JOIN (Jurnal Online Informatika)
Volume 6 No.1 | June 2021: 96-102
DOI: 10.15575/join.v6i1.670
p-ISSN: 2528-1682
e-ISSN: 2527-9165

Sentiment Analysis about Large-Scale Social Restrictions in Social Media Twitter Using Algoritm K-Nearest Neighbor

Ikhsan Romli[1], Shanti Prameswari R[2], Antika Zahrotul Kamalia[3]
[1]Industrial Engineering,[2,3]Information Technology, Faculty of Engineering, Pelita Bangsa University, Indonesia

Article Info

Article history:

Received December 03, 2020
Revised February 12, 2021
Accepted February 15, 2021
Published June 17, 2021

Keywords:

Sentiment Analysis
K-Nearest Neighbor
Euclidean Distance
Cosine Similarity
Manhattan Distance

ABSTRACT

Sentiment analysis is a data processing to recognize topics that people talk about and their sentiments toward the topics, one of which in this study is about large-scale social restrictions (PSBB). This study aims to classify negative and positive sentiments by applying the K-Nearest Neighbor algorithm to see the accuracy value of 3 types of distance calculation which are cosine similarity, euclidean, and manhattan distance for Indonesian language tweets about large-scale social restrictions (PSBB) from social media twitter. With the results obtained, the K-Nearest Neighbor accuracy by the Cosine Similarity distance 82% at k = 3, K-Nearest Neighbor by the Euclidean Distance with an accuracy of 81% at k = 11 and K-Nearest Neighbor by Manhattan Distance with an accuracy 80% at k = 5, 7, 9, 11, and 13. So, in this study the K-Nearest Neighbor algorithm with the Cosine Similarity Distance calculation gets the highest point.

Corresponding Author:

Ikhsan Romli,
Department of Industrial Engineering,
Universitas Pelita Bangsa,
Jl. Inspeksi Kalimalang, Tegal Danas arah Deltamas, Cikarang, Kab. Bekasi, Indonesia
Email: ikhsan.romli@pelitabangsa.ac.id

1. INTRODUCTION

In the era of technology, data is valuable. It progressively increases without end. With data we can find pattern or information which is beneficial for various fields. Data processing can be carried out easily to be identified, validated, and processed automatically. As an example, if the organization wants to know about community opinion about their products, or government who wants to know about social opinion about the proposed policy [1]. To know about those opinions, sentiment analysis can be conducted. Sentiment analysis can also include in NLP area (Natural Language Programming) which is a data processing process with the aim to identify and understand the issues that the society talked about and their sentiments towards the issues. Analysis sentiment can also be called as opinion mining which is a scientific field to analyze opinion, sentiment, appraisal, ethics, and emotion of the society towards products, services, organizations, individuals, events, issues, or topics expressed in text [2]. To conduct sentiment analysis, data which contains opinion towards an issue is needed. There are many ways to obtain that data, one of it is through social media. It is different from past in which to know about opinion can only be obtained by survey, group opinion, newspaper, radio, and television news [2]. Social media which is popularly used by the community to show their opinion is twitter with 6.43 users [3]. Twitter basically is a mobile network application which allows us to stay connected with people, colleagues, or organization with which we attractively share anything we do to everybody, whether we know them or not [4]. On social media twitter we can get various community opinions towards any issues. One of the community opinions related to implementation of policy large-scale social restrictions (PSBB) which is implemented by Indonesian Government to cope with covid-19 [5]. In this research, sentiment analysis will be carried out using an algorithm K-Nearest Neighbor, in which the K-Nearest Neighbor (KNN) method is one of the data mining techniques that is considered the top 10 techniques for data mining by classifying unknown samples based on the classification of known neighbors [6]. In other words, K-Nearest Neighbor (KNN) is a popular non-parametric text classifier that uses instance-based learning. K-Nearest Neighbor classifies text or documents based on the measurement of the similarity between two data

