sets related to the film analysis. The first data set consists of a total of 1400 data, 700 positive and 700 negative and the second data set consists of 1000 positive and 1000 negative data. Html tags, spaces and non-alphabetic characters were removed. The data sets was preprocessed by applying the stopwords process and by performing root separation. After the data sets was rooted, the first data set fell from 10450 features to 7614 features, and the second data set from 12860 features to 9058 features [4]. Comparison has been made using FF (Feature Frequency), FP (Feature Presence), and TF-IDF (Term Frequency-Inverse Document Frequency) feature weighting methods and the Support Vector Machine algorithm has been applied and the FP method has reached the highest accuracy score with 93% and SVM and TF-IDF method reached the highest score with 93.5% when it was applied to improve the predictions using the SVM.

In the study conducted by Denecke and Kerstin, a multilingual sentiment analysis application which has not been applied until today according to his claim was used. The data were translated into English by the Promt Excellent Translation Technology tool, LingPipe Classification which is a language classification tool, polarity analysis classification with SentiWordNet and Logistic Classifier in addition to SentiWordNet polarity analysis, total 3 types of classification techniques were used. Data with 1000 positive sensitivity and 1000 negative sensitivity were taken from the IMDB (Internet Movie Database) archive to train data [5]. Data with 100 positive and 100 negative sensitivity were taken from Amazon in a German language other than English. When it was four or 5 stars, it is considered positive. When it was 1 or 2 stars, it is considered negative. LingPipe and SentiWordNet classifiers had 58% and 59% accuracy, the study that was made by Logistic classifier and SentiWordNet was achieved by accuracy of 66 %.

Erik Boiy and Marie-Francine Moens used the n-gram technique and Support Vector Machine, Multinomial Naive Bayes and Maximum Entropy classification algorithms were applied to the texts in their studies and it was claimed that the best performance was unigram and maximum entropy [6].

In a similar study to this study, Sentiment analysis classification was applied with SentiWordNet [7]. Positive, negative and neutral states were deduced.

In the emotion analysis study using Machine Learning techniques by Bo Pang, Lillian Lee and Vaithyanathan, it was emphasized that emotion analysis can help business intelligence applications and advice systems, the IMDB (Internet Movie Database) data set was used, grades were automatically taken from this database and transformed into positive, negative and neutral states. The data set consists of 1301 positive and 752 negative comments. Two graduate students in computer science were asked to choose good indicator words for positive and negative emotions in their film reviews independently and it was found that one achieved 58% accuracy and the other reached 64% accuracy [8]. In pre-processing, the punctuation marks were converted from Html document format to plain form, punctuation marks were considered as separate word elements, unigrams and bigrams methods from n-gram operations which are the operations of dividing the text into word groups, as well as Support Vector Machine, Maximum Entropy and Naive Bayes algorithms were applied to achieve the highest accuracy. When the unigrams method was applied, the Support Vector Machine reached the score, when only the bigrams method was applied, the Maximum Entropy method reached the score.

In the sentiment analysis study of Alvaro Ortigosa, Jose M. Martin, and Rosa M. Carro on Facebook, the data were converted to lowercase letters in the preprocessing stage, then divided into sentences, tokenized for each sentence, punctuation marks were removed, repetitive words were removed [9]. Analysis was performed using NLTK (Natural Language Toolkit) and polarity scores were calculated. Working with the SentBulk tool was supported and comparative results were obtained by applying

Support Vector Machines, Naive Bayes and Decision Tree algorithms as a machine learning approach.

2.2 Election Prediction

The election prediction was applied and the elections were predicted with the Naive stochastic approach [10].

Prasetyo and Hauff conducted sentiment analysis and followed various approaches to predict the Indonesian Elections through tweets on Twitter [11].

In the study of Kazem Jahanbakhsk and Yumi Moon, it was stated that Twitter has the number of users representing 16% of internet users and it is a powerful tool to predict the election results. Tweepy library was used to fetch datas related to the 2012 US Presidential Election. The datas like Barack Obama, Mitt Romney, US Election, Paul Ryan and Joe Biden were filtered, data between 29 September 2012 and 6 November 2012 were collected, a total of 39 million tweets were collected and stored in MySql database. An index has been added to make Mysql Queries that work fast. Java-based ML-NLP (Machine Learning- Natural Language Processing) software tool was used for advanced machine learning and text analysis on many tweets. Hibernate ORM (Object Relationship Mapping) which is one of the object-relational matching tools for ML-NLP tool to access the database and Dropwizard tool for ML-NLP analysis to work on Restful Web Service were used. Four methods were applied with ML-NLP. The statistical method that performs all basic statistical calculations, the text analysis method for simple text analysis, the Naive Bayes classifier for sentiment analysis and the LDA (Linear Discriminant Analysis) algorithm for subject modeling were used. In the statistical analysis method, frequency classification was made according to dates, politicians and hashtags. 989 tweets were selected and manually tagged for the Naive Bayes classifier, alphanumeric characters in the tweets were removed to reduce noise before classification, tokenization was performed. Java Programming language, MySql database and Restful Web service were used and Naive Bayes Classification algorithm was applied [12]. The strongest classification method was Naive Bayes.

In a study by Tunggawan and Soelistio to predict the 2016 US Presidential Elections, Twitter Streaming Api On Tweepy was used to fetch data from Twitter. 371264 tweets with the hashtag #Election2016 were taken between December 16, 2015 and February 29, 2016, Url and pictures has been removed, filtering has been made with the names of presidential candidates and twitter hashtags data were manually tagged [13]. Presidential candidates were estimated by finding the candidate with the most positive emotions using Naive Bayes classifier. The data between December 16, 2015 and February 2, 2016 as the training data set for the classifier, the data between February 3, 2016 and February 29, 2016 as the test data set, data between February 3, 2016 and February 9, 2016 were selected as the predictive data set and 10 cross-validation methods were applied and the model achieved 95.8% accuracy, stating that the survey data on realpolitics.com had an accuracy of 26.7%. Twitter sentiment analysis proved to be a much more accurate method and a more successful method than a survey study.

In the study related to predict US elections, Twitter social media tool was used to predict the 2012 US Presidential Elections, a total of 40 million tweets were taken between September 29, 2012 and November 6, 2012 using Twitter's Restful api, over 40 million tweets that mention only Obama and Romney was selected and the number decreased to 27 million [14]. Sentiment analysis was performed using Stanford NLP (Natural Language Processing) library to train the data set, emphasized that the Naive Bayes algorithm is limited in text-based classifications and the Support Vector Machine algorithm was more healthy by classifying even over 27 million tweets. While the Support Vector Machine algorithm was applied, hyperparameter adjustment was applied with the GridSearchCV method, and the Support Vector Machine hyperparameters that have highest performance were also shown.

In the study conducted to predict the Indonesian Presidential Elections using sentiment analysis and similar to this study, it was stated that the total Indonesian population using the internet was 143 million based on 2017 and approximately 90 percent of the

population used social media tools such as Twitter, Facebook or Instagram. Twitter hashtags about Presidential Candidates Jokowi and Prabowo were used to fetch datas from Twitter and Urls, unnecessary words, special characters and repetitive tweets were removed from the data set and stopwords in Indonesian language was applied. 250 datas for training and 100 datas for test were selected and TextBlob library was used for sentimental analysis, and it was determined that the datas were in positive, negative or neutral polarity [15]. Polarity is calculated as the difference between the total number of positive and negative words in the sentence. If the polarity is greater than zero, the sentence was considered positive, if it is small, the sentence was considered negative, and if it is zero, the sentence was considered neutral. R programming language was used in the study. By comparing the positive, negative, and neutral tweet counts of Jokowi and Prabowo, Jokowi has 40 positive, 3 negative and 10 neutral polarities, while Prabowo has 16 positive, 8 negative and 4 neutral polarities. Due to Prabowo has more negative polarity and Jokowi has more positive polarity, it was estimated that Jokowi won in the experimental result and the most used words were determined by WordCloud library for 2 candidates. While it has 3 negative and 10 neutral polarities, Prabowo has 16 positive, 8 negative and 4 neutral polarities. Since Prabowo has more negative polarity and Jokowi has more positive polarity, it was estimated that Jokowi won in the experimental result and the most used words were determined by WordCloud library for 2 candidates. While it has 3 negative and 10 neutral polarities, Prabowo has 16 positive, 8 negative and 4 neutral polarities. Due to Prabowo has more negative polarity and Jokowi has more positive polarity, it was estimated that Jokowi won in the experimental result and the most used words for 2 candidates were determined by WordCloud library.

In the study conducted to predict the French elections in 2017, Data analysis based on Twitter was performed and the election was tried to be predicted by making sentiment analysis with Tf-Idf term weighting method [16].

In a study conducted in 2019, a different strategy was followed to elect prediction with machine learning using Recurrent Neural Networks that one of the deep learning methods, rather than support vector machines or naive bayes algorithms [17].

CHAPTER 3

BACKGROUND

This study was conducted contributing the election surveys, making choices by writing people's opinions without going to the polls or predicting election results according to machine learning algorithms based on a supervised learning, comparing emotion analysis libraries and machine learning algorithms and determining the highest performing sentiment analysis techniques and machine learning models study. In this study, the tweets were taken the Twitter but because of Twitter has the limits of processing, the datas related to Donald Trump and Joe Biden were taken from Kaggle platforms, thus, data augmentation applied the study. The text classifications, sentiment analysis and machine learning algoritms applied to the tweets and election results predicted according to the tweets.
The technologies which used in this study were shown below.

3.1 Python

Python is easy to write, object-oriented, containing dynamic data types and classes where there are libraries and platforms based on scientific computational methods such as machine learning and sentiment analysis. Python has been used in most of the above-mentioned works and in this study due to its wide applicability. It’s object-oriented nature and the availability of scientific and mathematical libraries have made itself as one of the most used languages in machine learning and artificial intelligence technologies.

3.2 JSon

JSon is a data interchange format that is easy to read and write. JSon can be created as an ordered list of values, as a tag and value-based key list. The tweets captured at JSon format on Tweepy in this study.

3.3 JSonpickle

It is a Python library was created to convert Python objects to JSon format or to convert data in JSon format to an object. This library was used to convert tweets in JSon format to Python objects. The data was converted to objects and printed to the file.

3.4 Csv

Csv (Comma Separated Values) format is a very common importing and exporting format used in databases and tabular datas. The Csv module is implemented to read and write tabular data in Csv format. In this study, datas were converted to csv format using Python's csv library and the data sets related to Trump and Biden were saved as csv format files.

3.5 Tweepy

It is an easy-to-use Python library to access the Twitter API. In order to access the Twitter API, access is provided by writing the login credentials given by Twitter over the Twitter Developer account via the method in the Tweepy library. After that, tweets according to the sought hashtag are fetched. In this study, Tweepy library is used to fetch tweets from Tweepy.

