Jurnal Elektronik Ilmu Komputer Udayana
Volume 9 No. 2, November 2020

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Sentiment Analysis of the Enforcement of PSBB Part II in Jakarta

Gede Putra Aditya Brahmantha[a1], I Wayan Santiyasa[a2]

[a]Informatics Department
[a]informatics Department, Udayana University
Bali, Indonesia
[1]adit.hermawan333@gmail.com
[2]santiyasa67@gmail.com

Abstract

In addition to communicating, Social Media is a place to issue opinions by the public on many things that are currently taking place, Twitter is one of these social medias that is widely used in conveying opinions regardless of whether these opinions are negative, positive, or even neutral. Tweets data about the Enforcement of PSBB Part II in Jakarta were obtained as many as 200 opinions using web crawling then advanced to the preprocessing stage before being classified using the K-Nearest Neighbor and Multinomial Naive Bayes algorithms. In 3 tests, the highest accuracy was 65.00% for K-Nearest Neighbor and the highest accuracy was 85.00% for Multinomial Naive Bayes method.

Keywords: *k-nearest neighbor, sentiment analysis, crawling, text mining, multinomial naive bayes*

1. Introduction

Large-Scale Social Restrictions (PSBB in Indonesian) Part II in Jakarta was announced on September 9, 2020 and enforced on September 14, 2020, have a significant impact on many communities in Jakarta. Many offices re-implement Work from Home practice for their employees and many business and companies are temporary closed down. Due to that, many people were at home. With PSBB in effect, the usage of social media is getting higher, and the internet networks is currently being used by the wider community to convey their aspirations and opinions freely, one of which is using social media [1].

Twitter is a social media microblog that allows its users to write about various topics and discuss current issues. Twitter has abundant active users, so it will provide comments or the latest information about things that are currently being discussed in the world and cause trending topics in Twitter [2].

Sentiment analysis or opinion mining is a process of understanding, extracting and processing textual data automatically to get sentiment information contained in an opinion sentence [3].

Previously related research is the Analysis of Public Opinion Sentiment on the Effects of PSBB on Twitter with the Decision Tree-KNN-Naïve Bayes Algorithm [4] written by Muhammad Syarifuddin at the enactment of the first PSBB, a collection of aspirations or comments from twitter users regarding the effects of PSBB, one of which is, can be used as an analysis of public opinion sentiment. Data regarding the effects of PSBB were obtained as many as 170 opinions, which then processed using data mining techniques, where there are text mining processes, tokenization, transformation, classification, and stem. Afterwards, it is calculated into three different algorithms to be compared, the algorithms used are Decision Tree, K-Nearest Neighbors (K-NN), and Naïve Bayes Classifier with the goal of finding the best accuracy on these classifiers. The highest result of this study is the Decision Tree algorithm with an accuracy value of 83.3%, precision 79% and a recall of 87.17%.

In this study, sentiment analysis was carried out to classify public opinion as negative or positive. This study used two different classification methods, namely K-Nearest Neighbor and Multinomial Naïve Bayes. This study also compares the accuracy of the two methods, which a better method will produce a higher accuracy.

2. Research Methods

2.1. Data Collection

In this research, secondary data is used. It is obtained by web crawling to get information in the form of tweets from people who live in Jakarta that contains the word "PSBB" which is included in the period September 9, 2020 to September 16, 2020. And this research will use 200 data divided into 100 data for the Negative Sentiment label, and 100 data for the Positive Sentiment label. Data labeling is performed manually by the author.

Table 1. Example of Dataset

Tweet	Label
‘saya cinta PSBB’	Positive
'RT @kompasiana: PSBB: Penyebaran Semakin Bertambah Banyak atau Pendapatan Semakin Berkurang Banyak? https://t.co/A6c9UVW8Rd'	Negative

2.2. Preprocessing

At this stage, preprocessing will be carried out for data which are still considered as 'dirty'. Preprocessing is carried out to process the dirty data so that the data can be cleaned and identified. The preprocessing stage consists of the case folding stage, data cleansing, language normalization, stopword removal, stemming, and tokenization like these following steps:

2.2.1. Case Folding

Case folding is the initial stage in preprocessing which goals is to change each word form to lowercase letters.

Table 2. Case Folding Process

Before Case Folding	After Case Folding
'RT @kompasiana: PSBB: Penyebaran Semakin Bertambah Banyak atau Pendapatan Semakin Berkurang Banyak? https://t.co/A6c9UVW8Rd'	‘rt @kompasiana: psbb: penyebaran semakin bertambah banyak atau pendapatan semakin berkurang banyak? https://t.co/a6c9uvw8rd’

2.2.2. Data Cleansing

Data Cleansing is the process of cleaning the text by removing irrelevant data such as usernames, hahstags, URLs, and emoticons.