e-ISSN 2528-1682
p-ISSN 2527-9165

points measured by estimating the distance from the majority of the closest neighbors of each data point [7]. There are many similarity measures that can be used to determine the k nearest neighbors. The most common distances are Euclidean distance, Manhattan distance, Minkowski distance, cosine similarity and Jaccard Similarity [8]. To perform tweet data processing, a Web Scrapping technique can be done which can be called web data extraction, or even web data mining which is a construction to download, parse, and manage data from the web automatically, or in other words: end users are humans who is clicking out in a web browser and copying and then pasting the part of interest to, say, a spreadsheet, meaning that web scraping is moving a task to a computer program that can do it or run it faster, and more precisely, than humans can do [9]. There is a tool that can be used to perform Web Scrapping called twint or the Twitter Intelligence Tool which is an advanced twitter scrapping tool written in python, making it possible to retrieve tweets from a twitter profile without using the Twitter API [10]. Previously, sentiment analysis research with K-Nearest Neighbor has been carried out several times. First, a study using K-Nearest Neighbor comparing Euclidean, Cosine Similarity, Manhattan, and Tchebychev distance calculations obtained the best accuracy of 92.559% at k = 9 for K-Nearest Neighbor and calculating the distance of Cosine Similarity [11]. Furthermore, sentiment analysis research using Naïve Bayes, K-Nearest Neighbor, and Support Vector Machine with the highest accuracy uses K-Nearest Neighbor 83.33% [12]. That is why the researcher chose the K-Nearest Neighbor method because it has a good level of accuracy and the ease and simplicity of implementation. However, the K-Nearest Neighbor method does not necessarily match every dataset, because each dataset has different information, so different predictions and results will appear. So in this study, sentiment analysis will be carried out to prove that the K-Nearest Neighbor classification algorithm by comparing 3 types of distance calculations, namely: cosine similarity, euclidean and Manhattan can have good accuracy for the analysis of social media sentiment twitter related to Large-Scale Social Restrictions (PSBB). This study will use data obtained from the results of scrapping tweet on social media twitter with the keyword "PSBB". Then, preprocessing and weighting of words with TF-IDF is carried out, so that the implementation of the K-Nearest Neighbor method can be carried out by comparing 3 types of distance calculations, namely: cosine similarity, euclidean and manhattan to find out which distance has the best accuracy.

2. METHOD

In the implementation of this research, there are several steps until it finally applies the K-Nearest Neighbor classification method and testing methods. In general, these steps are described in the flow chart below:

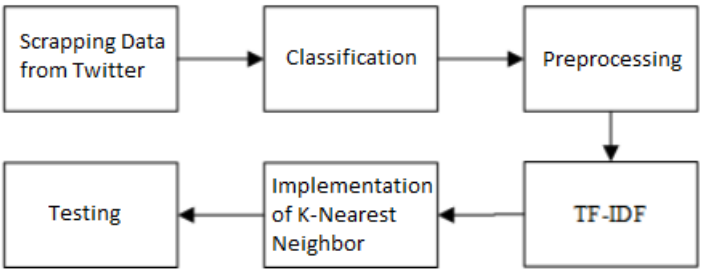

Figure 1. Research Plot

2.1. Scrapping Data From Twitter

In this process, data collection was carried out by scrapping tweet data from social media twitter using the twint tool. The scrapping results are in the form of a file in .csv format. There are 1500 tweets taken in Indonesian language containing the word 'psbb' with a retrieval time span from 07-04-2020 to 10-04-2020 which are then saved in the form of a file in .csv format with the file name 'datasetpsbb500.csv '.

2.2 Manual Classification

The dataset obtained from the scrapping results is labeled by dividing it into 2 categories, namely: negative labels and positive labels. This label assessment process is carried out manually and subjectively based on the views and experiences of the researcher and other people (friends and family). A sentence is labeled positive when it contains emotions of joy, encouragement, relaxation, suggestions and motivation. When the sentence contains emotions of sadness, anger, sarcasm, complaint and disappointment, it will be given a negative label. In addition, manual selection is also carried out to remove duplicate sentences. At this stage, only the data in the tweet column will be processed to be labeled. Data is selected manually to delete duplicate

sentences and then given a positive label with the number 1, and negative label with the number 2. The results of this manual classification are 1000 tweets with 800 positive tweets and 200 negative tweets.

