Table of Content

Twitter Sentiment Analysis on Fuel Inflation Issue in Indonesian using Random Forest, Naïve Bayes, and Support Vector Machine

Muhamad Rikbal Ikhsani[1], Bambang Ari Wahyudi[2], Irma Palupi[3]

[1,2,3]Fakultas Informatika, Universitas Telkom, Bandung
[4]Divisi Digital Service PT Telekomunikasi Indonesia
[1]muhamadrikbal@students.telkomu niversity.ac.id, [2]bambangari@telkomuniversity.ac.id,
[3]pembimbing2@telkomuniversity.ac.id

Abstract

Sentiment analysis is a technique used to analyze the subjectivity of opinions expressed in the text. In this research, we evaluate sentiment classification methods for analyzing public opinion about fuel inflation on Twitter, including Naive Bayes, Support Vector Machine (SVM), and Random Forest. Our results show that the SVM and Random Forest methods produced the highest accuracy rates of 78%, while Naive Bayes achieved an accuracy rate of 70%. Based on these findings, the SVM and Random Forest methods are good choices for sentiment analysis on public opinion about fuel price increases on Twitter.

Keywords: sentiment classification, sentiment analysis, twitter, svm, fuel price

1. Introduction

1.1 Background of The Study

As an archipelagic country with a large population, Indonesia has high transportation needs. The transportation sector is critical to supporting people's basic needs, such as clothing, food, and shelter. The transportation sector is very closely related to energy needs, with 90% of energy in fuel oil (BBM) [1]. Currently, the public often responds and criticizes political and public leaders through social media such as Twitter. Twitter is one of the social media that has a retweet feature that users can use to re-upload information or tweets, which allows the dissemination of information on Twitter social media to be faster [2]. When people respond to fuel price increases on Twitter, there are always many pros and cons.

The exponential increase in the amount of information available on social media has made sentiment analysis increasingly important [3]. Sentiment analysis is a technique used to analyze text and determine the subjectivity of opinions expressed in the text, such as in reviews or tweets [4]. This makes it possible to tell whether the general opinion on a particular topic is positive, negative, or neutral. In government performance, sentiment analysis can determine public opinion about the current government performance. If the general opinion is negative, it may indicate that the government needs to improve their performance in certain areas. If public opinion is positive, it can indicate that the government is performing well. Sentiment analysis can be used as an aid in making decisions and determining policies by policymakers.

Many classification methods can be used in sentiment analysis, including K-Nearest Neighbor, Naïve Bayes, Support Vector Machine, Random Forest, etc. One popular method used is the Support Vector Machine (SVM) classification. In L. Mandloi and R. Patel's research, researchers obtained good accuracy in the SVM method for classifying sentiments in the text [7]. SVM is also used in the classification of Twitter comments by MA Al-Ghonaim entitled "Deep Learning and SVM-Based Sentiment Analysis of Twitter Data", with high accuracy results [8]. Other popular methods are Naïve Bayes and Random Forest. In the research of Indra Budi and Dian Arianto [5], entitled Aspect-based Sentiment Analysis on Indonesia's Tourism Destinations Based on Google Maps User Code-Mixed Reviews (Study Case: Borobudur and Prambanan Temples), the Random Forest and Naïve Bayes methods were used for analysis sentiment on tourist destination reviews on Google Maps. Then, in research conducted by Negis et al. [6] entitled "Sentiment Analysis Using A Random Forest Classifier On Turkish Web Comments", the Random Forest method is also used for sentiment analysis on comments on Turkish-language websites.

The purpose of this research is to analyze sentiment on Twitter comments regarding the issue of rising fuel prices using the three methods that have been discussed. This research will present a comparison of the effectiveness of the three methods in sentiment classification. By comparing different methods, the writer will get results regarding the ability of different approaches to distinguish sentiment on Twitter comments about the fuel price hike issue.