3.6 Nltk

Nltk is a platform was created to be used in human-computer language interaction. Nltk is the abbreviation of the “Natural Language Toolkit”. This provides easy-to-use interfaces to more than 50 enterprise and word sources such as WordNet, with wrappers for industrial-strength NLP (Natural Language Processing) libraries, as well as text processing libraries for classification, token generation, root identification, tagging, parsing, and semantic reasoning. Stemming and Lemmatization operations were performed on data with Nltk to use on the SentiWordNet library in this study.

3.7 Textblob

TextBlob is a Python library was created to handle text data. It performs Natural language Processing (NLP) processes such as sentiment analysis and text classification. In this study, Sentiment Analysis method related to text data was used

to predict the election and it was compared with SentiWordNet.

3.8 SentiWordNet

SentiWordNet is a glossary of ideas derived from the WordNet database which a large English dictionary database where each term is associated with numerical scores showing positive and negative emotion information. In this study, the sentiment analysis method for text data sets was used to predict the election and it was compared with SentiWordNet.

3.9 Scikit-Learn

Scikit-learn is a machine learning library that implements supervised and unsupervised learning algorithms. It also performs operations such as model fitting, data pre-processing, model selection and evaluation. In this study, this library was used to preprocess the data sets and apply machine learning algorithms.

3.10 Countvectorizer

It is a method that counts the terms on the text. It is a technique that creates a matrix of terms and how many of those terms are in the text by counting the terms in the text. In this study, Countvectorizer was used in the text classification.

3.11 Tfidfvectorizer

It is a technique that gives frequency value to a term by looking at its importance in all texts as well as its frequency in the text. It creates a matrix consisting of terms on the text and frequency values given by the method applied to the terms.

$$\text{TF-IDF}(t,d) = \text{TF}(t,d) \times \log\left(N / \text{DF}(t)\right) \quad \textbf{(3.1)}$$

t =Term

d =Text - News

TF =Term frequency (number of times the tth term occurs in the news)

DF =Number of news articles containing the tth term

N =Total number of news

In this study, this was used in the text classification.

Text classification has been applied on tweets using CountVectorizer and Tfidfvectorizer.

3.12 WordCloud

WordCloud creates text images in an aesthetic way where words come together in mixed forms. The most repeated positive and negative words in tweets were visualized using WordCloud's Python plug-in.

3.13 Algorithms

3.13.1 Machine Learning

Machine Learning is the modeling of systems that make predictions by making inferences from data with mathematical and statistical operations. It is a branch of the artificial intelligence. Machine learning is the study of computer algorithms that allow computer programs to develop automatically through experience [18].

Machine Learning consists of 3 parts: Supervised Learning, Unsupervised Learning and Reinforcement Learning. Supervised learning was used in this study.

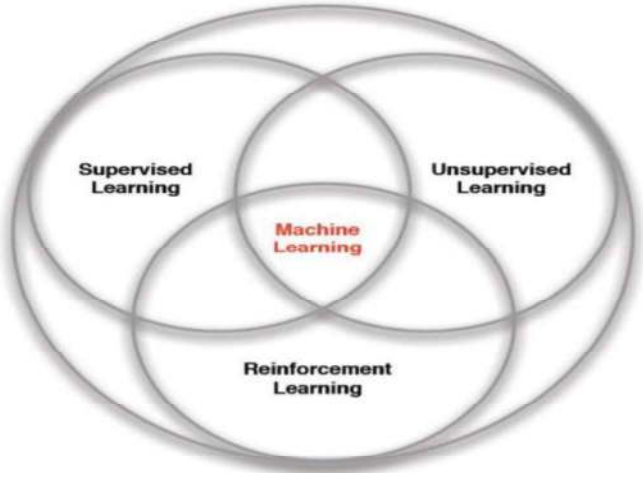

Figure 3.1 The kind of machine learning [21].

Supervised Learning

Supervised learning accounts for a lot of research activity in machine learning. Supervised learning is based on to be trained and predict labelled data.
The data is passed through a function and the output values of the inputs are estimated according to that function and the algorithm is implemented in this way.

It uses classification with a function in the form of Y = f (x) or learning techniques with regression technique.

For example, we have mail data. If we want to know that this mail data is not spam or spam, we can provide supervised learning with a classification technique using spam or non-spam labeled data on our input data.

For example, customers have a purchase price for a product. Those goods have some features of their own. If we want to predict how much the relevant good will cost according to those characteristics, we can estimate how much the relevant good will cost by the regression method by supervised learning.

In this study, classification algorithms from supervised learning algorithms were used, applied and comparative results were obtained according to positive or negative sentiment analysis. Because the output values are binary and the data are labeled in order to make a election prediction consisting of the binary output value.

Classification

Classification is a controlled machine learning approach applied to divide data into a number of different classes where we can assign tags to each class. For example, it is used to predict whether a tumor is cancerous or benign, whether Trump or Biden won the choice. It is a supervised learning algorithm that categorically separates the output values of the datas. There are many classifier types such as Logistic Regression,

Passive Agressive Classifier, Adaboost Classifier. Classification algorithms were used in this study.

3.13.2 Classification Algorithms

Logistic Regression

Logistic Regression is a linear binary classification algorithm often used for classification problems. The algorithm is capable of linearly separable classification. However, Linear Logistic Regression can not accurately classify nonlinear data, so Kernel Logistic Regression is used for nonlinear separable data classification. Logistic Regression analysis examines the relationship between a categorical dependent variable and a set of independent variables. The Logistic Regression is used when the dependent variable has only two values, such as 0 and 1 or Yes and No. The Multinomial Logistic Regression is usually used for the case where the dependent variable has three or more values.

In this study, after text vectorization with CountVectorizer or TfidfVectorizer, a binary logistic regression algorithm was applied to texts that have labeled output values from a text divided into vectors because of the output value is binary. The aim is to predict sentiment states accurately.

If it had multiple output values, the Multinomial Logistic Regression algorithm would be applied. Logistic regression uses the sigmoid function to predict binary values. All sigmoid functions have the ability to map the entire number line to a small range such as 0 to 1, so one use of the sigmoid function is to convert a real value to a value that can be interpreted as a probability. The formula of sigmoid function is:

$$S(x)=\frac{1}{1+e^{-x}} \quad \textbf{(3.2)}$$

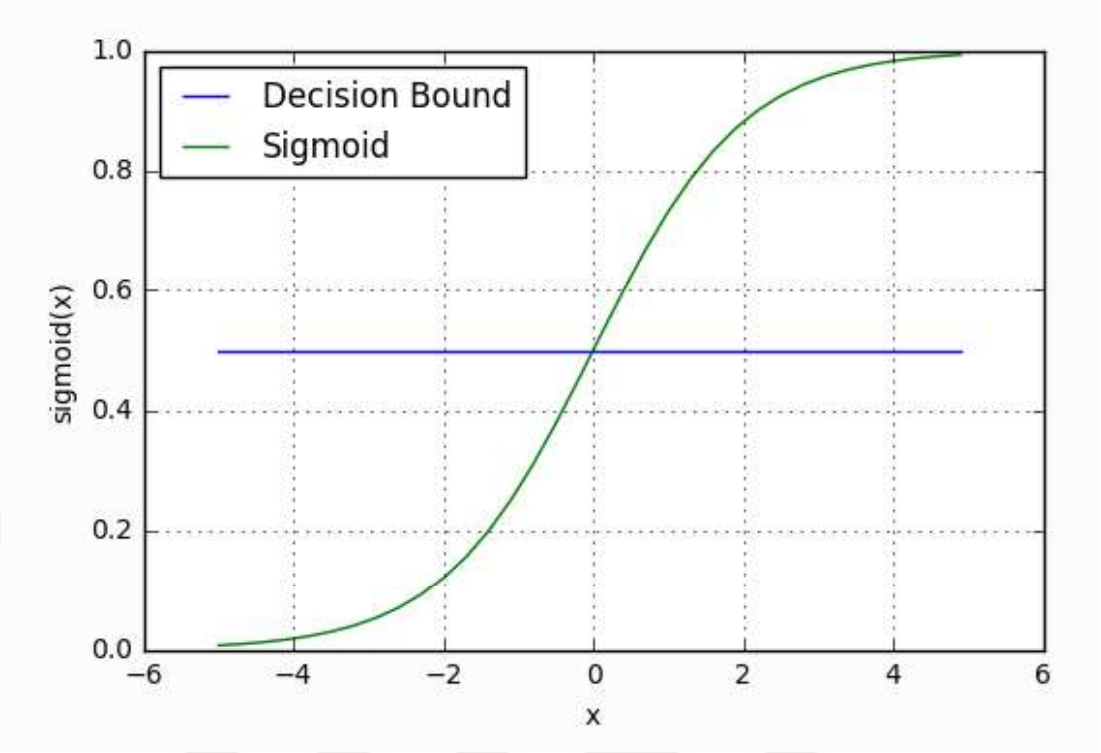

Figure 3.2 The sigmoid function graphic [22].

As it can be seen in the figure, the function makes predictions by generating a probabilistic value between 0 and 1.

Naive Bayes

Naive Bayes is a classification algorithm that is widely researched, used as a benchmark for comparisons [19].

It is used in many applications such as text classification, spam filtering, sentiment analysis, recommendation systems in real life [20]. The Naive Bayes algorithm is easy to understand and configure faster than many classification algorithms and easier to train in small data sets.

The way the algorithm works calculates the probability of each state for an element and classifies it according to the one with the highest probability value. The Naive Bayes classifier is operated by applying Bayes Theorem.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad \textbf{(3.3)}$$

($A \mid B$): The probability that "A" is true given that B is true

($B \mid A$): The probability that "B" is true given that A is true

(B): The probability that B is true

(A): The probability that A is true

There are some types of Naive Bayes algorithm:

Gaussian Naive Bayes: It is used for data that is continuous.

Multinomial Naive Bayes: It is used in multi-class categories.

Bernoulli Naive Bayes: Classes consist of only binary values (such as good and bad).

Naive Bayes works using the theorem as shown in the figure. In this study, after text vectorization with CountVectorizer or TfidfVectorizer, Bernaoulli Naive Bayes Classifier algorithm was applied to texts that have labeled output values from a vector separated text. The aim is to predict sentiment states accurately. In this study, Bernaoulli Naive Bayes technique has been used because of the positive or negative binary value can be predicted.

Passive Aggressive Classifier

Passive-Aggressive algorithms are a family of the machine learning algorithms. However, they can be very useful and efficient for certain applications.

Passive-Aggressive algorithms are often used for large-scale learning. It is an online learning algorithm. In online machine learning algorithms, the input data comes in sequence and the machine learning model is updated step by step as opposed to mass learning where the entire training dataset is used at the same time. This is very useful in situations where there is a large amount of data and it is not possible to train the entire data set computationally due to the size of the data. We can simply say that an online learning algorithm will take a training sample, update the classifier and then discard the sample.