Table 2. Data Cleansing Process

Before Data Cleansing	After Data Cleansing
‘rt @kompasiana: psbb: penyebaran semakin bertambah banyak atau pendapatan semakin berkurang banyak? https://t.co/a6c9uvw8rd’	psbb penyebaran semakin bertambah banyak atau pendapatan semakin berkurang banyak

p-ISSN: 2301-5373
e-ISSN: 2654-5101

2.2.3. Language Normalization

Language normalization in this research replaces common word abbreviations into the original word and replaces non-standard words into standard words.

Table 4. Example of word list for Language Normalization

Before Language Normalization	After Language Normalization
kmn	kemana
dmn	dimana
gue	saya
mo	mau
jkt	jakarta

2.2.4. Stopword Removal

A stopword is a list of unimportant and unused common words. In this process, these common words are deleted to reduce the number of words stored by the system.

Tabel 5. Stopword Removal Process

Before Stopword Removal	After Stopword Removal
psbb penyebaran semakin bertambah banyak atau pendapatan semakin berkurang banyak	psbb penyebaran bertambah atau pendapatan berkurang

2.2.5. Stemming

Stemming is replacing affixed words with basic words.

Tabel 6. Stemming Process

Before Stemming	After Stemming
psbb penyebaran bertambah atau pendapatan berkurang	psbb sebar tambah atau dapat kurang

2.2.6 Tokenization

Tokenization is the process of cutting words from a text into multiple tokens. This process will remove any spaces.

Tabel 7. Tokenization Process

Before Tokenization	After Tokenization
psbb sebar tambah atau dapat kurang	['psbb', 'sebar', 'tambah', 'atau', 'dapat', 'kurang']

2.3. Term Frequency Invers Document Frequency (TF-IDF)

Term Frequency-Inverse Document Frequency (TF-IDF) is a method used to calculate the weight of each word that has been extracted. The use of this method is generally done to count common words in information retrieval. The TF-IDF weighting model is a method that integrates the term frequency (tf) and inverse document frequency (idf) models. Term frequency (tf) is a process for counting the number of occurrences of terms in a document and inverse document frequency (idf) is used to calculate terms that appear in various documents (comments) which are considered general terms, and are considered not important [5].

The step of weighting with TF-IDF are:

a. Count term frequency *(tf)*
b. Count weighting term frequency (W_{tf})

$$W_{tf} = \begin{cases} 1 + log10tft, d, jika\ tft, d > 0 \\ 0, \end{cases}$$
(1)

c. Count document frequency (df)
d. Count the weight of inverse document frequency (idf)

$$idf_t = Log\frac{N}{df_t}$$
(2)

e. Count the weight of TF-IDF

$$W_{t,d} = Wtf_{t,d}\ x\ idf_t$$
(3)

Notes :

$tf_{t,d}$ = term frequency

$Wtf_{t,d}$ = weight of term frequency

df = the number of times the document contains a term

N = the total number of documents.

$W_{t,d}$ = weight of TF-IDF.

2.4 K-Nearest Neighbor

One of the simplest classification methods used in data mining and machine learning is K-Nearest Neighbor (KNN). It's the most accepted method of classification because of its practical convenience and efficiency: it does not require the installation of models and has been shown to have superior performance for classifying several types of data. However, the superior classification performance of the KNN is highly dependent on the metric used to calculate the pairwise distance between data points. The KNN classification rules were established by the training sample only, without any additional data. In a more complicated approach, the KNN classification finds a group of k objects in the training set that is closest to the test object, and bases the assignment of labels on the dominance of a particular class in this test environment. The K-Nearest Neighbor (KNN) algorithm is a method for classifying objects based on the

p-ISSN: 2301-5373
e-ISSN: 2654-5101

closest training example in the feature space. KNN is a type of example-based learning, or lazy learning where functions are only approached locally and all calculations are deferred up to classification [6].

The steps are carried out as follows :

a. Determine the number of k.

b. Calculate the distance of the object of each data group. The distance calculation uses the Euclidean distance equation.

$$D(x,y) = \sqrt{\sum_{1=1}^{n}(x_i - y_i)^2} \quad (4)$$

Notes :

D = Distance

x = Train data

y = Test data

c. Obtain the classification results.

2.5 Multinomial Naive Bayes

The multinomial model counts the frequency of each word that appears in the document. For example, there is a document d and a class set c. To calculate the class of document d, it can be calculated with the formula [7] :

$$P(c|term\ of\ document\ d) = P(c)\ x\ P(\ t_1|c)\ x\ P(\ t_2|c)\ x\ P(\ t_3|c)\ x\ \dots x\ P(\ t_n|c) \quad (5)$$

Notes :

$P(c)$ = Prior probability of class c.

t_n = The n-th word in document d.

$P(c|term\ of\ document\ d)$ = The probability that a document belongs to class c.

$P(\ t_n|c)$ = Probability of the n-th word known to class c.

The prior probability class c is determined by the formula:

$$P(c) = \frac{N_c}{N} \quad (6)$$

Notes :

N_c= The number of class c in the whole documents.

N = Total number of documents.