1.2 Topic and Limitation

This final project focuses on sentiment analysis in the case of rising fuel prices using the Support Vector Machine, Naïve Bayes and Random Forest methods. This study has limitations, namely: (1) The dataset used is a

collection of 2961 tweets in Indonesian from 2017-2022; (2) The data is divided into two categories of sentiment, namely positive and negative; (3) The selected discussion is only limited to the increase in fuel prices.

1.3 Purpose

This study aims to create naive Bayes algorithms, support vector machines and random forests to conduct sentiment analysis in cases of rising fuel prices to obtain the best algorithm.

1.4 Writing Organization

The writing organization in this study is structured as follows. The first section consists of an introduction. The second section contains an explanation of related studies from this research. The third section contains an explanation of the method used and the application of the system that has been built. The fourth section contains the test results and analysis. Then the fifth section contains the conclusions of this study.

2. Related Studies

2.1 Review Paper

References for this Final Project must be related to the topic under study.

In the sentiment analysis method using the Support Vector Machine, Naïve Bayes and Random Forest methods, many studies have tested the accuracy of the performance of these algorithms. In a study by Khartika et al. [14], the writers propose using a machine learning algorithm called Random Forest to classify reviews as positive, neutral, or negative based on different aspects of the product. They also compared the performance of Random Forest with another algorithm called Support Vector Machine (SVM) and found that Random Forest has a higher accuracy rate of 97% compared to SVM. Overall, this journal presents a method to more efficiently analyze and understand customer feelings in online reviews, which can benefit both customers and companies.

. Then, the research conducted by Meylan Wongkar and Apriandy Angdresey [2] discusses a study that conducts sentiment analysis of tweets about the 2019 Indonesian presidential pair using the Python programming language. The methodology of this study includes collecting data using the Python library, text processing, data testing and learning, and using the Naive Bayes method for text classification. The Naive Bayes method is used to classify tweet sentiments as positive or negative. This study found that the Jokowi-Ma'ruf Amin pair received a positive sentiment score of 45.45% and a negative score of 54.55%. The Prabowo-Sandiaga pair received a positive sentiment score of 44.32% and a negative score of 55.68 %. This study also compares the performance of the Naive Bayes method with other methods, such as Support Vector Machine (SVM) and K-Nearest Neighbor (K-NN), and finds that the Naive Bayes method has a better accuracy rate of 80.90% compared to 75.58 % for K-NN and 63.99% for SVM. The study concludes by suggesting further research on analyzing public sentiment on the performance of Indonesia's president-elect using data from other social media platforms such as Facebook and Instagram.

This journal article describes a study that conducted sentiment analysis on tweets related to the COVID-19 pandemic in Ireland [16]. Writers collected more than 2.49 million tweets using the "Twint" tool from January 1, 2020, to December 31, 2020. To simplify the sentiment analysis process, the writers deleted identical tweets and only tweets with hashtags, URLs, etc. The study found that most tweets were published during the pandemic and that the sentiment of the tweets changed from before the pandemic to during the pandemic. The writers used two different tools for sentiment analysis, the TextBlob and the AFI sentiment analysis tool, and found that the AFI tool had a higher accuracy rate of 65.74% compared to 64.81% for TextBlob. This study concluded that the majority of tweets during the pandemic were positive.

Based on the above review, the authors conducted a sentiment analysis using TextBlob and the Naive Bayes, SVM, and Random Forest methods because many studies have tested the performance accuracy of these algorithms. Several studies have shown that the Random Forest algorithm has a higher accuracy level than SVM. In comparison, the Naive Bayes method has a higher accuracy level than K-Nearest Neighbor and SVM. Therefore, using multiple algorithms can provide more accurate results and increase efficiency in analyzing and understanding the feelings of users or other objects.

2.2 TF-IDF (Term Frequency – Inverse Document Frequency)

Term Frequency - Inverse Document Frequency or TF-IDF is an algorithmic method that is useful for calculating the weight of each word commonly used [13]. This method is also efficient and easy and has accurate results. This method will calculate the Term Frequency (TF) and Inverse Document Frequency (IDF) values for each token (word) in each document in the corpus. In simple terms, the TF-IDF method determines how often a

word appears in a document.ovide more accurate results and increase efficiency in analyzing and understanding the feelings of users or other objects.