Detecting fake news on a social media website such as Twitter, where new data is added every second, can be a very good example. To read data constantly on Twitter dynamically, the data will be very large and it would be ideal to use an online learning algorithm.

Passive-Aggressive algorithms are somewhat similar to a Perceptron model in that they do not require a learning speed. However, they contain a regularization parameter.

Adaboost Classifier

Adaboost algorithm is an association method that uses the incrementing method. It was introduced in 1995 by Yoav Freund and Robert E. Schapire. Many poor learners come together to form a community of classifiers. The purpose of the Adaboost method is to increase the success of this classifier community. In this method, training and prediction is made with the training set. The weight is increased for incorrectly predicted data and the next classifier is trained with the new weights. It is ensured that a classifier pays more attention to the data that the previous classifier predicts incorrectly.

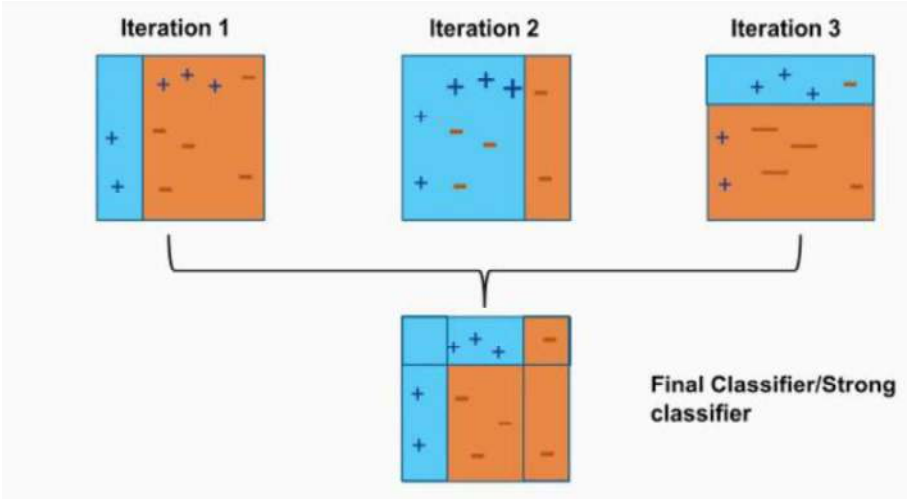

Figure 3.3 Adaboost classifier [23].

Perceptron

Perceptron is the simplest single layer neural network model. They basically consist of a single artificial nerve cell that can be trained.

Before moving on to the Perceptron topic, Artificial neural networks briefly collect information about examples, make generalizations, and then make decisions about those examples by using the information they have learned when compared to examples that they have never seen. In other words, they are computer programs that imitate biological neural networks.

The building blocks of artificial neural networks and the structure of neurons that have functions are as follows.

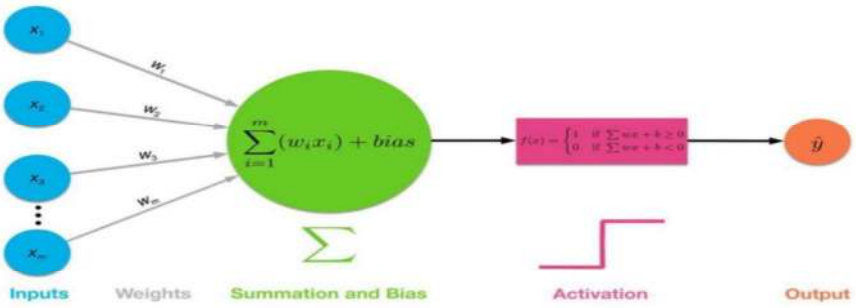

Figure 3.4 Perceptron [24].

The usage areas are control and system identification, image and voice recognition, prediction, fault analysis, medicine, communication, traffic and production management.

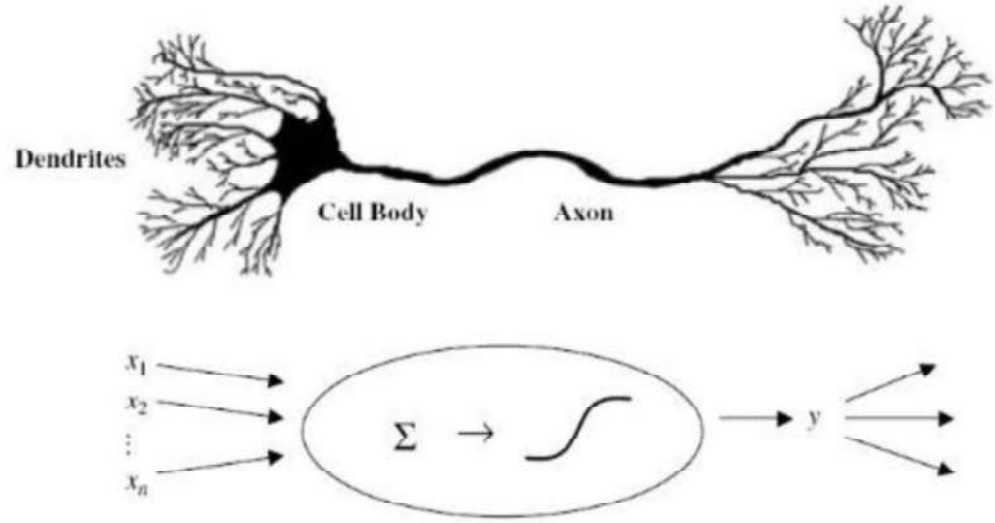

Figure 3.5 Neuron [25].

The models developed for artificial neural networks; single layer sensors are Perceptron and Adaline / Madaline.

Single Layer Sensors

Single layer neural networks consist of only input and output (O) layers. Output units are connected to all input units (X) and each link has a weight (W).

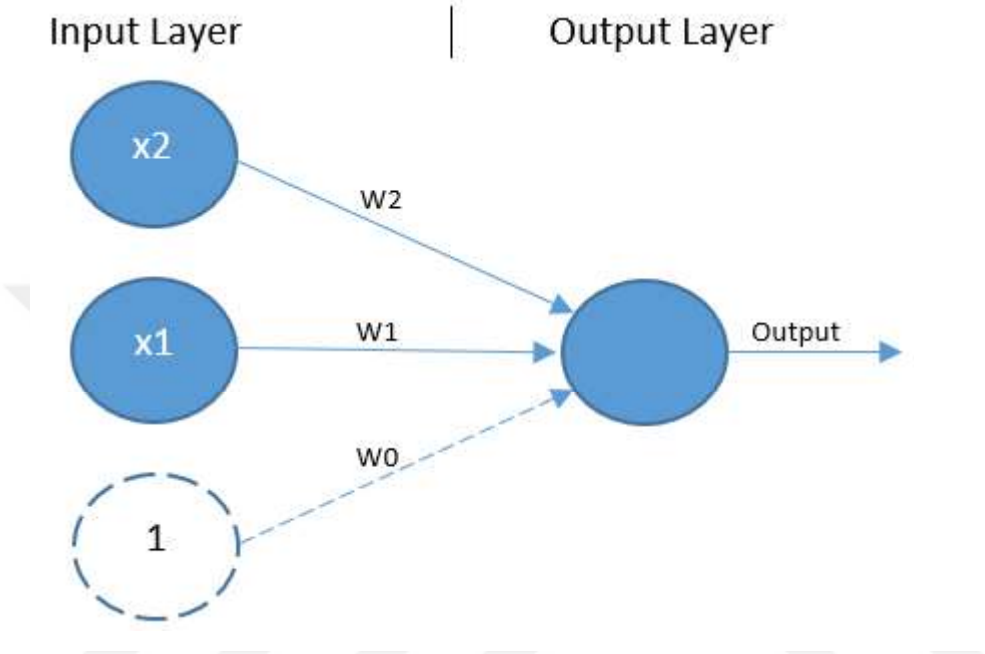

Figure 3.6 Single layer [26].

Perceptron is an algorithm implemented for the supervised learning of binary classifiers in machine learning.

A binary classifier is a function that can decide whether an input represented by a vector of numbers belongs to a particular class.

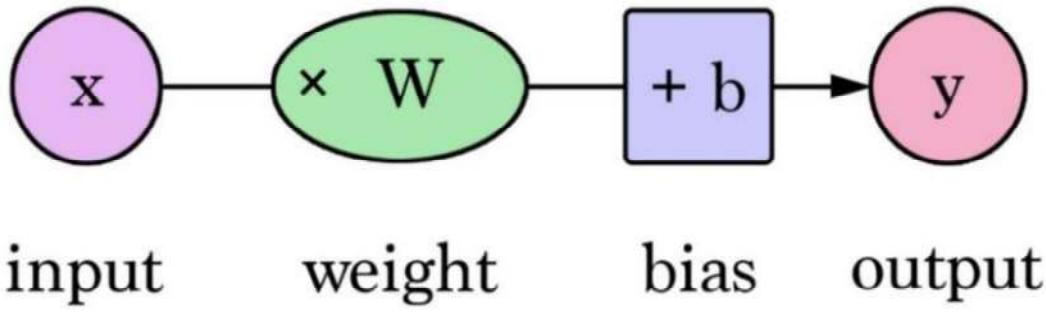

Figure 3.7 The layer parameters

The counterpart of the network described above in the artificial neural network is perceptron. In this function, as shown above, W value is defined as the weight parameter, x value as input, b value as bias and y value as output of the network., For example if we know cat pictures, x is the matrix of the cat picture and y is the score of how similar this picture is to the cat. Our parameters W weight and b bias values are used to improve this output score. In this sense, the main thing that we make in multi layer artificial neural networks or in deep learning is to calculate the w and b parameter values that will give the best score for our model.

Single layer perceptrons are subject to some limitations.

Single Layer Perceptrons cannot classify nonlinear separable data points. Complex problems involving multi layer parameters cannot be solved with Single Layer Perceptrons. Single Layer Perceptrons cannot classify nonlinear separable data points.

3.14 Performance Measures

In classification problems, a classification model is usually trained on a data set and then tested on that model. There are various measurements that measure the success rate of the test process. In the text classification set which is a two-class data set, positive comments are considered as positive and negative comments are considered as negative. This cluster was divided into two parts as training and test clusters and the

classification methods were tried to determine positive and negative sentiments.The accuracy of the predictions made by the classification method on the test set gives us the success of the method.

3.14.1 Confusion Matrix

The confusion matrix is a table showing the performance of a classification algorithm. The table shows the numbers of correctly predicted and incorrectly predicted data. The confusion matrix is used in the calculation of values such as accuracy, precision, specificity and f-measure. Matthews correlation coefficient is used to measure the performance of machine learning models.

Table 3.1 Confusion matrix

	Negative (Guess)	**Positive (Guess)**
Negative (Real)	TN	FP
Positive (Real)	FN	TP

- **TN (True Negative):** A sample that is negative in the data set is classified as negative by the classification model.