Meanwhile, the Multinomial formula used with TF-IDF word weighting as follows [7] :

$$P(t_n|c) = \frac{W_{ct}+1}{(\sum W' \in VW'_{ct})+B'} \quad (7)$$

Notes :

W_{ct} = tfidf weighting score or W of term t in class c

$\sum W' \in VW'$ = The total number of W from all terms in class c

B' = Total number of unique words of W (idf score not multiplied by tf) in all documents.

3. Research Results and Discussion

This study uses 200 data divided into 100 negative sentiments and 100 positive sentiments. From the dataset, it's divided by 80% for randomized training data and 20% for randomized test data.

The data must go through the preprocessing stage, namely changing all letters to lowercase, and then deleting irrelevant text, normalizing language, removing common words, returning words to their basic form, and finally separating sentences into words. After going through the preprocessing stage, the TF-IDF weighting was carried out following formula (3). After the weights are obtained, classification is carried out using the K-Nearest Neighbor and Multinomial Naive Bayes methods.

The results of the system evaluation are obtained by calculating the size of the correctly classified data divided by all the test data.

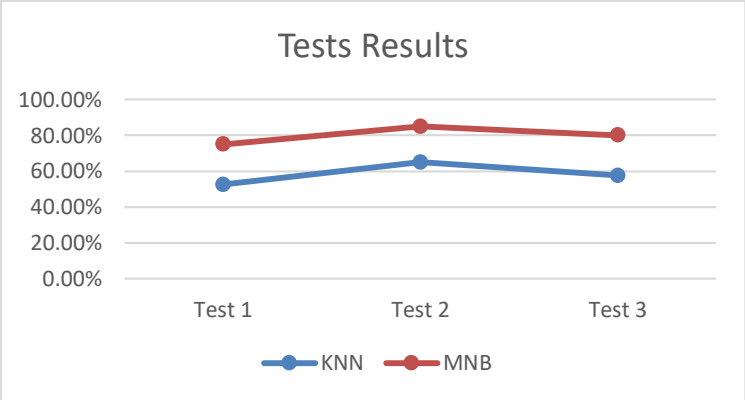

Figure 1. Graph of Test Results

In Figure 1, We conducted a number of experiments that obtained different accuracies. This is because the distribution of test data and training data is randomized from the percentages stated previously. From these results, the highest accuracy is 85.00% with the Multinomial Naive Bayes method and 65.00% with the K-Nearest Neighbor method with an average of 80.00% for Multinomial Naive Bayes and 58.33% for K-Nearest Neighbor.

4. Conclusions and Suggestions

From the results of the research that has been completed, it can be seen that the Multinomial Naive Bayes method produces higher accuracy than the K-Nearest Neighbor for classify the Sentiments of Enforcement of PSBB in Jakarta at least on this research. Furthermore, it is possible to improvise in preprocessing stage, such as a dictionary-based approach to language normalization and to experiment with changes in the k value in K-Nearest Neighbor to produce higher accuracy.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

References

[1] L. Septiani and Y. Sibaroni, “Sentiment Analysis Terhadap Tweet Bernada Sarkasme Berbahasa Indonesia,” *J. Linguist. Komputasional*, 2019, doi: 10.26418/jlk.v2i2.23.

[2] S. Fransiska, “Seri Sains dan Teknologi ANALISIS SENTIMEN TWITTER UNTUK REVIEW FILM MENGGUNAKAN ALGORITMA NAIVE BAYES CLASSIFIER (NBC) PADA SENTIMEN R Jurnal Siliwangi Vol . 5 . No . 2 , 2019 Seri Sains dan Teknologi P-ISSN 2477-3891 E-ISSN 2615-4765,” vol. 5, no. 2, 2019.

[3] G. A. Buntoro, “Analisis Sentimen Hatespeech Pada Twitter Dengan Metode Naïve Bayes Classifier Dan Support Vector Machine,” *J. Din. Inform.*, 2016, doi: 10.1016/j.cya.2015.11.011.

[4] M. Syarifuddin, “ANALISIS SENTIMEN OPINI PUBLIK TERHADAP EFEK PSBB PADA TWITTER DENGAN ALGORITMA DECISION TREE-KNN-NAÏVE BAYES,” vol. 15, no. 1, pp. 87–94, 2020, doi: 10.33480/inti.v15i1.1433.

[5] C. D. Manning, P. Raghavan, and H. Schutze, *Introduction to Information Retrieval*. 2008.

[6] A. K. Nikhath, K. Subrahmanyam, and R. Vasavi, “Building a K-Nearest Neighbor Classifier for Text Categorization,” *Int. J. Comput. Sci. Inf. Technol.*, 2016.

[7] A. Rahman, W. Wiranto, and A. Doewes, “Online News Classification Using Multinomial Naive Bayes,” *ITSMART J. Teknol. dan Inf.*, vol. 6, no. 1, pp. 32–38, 2017, doi: 10.20961/ITSMART.V6I1.11310.

This page is intentionally left blank