In calculating Term Frequency (TF), several types of formulas can be used, namely:

1. Binary TF (binary TF) is a method that pays attention to whether a word exists in the document, and if there is, it is given a value of one. Otherwise, it is given a value of zero.
2. Pure TF (raw TF), the TF value is given based on the number of occurrences of a word in the document. For example, if it appears five times, the word will be worth five.
3. Logarithmic TF is to avoid dominating documents that contain few words in the query but have a high frequency.

$$tf = 1 + log\ (tf) \tag{1}$$

$$idfj = (\frac{l}{dfj}) \tag{2}$$

$$wij\ =\ tfij\ .(\frac{l}{dfj})\ ) \tag{3}$$

$$wij\ =\ tifij\ .log\ (\frac{l}{dfj}) \tag{4}$$

2.3 Naïve Bayes

The Naive Bayes method is a popular technique for sentiment analysis in text mining, known for its efficiency in classification calculation and data consistency. It is widely used for classifying tweets using different techniques like Unigram, Multinomial and Maximum Entropy. The key advantage of the Naive Bayes classification is that it provides a robust hypothesis for any given event or condition [9].

The Naive Bayes algorithm in scikit-learn has several hyperparameters that can adjust to improve the performance of the model on specific task. Here are the most commonly used hyperparameters for the MultinomialNB class:

- alpha: This is a smoothing parameter that helps to avoid the zero probabilities problem. It represents the strength of the regularization and should be set to a small value (e.g., 1e-3) to prevent overfitting.
- fit_prior: This is a Boolean parameter that specifies whether to learn the class prior probabilities or to use uniform prior probabilities. By default, it is set to True, which means that the model will learn the prior probabilities from the training data [19].

The formula for the fitted model of a Naive Bayes classifier with fit_prior=True in scikit-learn is::

$$P(y \mid x)\ =\ P(x \mid y)\ *\ P(y)\ /\ P(x) \tag{5}$$

Where:

P(y | x) is the posterior probability of class y given the input x
P(x | y) is the likelihood of x given the class y
P(y) is the prior probability of class y
P(x) is the evidence or marginal likelihood of x (which is a constant for a given input x and does not affect the classification decision)

2.4 Support Vector Machine

Support Vector Machine Is a classification method to find the best hyperplane by dividing the class into the input space. The basic principle of a Support Vector Machine is linear classification. The first thing that underlies understanding classification with SVM is finding the optimal line (hyperplane). This serves to separate two different data classes, namely positive (+1) and negative (-1) [11].

The SVC class in scikit-learn provides a number of parameters that can be used to configure the behavior of the SVM algorithm. Some of the most commonly used parameters are:

- C: the penalty parameter for the error term. It controls the trade-off between achieving a low training error and a low testing error, and can be used to control the degree of regularization applied to the model. Higher values of C allow for more complex decision boundaries, while lower values of C encourage simpler decision boundaries.
- kernel: the kernel function used to map the input data to a higher-dimensional space. The most commonly used kernel functions are linear, rbf, poly, and sigmoid. Each kernel has its own set of hyperparameters that can be tuned to achieve better performance.
- probability: a boolean parameter that controls whether to enable probability estimates. When this parameter is set to True, the predict_proba method can be used to obtain class probabilities, in addition to the predicted class labels.

In summary, *C* controls the regularization strength of the model*, kernel* determines the shape of the decision boundary, and *probability* allows for obtaining class probabilities in addition to class predictions [19].

However, there are many more hyperparameters available that can be used to further customize the behavior of the SVM algorithm.

2.5 Random Forest

Random Forest is a machine learning algorithm used to classify large data sets. Because its function can be used for many dimensions with various scales and high performance, this classification is done by merging the trees in the decision tree employing existing training datasets [12].