- **TP (True Positive):** A sample that is positive in the data set is classified as positive by the classification model.

- **FN (False Negative):** A sample that is positive in the data set is classified as negative by the classification model.

- **FP (False Positive):** A sample that is negative in the data set is classified as negative by the classification model.

Accuracy: The ratio of correctly classified samples to the entire data set.

$$\text{Accuracy} = (TP+TN) / (TP+TN+FP+FN) \quad \textbf{(3.4)}$$

Sensitivity -Recall: It is the ratio of correctly classified positive samples to all positive.

$$\text{Recall} = TP / (TP+FN) \quad \textbf{(3.5)}$$

Precision: It is the ratio of correctly classified positive samples to samples classified as positive.

$$\text{Precision} = TP / (TP+FP) \quad \textbf{(3.6)}$$

Specificity: The ratio of correctly classified negative samples to all negatives.

$$\text{Specificity} = TN/(TN+FP) \quad \textbf{(3.7)}$$

Error Rate: It is the erroneous classification rate.

$$\text{Error Rate} = 1-\text{Accuracy} \quad \textbf{(3.8)}$$

F-measure (F1 Score): F-measure is the harmonic mean of precision and recall value. The high F-scale indicates that the classification method works well on the positive class.

$$\text{F-measure} = (2*(\text{precision}*\text{recall})) / (\text{precision} + \text{recall}) \quad \textbf{(3.9)}$$

3.14.2 Cross Validation

Cross-validation is a statistical resampling method used to evaluate the performance of the machine learning model on data that can not see as objectively and accurately as possible.

In this study, it has been studied on K-Fold CV method which is one of the most basic cross-validation methods. In the K-Fold Cross Validation, the dataset is shuffled optional randomly, divided into k groups, selected group is used as a validation set, all other groups (k-1) group are used as train sets. The model is set up using the train set and evaluated with the validation set. The model's rating score is stored in a list. The statistical summary of the evaluation scores is checked. (such as mean, standard deviation, maximum, minimum etc.)

3.14.3 Overfitting and Underfitting

If our model has begun to work on our dataset for training and memorize it too much, or if our training set is uniform, the risk of overfitting is high. We will probably get a very low score when we show our test data to this model where we got a high score on the training set. Due to the model memorizes the situations in the training set and searches for these situations in the test data set. Since memorized situations cannot be found in the slightest change, we can get very bad prediction scores in the test data set. Models with overfitting problems show high variance and low bias.

This usually happens when the model is too complex (i.e. there are too many features / variables compared to the number of observations). This model will have very high prediction accuracy in training data, but it will not be able to predict so accurately in untrained or new data probably. This problem arises from the model's inability to generalize. Such models learn or explain the "noise" in the training data rather than the actual relationships between variables in the data.

Overfitting problem can be solved by applying the following methods;

- **Reducing the number of attributes:** Columns with high correlation to each other can be deleted or a single variable can be created from these variables by methods such as factor analysis.
- **Adding more data:** If the training set is uniform, data diversity is increased by adding more data.
- **Regularization:** Editing is a technique used to reduce the complexity of the model. It applies this by penalizing the loss function. In other words, by reducing the weight of the variables with high weight in the model, it decreases the impact rate of these variables. This method helps to solve the problem of overfitting. The loss function is the sum of the squares of the difference between the true value and the predicted value. To reduce the weight of the variables, it is necessary to increase the regularization value. The most popular Regularization methods are Lasso and Ridge techniques.
- **Underfitting:** Unlike overfitting, if a model has insufficient learning, it means that the model does not fit the training data and therefore misses trends in the data. It also means that the model cannot be generalized to new data. As we can imagine, this problem is often the result of a very simple model.

Models with underfitting problems have a high error rate in both training and test data sets. It has low variance and high bias. Instead of following training data too closely, these models ignore things that learned from training data and fail to learn the basic relationship between inputs and outputs.

It is worth noting that underfitting is not as common as overfitting. However, it is necessary to avoid both problems in data analysis.

CHAPTER 4

RESEARCH METHODOLOGY AND RESULTS

In this study, tweets related to Donald Trump and Joe Biden hashtags were taken through the Twitter API using the Tweepy library from the Twitter social media platform, but because of Twitter API can give limited datas, the data augmentation was applied using data sets from Kaggle Platform. Retweets were removed, user tags, retweet tags and numeric characters have been deleted and converted to lowercase letters. Due to Trump and Biden are the candidates, the words which were written Trump and Biden were removed in the tweets. In this way, the effect of unnecessary characters on the data was removed. When vectorization is applied to uppercase and lowercase words, the same uppercase and lowercase words are called different words. Therefore, uppercase words were converted to lowercase words. We ensure that the same word is not perceived as different words by dividing it into lowercase letters, thus reducing our processing load. Then, It can predict the percentages due to positive and negative situations and who will win the election accordingly by analyzing and classifying the data with various libraries, by applying machine learning algorithms on the datas which algorithm is successful and vectorization operations with unigram, bigram and trigram methods. We also found the comparison among these.

4.1 Architectural Design

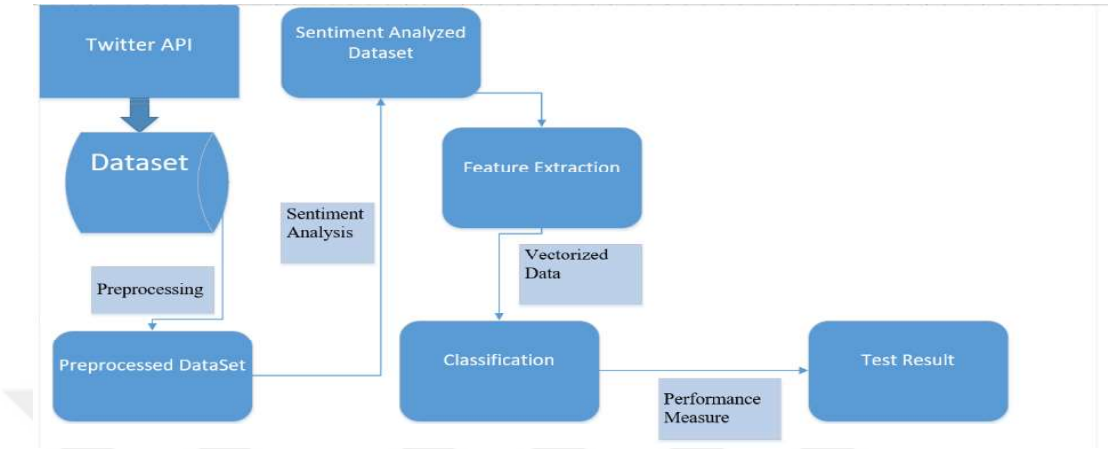

Figure 4.1 Architectural design

As seen in Figure 4.1, The tweets were taken with the Twitter API. Due to Twitter gives less data withdrawal limits, tweets about Donald Trump and Joe Biden have been received from Kaggle Platform which one of the best machine learning platforms. The data sets were preprocessed, the tweets were classified through sentiment analysis libraries after preprocessing and the feature extraction was performed by applying vectoring processes such as CountVectorizer and TfidfVectorizer and classified with machine learning algorithms and performance criteria and test results.

Among many classification methods, the machine learning classification algorithm has been applied according to the supervised learning technique and labeled accordingly.

When classifying sentiment analysis, a comparison was made between TextBlob and SentiWordNet libraries and when it was observed that SentiWordNet predicted close to the election results and was more successful when classifying between these 2 libraries. SentiWordNet library was used for sentiment analysis.

Due to the data classified as neutral have no effect on the election prediction, neutral data were not taken into account when predicting the elections, as only positive and negative data were observed to have an effect. Hence, Neutral data has been deleted.

With the WordCloud library, the most mentioned words were classified at positive and negative levels.

In order to predict the choices with machine learning algorithms, various machine learning classification algorithms within the scope of supervised learning were applied on the data set with vectoring operations and n-gram techniques and election prediction was made with machine learning.

4.2 Data Exploration

A total of approximately 520,000 data were taken including data taken on Twitter around approximately 260,000 belonging to Donald Trump and around approximately 260,000 belonging to Biden. The captured datas were taken with Kaggle.

The datas were ensured that data with different identities were captured and the number of times users retweeted those tweets was also observed, the number of data was not added to the data because the number of retweets was considered in the number of tweets, only the number of times that data was retweeted were considered.

The hashtag characters and numeric characters on the data have been removed and the data has been preprocessed by converting to lowercase letters.

The number of tweets belonging to Trump and the number of tweets belonging to Biden were equalized, Data sets were also shown in the figures below:

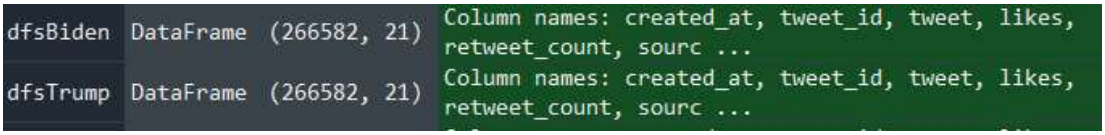

Figure 4.2 Total tweets of trump and biden

Figure 4.3 Trump's tweets

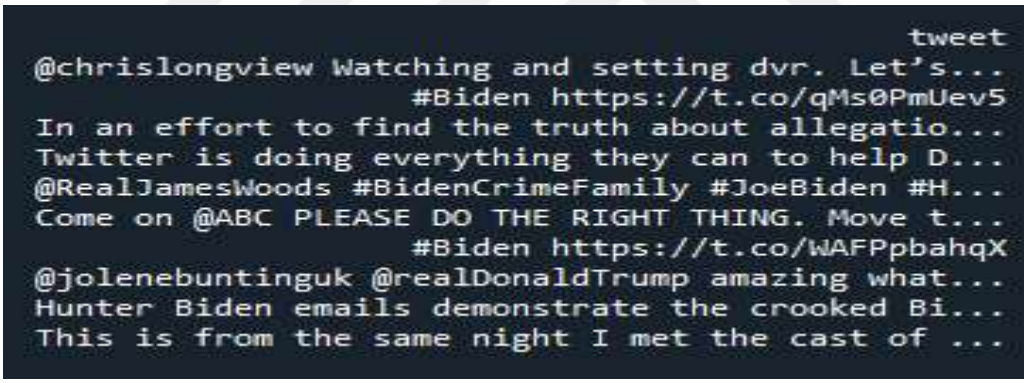

Figure 4.4 Biden's tweets

Due to Tweepy makes the tweeting feature bound to a certain limit, studies have also been carried out on a file containing 266582 thousand Trump and Biden tweets, totally 533164 tweets on Kaggle that one of the largest machine learning platforms.

If the tweets have the same username, the same or different tweets from the same user have been deleted and the data has been edited to be from different users.