Random forest produces accurate and stable predictions by applying the bagging method (bootstrapped aggregation). Bagging is a technique that collects several meta-algorithms to improve the accuracy of the machine learning algorithm. Bagging takes random samples from the dataset. The original data is taken as a sample through raw sampling. Then, the sample obtained from the raw sampling is replaced. This process is called bootstrapping and produces a bootstrapped sample [10]. Random forests are known to have high accuracy in classification tasks compared to other models. They are often one of the best performing models in terms of prediction accuracy [17].

The RandomForestClassifier in scikit-learn comes with a set of default hyperparameters, which are used when the user does not specify any hyperparameters. the following default hyperparameters:

- n_estimators : the number of decision trees in the forest (default is 100)
- max_depth : the function used to measure the quality of a split (default is "gini")
- min_samples_split : the maximum depth of each decision tree (default is 2)
- min_samples_leaf : the minimum number of samples required to be at a leaf node (default is 1)
- max_features : the number of features to consider when looking for the best split (default is "auto", meaning max_features=sqrt(n_features))
- bootstrap : whether to sample the data with replacement when building trees (default is True) [19].

2.6 TextBlob Library

TextBlob is a powerful Python library designed for the efficient processing of textual data. With its straightforward API, TextBlob simplifies the often complex and nuanced task of natural language processing (NLP), offering a range of functionality such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and beyond [18].

By utilizing the natural language processing capabilities of TextBlob, it is possible to determine the polarity and subjectivity of a given text. This information can be utilized to support decision-making processes related to text classification and labeling. Specifically, after importing the TextBlob library and creating a TextBlob object for the text in question, the sentiment method can be called to generate polarity and subjectivity scores. These scores can then be analyzed to make informed decisions about how to appropriately label the text. It should be noted, however, that the accuracy and applicability of TextBlob's pre-trained sentiment analysis model may be limited in certain contexts and should be evaluated accordingly.