Tweets were preprocessed. Html tags, username tags, hashtag tags, punctuation marks, the names and surnames of Donald Trump and Joe Biden have been removed applying regular expression operations from the tweets, tweets converted to lowercase characters to optimize text classification.

In this study, some natural language processing techniques were used and applied to study. The brief definitions of the using techniques and how this techniques were applied in this study are below.

Tokenization: It is the technique that separating sentences to words. To separate sentences to words in the tweets, this technique was used. The cause of separating sentences to words is able to apply vectorization technique and use machine learning algorithms. Tokenization was applied using NLTK (Natural Language Toolkit) library's tokenization package.

Stopwords: They are the common using words such as conjuctions. Stopwords are the unnecessary and not important in this study. It does not contribute sentiment analysis and election prediction. Therefore, Stopwords of the tweets were removed in the tweets. Removing stopwords in the tweets was applied using NLTK (Natural Language Toolkit) library's stopwords package.

Lemmatization: It is the technique that convert word to real and meaningful root word of the word. For example, we have "loved" and "lovely" words. Lemmatization converts these words to "love". In order to convert words to meaningful root word, it was applied in this study. Lemmatization was applied using NLTK (Natural Language Toolkit) library's lemmatization package.

Stemming : It is the technique that convert word to root word. It is not efficient as lemmatization. In spite of stemming is similar to lemmatization, stemming can not conver words to meaningful root word. For example we have "finally" and "finalized" words. Stemming doesn't convert these words to "final". It converts those words to

“fina”. It was constructed in this study when lemmatization could not applied and gave error. Stemming was applied using NLTK (Natural Language Toolkit) library’s stemming package.

4.3 Sentiment Analysis

Sentiment analysis libraries of the data were compared and when the SentiWordNet library was observed to be successful, sentiment analysis based on the SentiWordNet library was applied and after the data was classified according to their positive, negative or neutral states, the data labelled neutral status was removed because it would not work for us in the election prediction. Neutral states were determined with positive and negative states using SentiWordNet and TextBlob libraries. Neutral states were removed using lambda and conditional functions. Thus, the counts of the Trump’s tweets decreased 109453, the counts of the Biden’s tweet decreased 113666. In the below figures and sentiment analysis comparisons of the Trump and Biden tweets for SentiWordNet and TextBlob libraries were shown.

The classification percentages according to the positive and negative situations were calculated and the winner of the election was determined according to the positive and negative thought status based on the sentiment analysis data of Trump and Biden.

SentiWordNet gave higher accurate results than TextBlob. Because, according to 2020 USA Elections, Donald Trump has %46.9, Joe Biden has %51.4 vote percentage. Joe Biden won by a shave. SentiWordNet library gave closer results according to SentiWordNet. Therefore, SentiWordNet library used and applied in the next stages.

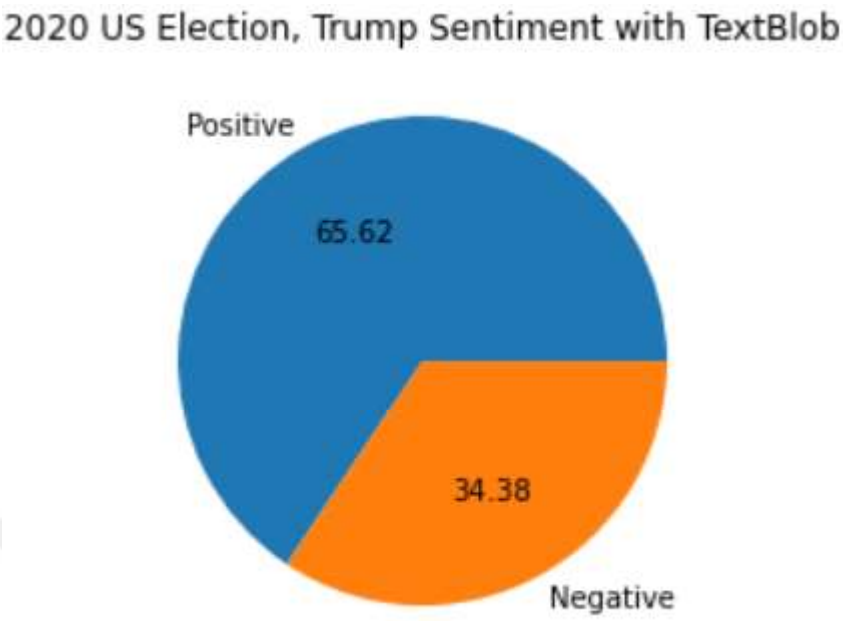

Figure 4.5 The sentiment analysis of the trump's tweets according to textblob library

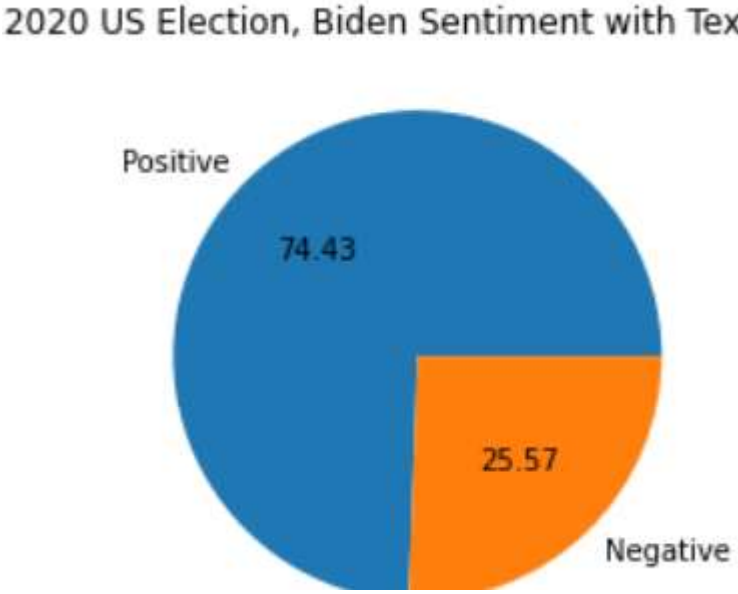

Figure 4.6 The sentiment analysis of the biden's tweets according to textblob library

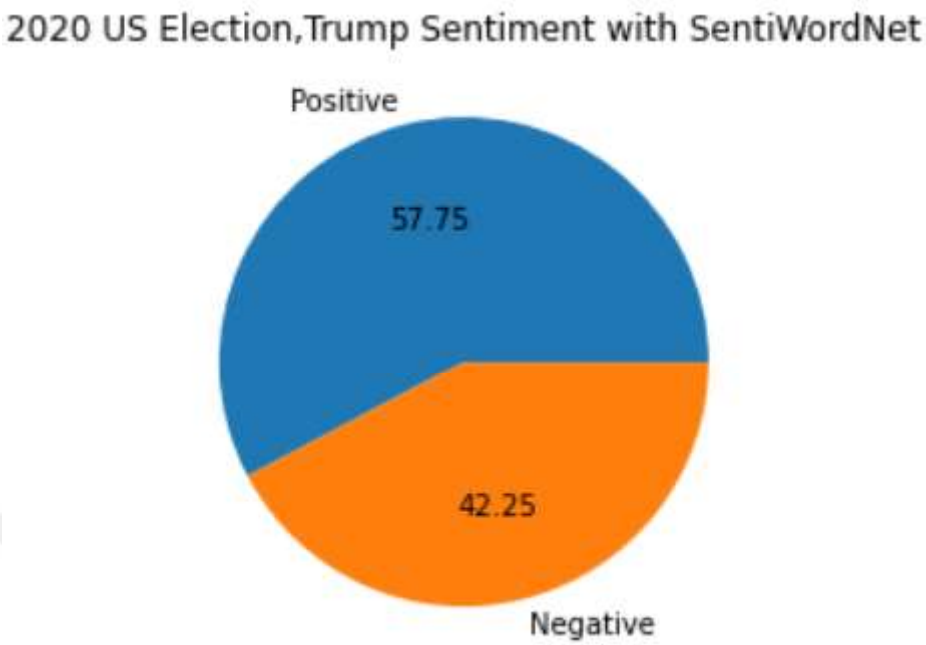

Figure 4.7 The sentiment analysis of the trump's tweets according to sentiwordnet library

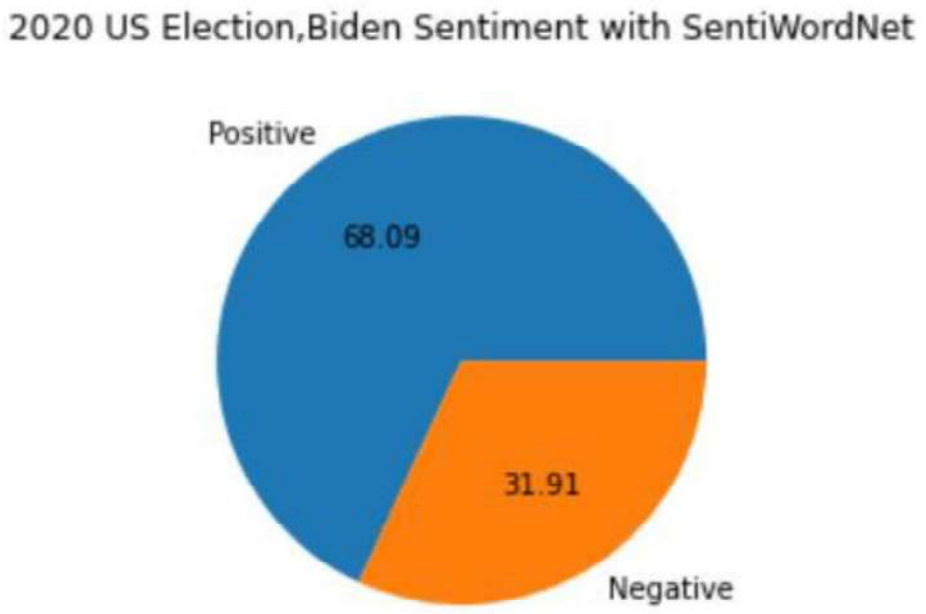

Figure 4.8 The sentiment analysis of the biden's tweets according to sentiwordnet library

Table 4.1 The results according to textblob library

	Positive	**Negative**
Trump	65.62 %	34.38 %
Biden	75.43 %	25.57 %

Table 4.2 The results according to sentiwordnet library

	Positive	**Negative**
Trump	57.75 %	42.25 %
Biden	68.09 %	31.91 %

As can be seen from the figures above, when looking at the 2020 US Election Results, the tweets about Biden should be positive as Biden won the election. During the sentiment analysis process with the TextBlob library, TextBlob showed Biden tweets 74.43 % positive and Trump's tweets 65.62 % positive. According to the data set and the process of the TextBlob library, it was determined that Biden won the election. It has shown better success than that.

In the SentiWordNet, Biden's tweets has 68.09 % positive and Trump's tweets has 57.75 %. SentiWordNet's results closer to real election results.

Since SentiWordNet is successful in the study based on the SentiWordNet library made by these studies [5,7]. If the library which was used in this study used in this study [15], this study would have been more successful. SentiWordNet was used after using stopwords, stemming and lemmatization. Because SentiWordNet was more successful than TextBlob.

For this reason, Texts classified with SentiWordNet were used in our study. CountVectorizer, TfidfVectorizer, WordCloud methods and machine learning algorithms were applied on these texts.

4.4 WordCloud

Word frequencies were aesthetically visualized with WordCloud over texts classified as positive and negative by SentiWordNet of Trump and Biden with the WordCloud transaction.

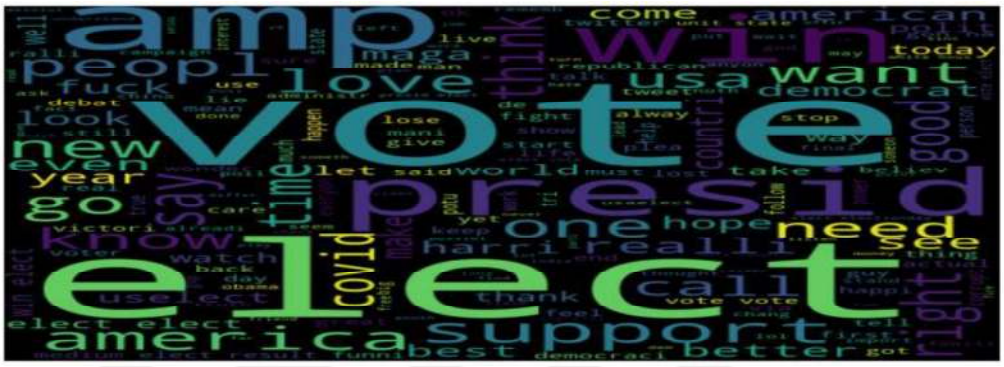

Figure 4.9 The positive words of trump's tweets

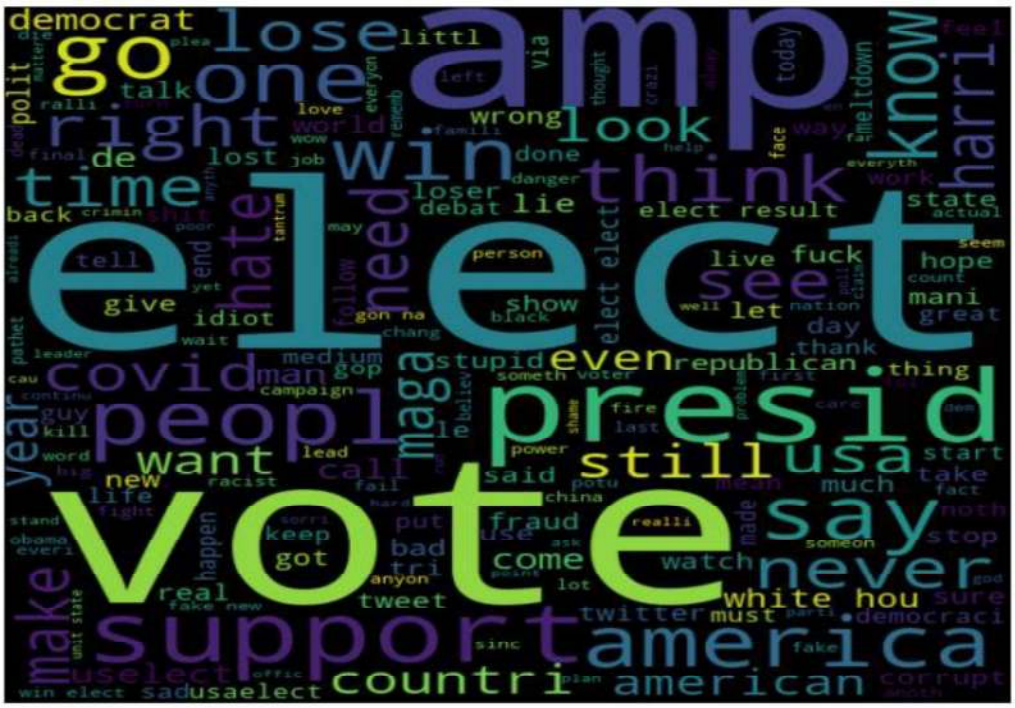

Figure 4.10 The negative words of trump's tweets

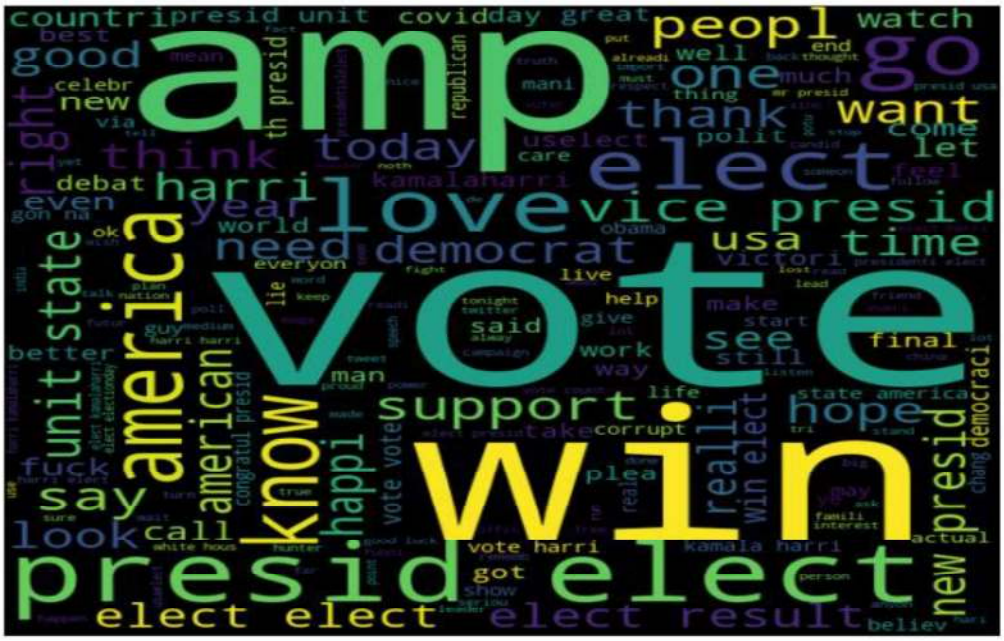

Figure 4.11 The positive words of biden's tweets

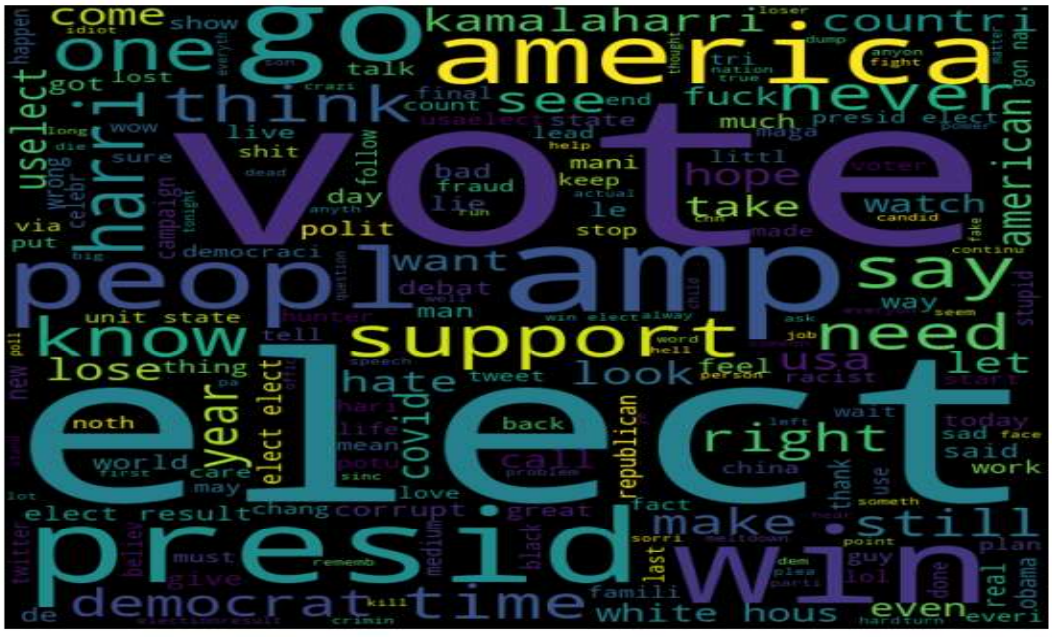

Figure 4.12 The negative words of biden's tweets

The main factor increasing Trump's voting rate and decreasing Biden's vote can be determined looking at the figures above. Thus, the factors that increase and decrease the voting rates of leaders can be known with WordCloud.

4.5 Implementation Of CountVectorizer, Tfidfvectorizer And Machine Learning Algorithms and Results

In the feature selection process, the terms in the texts were weighted by using CountVectorizer and TfidfVectorizer, the machine learning algorithms were designed to be applied and the machine learning algorithms were able to make positive or negative predictions over the matrices applied with the CountVectorizer and TfidfVectorizer techniques. Trump and Biden's tweets were handled separately.

The datas were divided into 90% training datas, %5 validation datas and %5 test datas.

In addition, 10 K-Folds and Cross validation method were applied to increase the performance.

The accuracy, recall, precision and F1 Score calculated to measure performances of the models. Accuracy is a widely used metric to measure the success of a model, but it seems to be insufficient by itself. F1 Score has also to be measured. Because, The main reason for using the F1 Score value is not making an incorrect model selection in uneven data sets. In addition, F1 Score is very important as we need a measurement metric that includes not only False Negative or False Positive but also all error costs. Since there is an uneven distribution in many real-life classification problems, it would be more appropriate to use an F1 score to evaluate our model.

The figures formed after applying Machine Learning Algorithms and vectoring processes on Trump's tweets are as follows.