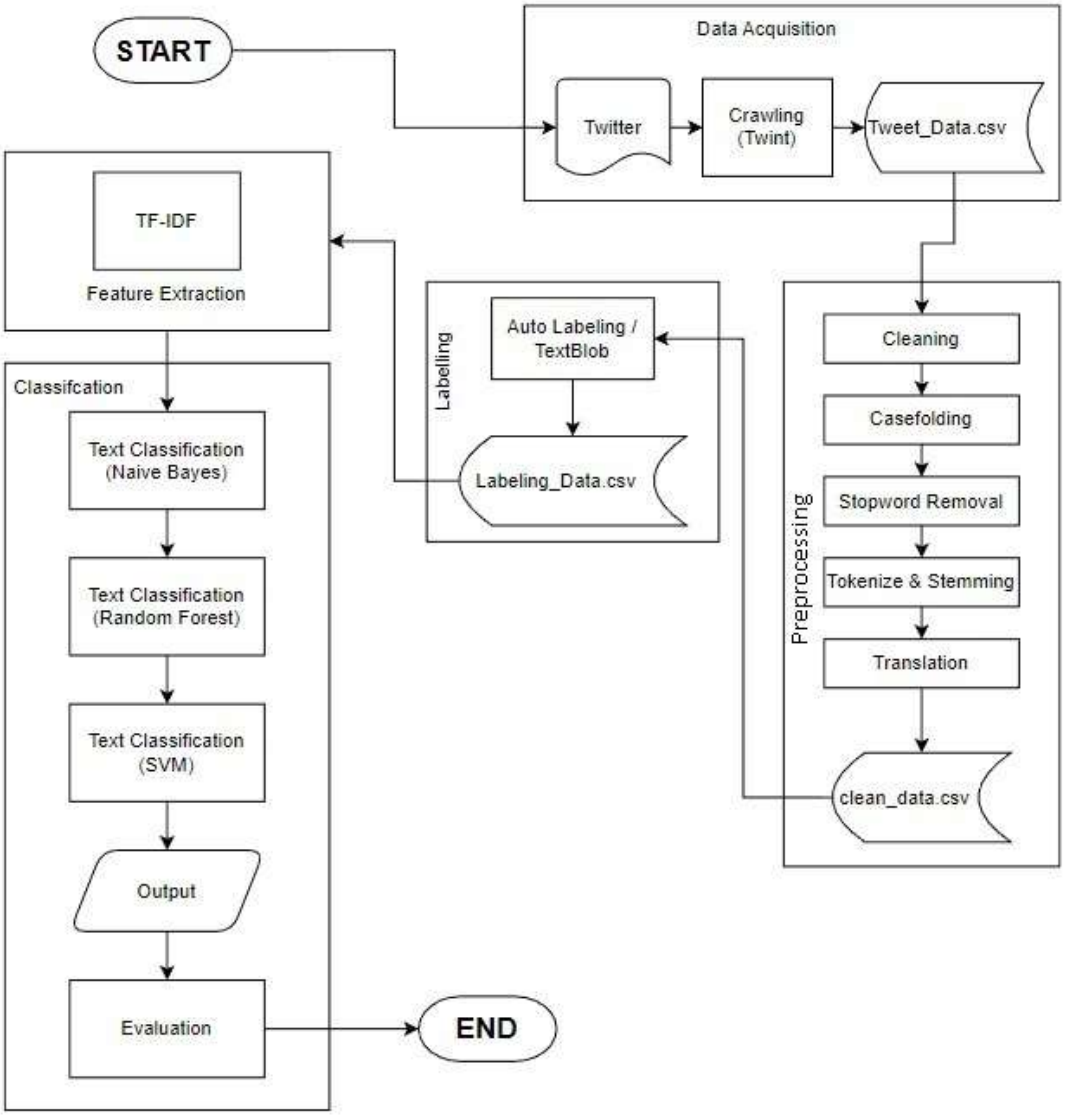
In the realm of sentiment analysis, the polarity score is a floating point number between -1.0 and 1.0 that represents the overall sentiment of a text. A higher polarity value signifies a more positive sentiment, whereas a lower polarity value indicates a more negative sentiment. This metric can be useful in various applications, such as determining the sentiment of customer feedback or social media posts [18].

3. Built System

This sentiment analysis system includes several processes, starting with data collection, where data is taken from Twitter. Then comes the preprocessing stage, where the data is adjusted into several stages including cleaning from numbers and punctuation. After that is the labeling stage, which determines the process of determining positive and negative tweets, followed by feature extraction and finally the classification stage which will be further explained.

This is a system for performing sentiment analysis testing. the system will be explained in the Image 3.1

Image 3. 1 System Design



3.1 Data Collection

To collect tweets originating from Twitter, the author uses an open-source program called Twint. The official way to retrieve tweets is using the Twitter API. However, this requires a license and can take some time to obtain. Twint is an open-source alternative that can fetch tweets without a time limit and accepts input parameters such as keywords, language, and geo-location. Twint also offers configurations that can be adapted

to specific tasks [3]. The tweet data used in this study covers the topic of increasing fuel prices from 2017 to 2022. The author only uses keyword, namely “Kenaikan BBM” And “BBM Naik” or “Increasing of fuel” And “Fuel Rise”, with a total data crawled or raw data is 3111 tweets.

3.2 Preprocessing

The data preprocessing process for this study includes five stages, namely:

1. Cleaning is the process of removing numbers and punctuation.
   The results of cleaning the numbers on the tweets data are shown in table 3.1

Table 3. 1 Results of Cleaning Numbers in Data

Before Numerical Cleanup	After Numerical Cleanup
Rabu(15/11), Kristina Edita Abuk, S.Kel, Luhkan Kab. Malaka, kegiatan Pembagian BLT akibat dampak kenaikan BBM kepada Pelaku utama dan pelaku usaha di 8 desa pesisir yang berada di Kecamatan Wewiku Kab. Malaka #GiatLuhkanSatminkalGONDOL #Malaka https://t.co/6G8v2BRIMy	Rabu(/), Kristina Edita Abuk, S.Kel, Luhkan Kab. Malaka, kegiatan Pembagian BLT akibat dampak kenaikan BBM kepada Pelaku utama dan pelaku usaha di desa pesisir yang berada di Kecamatan Wewiku Kab. Malaka #GiatLuhkanSatminkalGONDOL #Malaka https://t.co/GvBRIMy