4.5.1 Countvectorizer

It is a vectorization method that measures the term frequency in a text and converts the text into a term matrix. It converts terms to matrix according to the number of word association. The results of CountVectorizer according to the number of word associations and applied machine learning algorithms are given below:

When n_gram = 1:

The findings of the algorithms as a result of the CountVectorizer and machine learning algorithms applied in n_gram unigram format are as follows:

Bernaoulli Naive Bayes Algorithm:

Table 4.3 The results according to countvectorizer, unigram and bernaoulli naive bayes

Accuracy Test Score	76 %
Accuracy Train Score	83 %
Recall Test Score	86 %
Precision Test Score	76 %
F1 Test Score	81 %
Cross Validation Train Score Mean	84 %
Cross Validation Test Score Mean	76 %
Cross Validation Train Score Standard Deviation	0.0003
Cross Validation Test Score Standard Deviation	0.0032
Time (Second)	40.88

Logistic Regression Algorithm:

Table 4.4 The results according to countvectorizer, unigram and logistic regression

Accuracy Test Score	84 %
Accuracy Train Score	91 %
Recall Test Score	87 %
Precision Test Score	86 %
F1 Test Score	87 %
Cross Validation Train Score Mean	91 %
Cross Validation Test Score Mean	84 %
Cross Validation Train Score Standard Deviation	0.0004
Cross Validation Test Score Standard Deviation	0.0030
Time (Second)	117.44

AdaBoost Classification Algorithm:

Table 4.5 The results according to countvectorizer, unigram and adaboost classification

Accuracy Test Score	73 %
Accuracy Train Score	73 %
Recall Test Score	92 %
Precision Test Score	70 %
F1 Test Score	80 %
Cross Validation Train Score Mean	73 %
Cross Validation Test Score Mean	73 %
Cross Validation Train Score Standard Deviation	0.0006
Cross Validation Test Score Standard Deviation	0.0057

Time (Second)	163.41

Passive Agressive Classification Algorithm:

Table 4.6 The results according to countvectorizer, unigram and passive aggressive classification

Accuracy Test Score	80 %
Accuracy Train Score	91 %
Recall Test Score	84 %
Precision Test Score	81 %
F1 Test Score	83 %
Cross Validation Train Score Mean	91 %
Cross Validation Test Score Mean	80 %
Cross Validation Train Score Standard Deviation	0.0052
Cross Validation Test Score Standard Deviation	0.0053
Time (Second)	53.44

Perceptron Classification Algorithm:

Table 4.7 The results according to countvectorizer, unigram and perceptron classification

Accuracy Test Score	79 %
Accuracy Train Score	91 %
Recall Test Score	81 %
Precision Test Score	83 %
F1 Test Score	82 %
Cross Validation Train Score Mean	91 %

Cross Validation Test Score Mean	79 %
Cross Validation Train Score Standard Deviation	0.0059
Cross Validation Test Score Standard Deviation	0.0057
Time (Second)	51.80

When n_gram = 2:

The findings of the algorithms as a result of the CountVectorizer and machine learning algorithms applied in n_gram bigram format are as follows:

Bernaoulli Naive Bayes Algorithm:

Table 4.8 The results according to countvectorizer, bigram and bernaoulli naive bayes

Accuracy Test Score	66 %
Accuracy Train Score	90 %
Recall Test Score	97 %
Precision Test Score	63 %
F1 Test Score	77 %
Cross Validation Train Score Mean	90 %
Cross Validation Test Score Mean	65 %
Cross Validation Train Score Standard Deviation	0.0007
Cross Validation Test Score Standard Deviation	0.0061
Time (Second)	115.69

Logistic Regression Algorithm:

Table 4.9 The results according to countvectorizer, bigram and logistic regression

Accuracy Test Score	84 %
Accuracy Train Score	99 %
Recall Test Score	88 %
Precision Test Score	85 %
F1 Test Score	86 %
Cross Validation Train Score Mean	99 %
Cross Validation Test Score Mean	84 %
Cross Validation Train Score Standard Deviation	0.0001
Cross Validation Test Score Standard Deviation	0.0041
Time (Second)	641.21

AdaBoost Classification Algorithm:

Table 4.10 The results according to countvectorizer, bigram and adaboost classification

Accuracy Test Score	73 %
Accuracy Train Score	73 %
Recall Test Score	92 %
Precision Test Score	70 %
F1 Test Score	80 %
Cross Validation Train Score Mean	73 %
Cross Validation Test Score Mean	73 %
Cross Validation Train Score Standard Deviation	0.0006

Cross Validation Test Score Standard Deviation	0.0057
Time (Second)	558.93

Passive Agressive Classification Algorithm:

Table 4.11 The results according to countvectorizer, bigram and passive aggressive classification

Accuracy Test Score	83 %
Accuracy Train Score	99 %
Recall Test Score	87 %
Precision Test Score	84 %
F1 Test Score	85 %
Cross Validation Train Score Mean	99 %
Cross Validation Test Score Mean	83 %
Cross Validation Train Score Standard Deviation	0.0005
Cross Validation Test Score Standard Deviation	0.0048
Time (Second)	120.66

Perceptron Classification Algorithm:

Table 4.12 The results according to countvectorizer, bigram and perceptron classification

Accuracy Test Score	83 %
Accuracy Train Score	99 %
Recall Test Score	85 %
Precision Test Score	85 %
F1 Test Score	85 %

Cross Validation Train Score Mean	99 %
Cross Validation Test Score Mean	83 %
Cross Validation Train Score Standard Deviation	0.0003
Cross Validation Test Score Standard Deviation	0.0034
Time (Second)	120.16

When n_gram = 3:

The findings of the algorithms as a result of the CountVectorizer and machine learning algorithms applied in n_gram trigram format are as follows:

Bernaoulli Naive Bayes Algorithm:

Table 4.13 The results according to countvectorizer, trigram and bernaoulli naive bayes

Accuracy Test Score	60 %
Accuracy Train Score	89 %
Recall Test Score	99 %
Precision Test Score	59 %
F1 Test Score	74 %
Cross Validation Train Score Mean	89 %
Cross Validation Test Score Mean	60 %
Cross Validation Train Score Standard Deviation	0.0007
Cross Validation Test Score Standard Deviation	0.0041
Time (Second)	200.41

Logistic Regression Algorithm:

Table 4.14 The results according to countvectorizer, trigram and logistic regression

Accuracy Test Score	84 %
Accuracy Train Score	99 %
Recall Test Score	88 %
Precision Test Score	85 %
F1 Test Score	86 %
Cross Validation Train Score Mean	99 %
Cross Validation Test Score Mean	84 %
Cross Validation Train Score Standard Deviation	0.0001
Cross Validation Test Score Standard Deviation	0.0043
Time (Second)	1212.38

AdaBoost Classification Algorithm:

Table 4.15 The results according to countvectorizer, trigram and adaboost classification

Accuracy Test Score	73 %
Accuracy Train Score	73 %
Recall Test Score	92 %
Precision Test Score	70 %
F1 Test Score	80 %
Cross Validation Train Score Mean	73 %
Cross Validation Test Score Mean	73 %
Cross Validation Train Score Standard Deviation	0.0006

Cross Validation Test Score Standard Deviation	0.0057
Time (Second)	1576.94

Passive Agressive Classification Algorithm:

Table 4.16 The results according to countvectorizer, trigram and passive aggressive classification

Accuracy Test Score	82 %
Accuracy Train Score	99 %
Recall Test Score	87 %
Precision Test Score	84 %
F1 Test Score	85 %
Cross Validation Train Score Mean	99 %
Cross Validation Test Score Mean	83 %
Cross Validation Train Score Standard Deviation	0.0002
Cross Validation Test Score Standard Deviation	0.0029
Time (Second)	268.03

Perceptron Classification Algorithm:

Table 4.17 The results according to countvectorizer, trigram and perceptron classification

Accuracy Test Score	83 %
Accuracy Train Score	99 %
Recall Test Score	85 %
Precision Test Score	85 %
F1 Test Score	85 %
Cross Validation Train Score Mean	99 %

Cross Validation Test Score Mean	83 %
Cross Validation Train Score Standard Deviation	0.0002
Cross Validation Test Score Standard Deviation	0.0037
Time (Second)	282.41

4.5.2 Tfidfvectorizer

TfidfVectorizer is the vectorization technique that weight factor calculated by the statistical method that shows the importance of a term in the document.

When n_gram = 1:

The findings of the algorithms as a result of the TfidfVectorizer and machine learning algorithms applied in n_gram unigram format are as follows:

Bernaoulli Naive Bayes Algorithm:

Table 4.18 The results according to tfidfvectorizer, unigram and bernaoulli naive bayes

Accuracy Test Score	76 %
Accuracy Train Score	83 %
Recall Test Score	86 %
Precision Test Score	76 %
F1 Test Score	81 %
Cross Validation Train Score Mean	84 %
Cross Validation Test Score Mean	76 %
Cross Validation Train Score Standard Deviation	0.0003
Cross Validation Test Score Standard Deviation	0.0032

Time (Second)	54.24

Logistic Regression Algorithm:

Table 4.19 The results according to tfidfvectorizer, unigram and logistic regression

Accuracy Test Score	84 %
Accuracy Train Score	88 %
Recall Test Score	88 %
Precision Test Score	85 %
F1 Test Score	86 %
Cross Validation Train Score Mean	88 %
Cross Validation Test Score Mean	84 %
Cross Validation Train Score Standard Deviation	0.0004
Cross Validation Test Score Standard Deviation	0.0039
Time (Second)	114.61

AdaBoost Classification Algorithm:

Table 4.20 The results according to tfidfvectorizer, unigram and adaboost classification

Accuracy Test Score	72 %
Accuracy Train Score	73 %
Recall Test Score	92 %
Precision Test Score	70 %
F1 Test Score	79 %
Cross Validation Train Score Mean	73 %

Cross Validation Test Score Mean	73 %
Cross Validation Train Score Standard Deviation	0.0011
Cross Validation Test Score Standard Deviation	0.0060
Time (Second)	333.32

Passive Aggressive Classification Algorithm:

Table 4.21 The results according to tfidfvectorizer, unigram and passive aggressive classification

Accuracy Test Score	81 %
Accuracy Train Score	94 %
Recall Test Score	83 %
Precision Test Score	83 %
F1 Test Score	83 %
Cross Validation Train Score Mean	95 %
Cross Validation Test Score Mean	81 %
Cross Validation Train Score Standard Deviation	0.0010
Cross Validation Test Score Standard Deviation	0.0033
Time (Second)	79.77

Perceptron Classification Algorithm:

Table 4.22 The results according to tfidfvectorizer, unigram and perceptron classification

Accuracy Test Score	79 %
Accuracy Train Score	90 %

Recall Test Score	80 %
Precision Test Score	83 %
F1 Test Score	82 %
Cross Validation Train Score Mean	91 %
Cross Validation Test Score Mean	79 %
Cross Validation Train Score Standard Deviation	0.0023
Cross Validation Test Score Standard Deviation	0.0036
Time (Second)	58.67

When n_gram = 2:

The findings of the algorithms as a result of the TfidfVectorizer and machine learning algorithms applied in n_gram bigram format are as follows:

Bernaoulli Naive Bayes Algorithm:

Table 4.23 The results according to tfidfvectorizer, bigram and bernaoulli naive bayes

Accuracy Test Score	66 %
Accuracy Train Score	90 %
Recall Test Score	97 %
Precision Test Score	63 %
F1 Test Score	77 %
Cross Validation Train Score Mean	90 %
Cross Validation Test Score Mean	65 %
Cross Validation Train Score Standard Deviation	0.0007
Cross Validation Test Score Standard Deviation	0.0061
Time (Second)	150.67

Logistic Regression Algorithm:

Table 4.24 The results according to tfidfvectorizer, bigram and logistic regression

Accuracy Test Score	83 %
Accuracy Train Score	92 %
Recall Test Score	88 %
Precision Test Score	83 %
F1 Test Score	86 %
Cross Validation Train Score Mean	92 %
Cross Validation Test Score Mean	83 %
Cross Validation Train Score Standard Deviation	0.0003
Cross Validation Test Score Standard Deviation	0.0038
Time (Second)	589.37

AdaBoost Classification Algorithm:

Table 4.25 The results according to tfidfvectorizer, bigram and adaboost classification

Accuracy Test Score	73 %
Accuracy Train Score	73 %
Recall Test Score	92 %
Precision Test Score	70 %
F1 Test Score	79 %
Cross Validation Train Score Mean	73 %
Cross Validation Test Score Mean	73 %
Cross Validation Train Score Standard Deviation	0.0012

Cross Validation Test Score Standard Deviation	0.0058
Time (Second)	893.83

Passive Agressive Classification Algorithm:

Table 4.26 The results according to tfidfvectorizer, bigram and passive aggressive classification

Accuracy Test Score	83 %
Accuracy Train Score	99 %
Recall Test Score	86 %
Precision Test Score	85 %
F1 Test Score	85 %
Cross Validation Train Score Mean	99 %
Cross Validation Test Score Mean	84 %
Cross Validation Train Score Standard Deviation	0.0001
Cross Validation Test Score Standard Deviation	0.0043
Time (Second)	184.75

Perceptron Classification Algorithm:

Table 4.27 The results according to tfidfvectorizer, bigram and perceptron classification

Accuracy Test Score	82 %
Accuracy Train Score	99 %
Recall Test Score	85 %
Precision Test Score	85 %
F1 Test Score	85 %

Cross Validation Train Score Mean	99 %
Cross Validation Test Score Mean	82 %
Cross Validation Train Score Standard Deviation	0.0002
Cross Validation Test Score Standard Deviation	0.0040
Time (Second)	189.70

When n_gram = 3:

The findings of the algorithms as a result of the TfidfVectorizer and machine learning algorithms applied in n_gram trigram format are as follows:

Bernaoulli Naive Bayes Algorithm:

Table 4.28 The results according to tfidfvectorizer, trigram and bernaoulli naive bayes

Accuracy Test Score	60 %
Accuracy Train Score	89 %
Recall Test Score	99 %
Precision Test Score	59 %
F1 Test Score	74 %
Cross Validation Train Score Mean	89 %
Cross Validation Test Score Mean	60 %
Cross Validation Train Score Standard Deviation	0.0007
Cross Validation Test Score Standard Deviation	0.0041
Time (Second)	318.20

Logistic Regression Algorithm:

Table 4.29 The results according to tfidfvectorizer, trigram and logistic regression

Accuracy Test Score	82 %
Accuracy Train Score	93 %
Recall Test Score	88 %
Precision Test Score	82 %
F1 Test Score	85 %
Cross Validation Train Score Mean	93 %
Cross Validation Test Score Mean	82 %
Cross Validation Train Score Standard Deviation	0.0003
Cross Validation Test Score Standard Deviation	0.0041
Time (Second)	1292.08

AdaBoost Classification Algorithm

Table 4.30 The results according to tfidfvectorizer, trigram and adaboost classification

Accuracy Test Score	72 %
Accuracy Train Score	73 %
Recall Test Score	92 %
Precision Test Score	70 %
F1 Test Score	79 %
Cross Validation Train Score Mean	73 %
Cross Validation Test Score Mean	73 %
Cross Validation Train Score Standard Deviation	0.0017

Cross Validation Test Score Standard Deviation	0.0055
Time (Second)	1356.56

Passive Agressive Classification Algorithm:

Table 4.31 The results according to tfidfvectorizer, trigram and passive aggressive classification

Accuracy Test Score	84 %
Accuracy Train Score	99 %
Recall Test Score	87 %
Precision Test Score	85 %
F1 Test Score	86 %
Cross Validation Train Score Mean	99 %
Cross Validation Test Score Mean	84 %
Cross Validation Train Score Standard Deviation	0.0001
Cross Validation Test Score Standard Deviation	0.0041
Time (Second)	232.86

Perceptron Classification Algorithm:

Table 4.32 The results according to tfidfvectorizer, trigram and perceptron classification

Accuracy Test Score	82 %
Accuracy Train Score	99 %
Recall Test Score	86 %
Precision Test Score	83 %
F1 Test Score	85 %

Cross Validation Train Score Mean	99 %
Cross Validation Test Score Mean	83 %
Cross Validation Train Score Standard Deviation	0.0002
Cross Validation Test Score Standard Deviation	0.0040
Time (Second)	225.16

As a result of comparing models, cross validation technique was used but accuracy results were better than cross validation results and overfitting or underfitting was not seen, therefore, the accuracy results and time results are enough to compare.

There are not overfitting or underfitting in the results. The most of the results are so successful and can be applied to predict other elections.

In this study, the most efficient machine learning algorithm and vectorization technique are Passive Aggressive Classifier Algorithm and Tfidf Vectorizer technique by trigram.

In spite of logistic regression algorithm more successful narrowly than Passive Aggressive Classifier algorithm, Passive Aggressive Classifier algorithm has its online algorithm feature and feature that can be updated own model when new data added and it has short processing time. Therefore, Passive Aggressive Classifier Algorithm is the most efficient algorithm in this study.

CHAPTER 5

CONCLUSION

This study was conducted to predict election, contribute the studies of the survey companies and to be study example to election system studies which can be applied using opinions on social media without polls.
In this study, the main datas were taken from Kaggle Platform because of Twitter has restricted limits. Sentiwordnet and Textblob were applied for sentiment analysis. They compared and the best results were in the SentiWordNet library. The datas preprocessed and Countvectorizer, Tfidfvectorizer and n-gram techniques were applied to the datas.

The datas were divided three groups as training data, test data and validation datas. Training datas were used to train, test data were used to test and validation datas were used to validate.

Logistic Regression, Bernaoulli Naive Bayes, Adaboost Classifier, Passive Aggressive Classifier and Perceptron supervised learning classification algorithms were used.

N-gram techniques were applied as unigram, bigram and trigram. The vectorization techniques were applied according to the n-gram situations.

Machine learning classification algorithms were applied to the datas, compared models and the results were shown in the chapter of the research methodology and results.

Passive aggressive classification algorithms, Tfidfvectorizer technique and SentiWordNet library were observed as the most efficient methods.

REFERENCES

[1] Hatzivassiloglou, V., & McKeown, K. (1997, July). Predicting the semantic orientation of adjectives. In 35th annual meeting of the association forcomputational linguistics and 8th conference of the european chapter of the association for computational linguistics (pp. 174-181).

[2] Bae, Y., & Lee, H. (2012). Sentiment analysis of twitter audiences: Measuring the positive or negative influence of popular twitterers. Journal of the American Society for Information Science and Technology, 63(12), 2521-2535.

[3] Montoyo, A., MartíNez-Barco, P., & Balahur, A. (2012). Subjectivity and sentiment analysis: An overview of the current state of the area and envisaged developments

[4] Haddi, E., Liu, X., & Shi, Y. (2013). The role of text pre-processing in sentiment analysis. Procedia Computer Science, 17, 26-32.

[5] Denecke, K. (2008, April). Using sentiwordnet for multilingual sentiment analysis. In 2008 IEEE 24th international conference on data engineering workshop (pp. 507-512). IEEE

[6] Boiy, E., & Moens, M. F. (2009). A machine learning approach to sentiment analysis in multilingual Web texts. Information retrieval, 12(5), 526-558.

[7] Cui, A., Zhang, M., Liu, Y., & Ma, S. (2011, December). Emotion tokens: Bridging the gap among multilingual twitter sentiment analysis. In *Asia information retrieval symposium* (pp. 238-249). Springer, Berlin, Heidelberg.

[8] Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? Sentiment classification using machine learning techniques. arXiv preprint cs/0205070.

[9] Ortigosa, A., Martín, J. M., & Carro, R. M. (2014). Sentiment analysis in Facebook and its application to e-learning. Computers in human behavior, 31, 527-541.

[10] KB, S. K., Devi, V. S., Rajeev, K. K., & Bhatia, A. (2014, September). Probabilistic algorithms for election result prediction. In 2014 International Conference on Soft Computing and Machine Intelligence (pp. 79-82). IEEE.

[11] Dwi Prasetyo, N., & Hauff, C. (2015, August). Twitter-based election prediction in the developing world. In Proceedings of the 26th ACM Conference on Hypertext & Social Media (pp. 149-158).

[12] Jahanbakhsh, K., & Moon, Y. (2014). The predictive power of social media: On the predictability of us presidential elections using twitter. arXiv preprint arXiv:1407.0622.

[13] Tunggawan, E., & Soelistio, Y. E. (2016, October). And the winner is…: Bayesian Twitter-based prediction on 2016 US presidential election. In 2016 International Conference on Computer, Control, Informatics and its Applications (IC3INA) (pp. 33-37). IEEE.

[14] Attarwala, A., Dimitrov, S., & Obeidi, A. (2017, September). How efficient is twitter: Predicting 2012 us presidential elections using support vector machine via twitter and comparing against iowa electronic markets. In 2017 Intelligent Systems Conference (IntelliSys) (pp. 646-652). IEEE.

[15] Budiharto, W., & Meiliana, M. (2018). Prediction and analysis of Indonesia Presidential election from Twitter using sentiment analysis. *Journal of Big data*, *5*(1), 51.

[16] Wang, L., & Gan, J. Q. (2017, September). Prediction of the 2017 French election based on Twitter data analysis. In 2017 9th Computer Science and Electronic Engineering (CEEC) (pp. 89-93). IEEE.

[17] Tsai, M. H., Wang, Y., Kwak, M., & Rigole, N. (2019, December). A Machine Learning Based Strategy for Election Result Prediction. In 2019 International Conference on Computational Science and Computational Intelligence (CSCI) (pp. 1408-1410). IEEE.

[18] Carbonell, J. G., Michalski, R. S., & Mitchell, T. M. (1983). An overview of machine learning. In *Machine learning* (pp. 3-23). Morgan Kaufmann.

[19] Hoare, Z. (2008). Landscapes of naive Bayes classifiers. Pattern Analysis and Applications, 11(1), 59-72.

[20] Kaviani, P., & Dhotre, S. (2017). Short survey on naive bayes algorithm. *International Journal of Advance Engineering and Research Development*, *4*(11), 607-611.

[21] https://www.nosimpler.me/reinforcement-learning Received January, 18,2021.

[22] https://ml-cheatsheet.readthedocs.io/en/latest/logistic_regression.html# Received January, 19 2021.