Then the data that has been removed is followed by cleaning punctuation and symbols such as periods (.), commas (,), tags (@), hashtags (#), and other punctuation marks. The results of cleaning tweets data from punctuation are shown in Table 3.2

Table 3. 2 Results of Cleaning Punctuation in Data

Before Punctuation Cleaning	After Punctuation Cleaning
Rabu(/), Kristina Edita Abuk, S.Kel, Luhkan Kab. Malaka, kegiatan Pembagian BLT akibat dampak kenaikan BBM kepada Pelaku utama dan pelaku usaha di desa pesisir yang berada di Kecamatan Wewiku Kab. Malaka #GiatLuhkanSatminkalGONDOL #Malaka https://t.co/GvBRIMy	Rabu Kristina Edita Abuk SKel Luhkan Kab Malaka kegiatan Pembagian BLT akibat dampak kenaikan BBM kepada Pelaku utama dan pelaku usaha di desa pesisir yang berada di Kecamatan Wewiku Kab Malaka GiatLuhkanSatminkalGONDOL Malaka

2. Casefolding: change all letters in sentences to lowercase or lowercase. The results of the case folding stage are shown in Table 3.3

Table 3. 3 Casefolding Results

Before Casefolding	After Casefolding
Rabu Kristina Edita Abuk SKel Luhkan Kab Malaka kegiatan Pembagian BLT akibat dampak kenaikan BBM kepada Pelaku utama dan pelaku usaha di desa pesisir yang berada di Kecamatan Wewiku Kab Malaka GiatLuhkanSatminkalGONDOL Malaka	rabu kristina edita abuk skel luhkan kab malaka kegiatan pembagian blt akibat dampak kenaikan bbm kepada pelaku utama dan pelaku usaha di desa pesisir yang berada di kecamatan wewiku kab malaka giatluhkansatminkalgondol malaka

3. Stopword removal: namely, the removal of meaningless words. In stopword removal, the writer uses an academic library and provides additional stopwords that are unavailable in the literary library. A list of additional stopwords is written in table 3.4.

Table 3. 4 Additional Stopword List

Additional Stopword List
['yg', 'dg', 'mbroo', 'ya', 'kok', 'klo', 'tdk', 'g', 'bkn','oh', 'deh', 'aja', 'ttg', 'aja', 'dimana', 'kenapa', 'siapa', 'bagaimana', 'sp', 'cie', 'cebong', 'klo', 'klw', 'knp','sp', 'dmn', 'jg', 'tetep', 'ttp', 'tp', 'sngt', 'sngat', 'nya', 'x', 'iii', 'doang', 'jo', 'po', 'sy', 'hny', 'cuma', 'hanya', 'dkk', 'nah', 'tld', 'blt','prettttt', 'wkwkwkwkwk', 'wkwk', 'wkwkwk', 'sprt', 'bs', 'dong', 'donk','sgt','bla', 'ga', 'sby', 'presiden', 'jokowi', 'cnnindonesia','tempodotco', 'ma', 'nang', 'pak', 'jkw', 'si','ahok', 'pk', 'bu', 'pak', 'pks', 'sby', 'esbeye', 'gw', 'w', 'ente', 'lu', 'lo', 'gue', 'gwe', 'krn', 'j', 'jg', 'aj', 'cnnindonesia', 'pemkot', 'kutai','probolinggo', 'kalimantan', 'ahoker', 'xl', 'axiata', 'detikcom', 'cnbcindonesia', 'vivacoid', 'po', 'kok', 'yawlah', 'kspgoid', 'kemenpu', 'pramonoanung', 'bpkri', 'bpkpgoid']