2.3. Preprocessing

Furthermore, the dataset enters the preprocessing stage to transform unstructured data into structured ones [13]. At this stage there are several steps as follows:

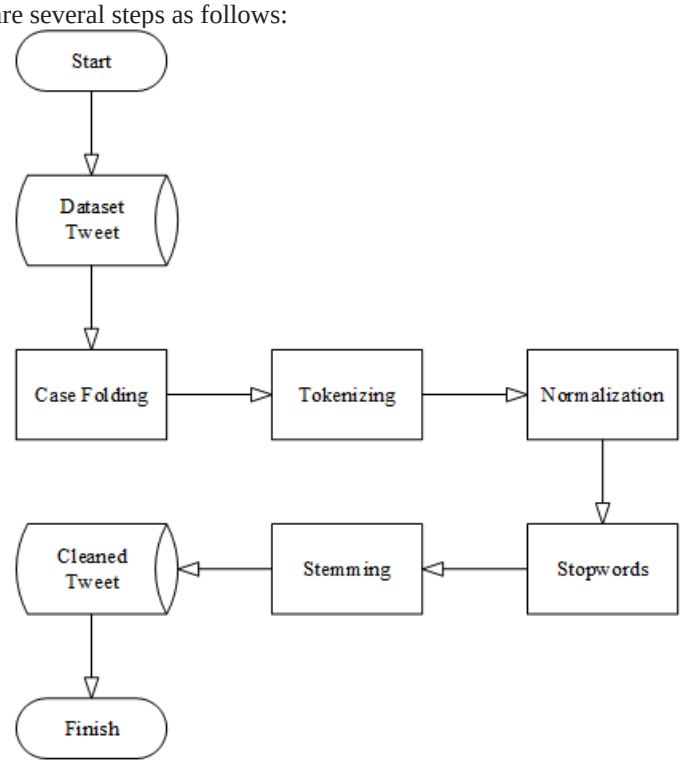

Figure 2. Preprocessing Plot

1. Case Folding

In the Case Folding process, all tweet data will be converted to lower case using the lower () function in the class of Series.str library Pandas. Example:

Table 1. Example Case Folding

Input	Output
Large-Scale Social Restrictions = Work From Homie	large-scale social restrictions = work from homie

2. Tokenizing

At this stage, splitting of words is carried out on all tweet data with word_tokenize (). In addition, at this stage, all tweet data is cleaned from words which attributes are not related to classification, such as: numbers, symbols, or whitespace. For example, a tweet is usually followed by a mention that has an attribute ('@'), ('#'), a link with an attribute ('http', 'bit.ly') and various symbolic characters such as (! $% ^ & * _ - + = [] {} \ /;:,. <> |). Example:

Table 2. Example Tokenizing

Input	Output
large-scale social restrictions = work from homie	large scale social restrictions work from homie

3. Normalization

This stage is used to uniform terms or words that have the same meaning but are written differently. Differences in terms or words can be caused by writing errors, shortening words, or "slang". The data terms for normalization are stored in the .xlxs file format created by the author. Example:

Table 3. Example Normalization

Input	Output
large scale social restrictions work from homie	large scale social restrictions work from home

4. Stopwords

At this stage, common words that usually appear in large numbers but are considered meaningless, such as 'yang', 'dan', 'dari', etc., called stopwords, will be removed from the data. Stopwords lists are saved in the .txt file format created by the author and from NLTK Stopwords for Indonesian. If the data contains words that match the stopwords list, the word will be omitted. Example:

Table 4. Example Stopwords

Input	Output
large scale social restrictions work from home	social restrictions work home

5. Stemming

At this stage the writer will use the stemmer function of the Sastrawi library to return the word to its basic form. Because the stemmer function in the Sastrawi library is slow, we can use the swifter library to speed up the stemming process on the Dataframe by running tasks in parallel. Example:

Table 5. Example Stemming

Input	Output
social restrictions work home	social restriction work home

2.4. TF-IDF

For the word weighting process in this system, the TfidfVectorizer from Sklearn is used. Example of the results of the TF-IDF process with a system using a jupyter notebook. Term Frequency is a term that refers to the number of times a term (t) appears in a document (d). Inverse Document Frequency is a measure of general or rare terms in the corpus of a particular document. TF-IDF is indicated by the following equation [14]:

$$IDF\ (w) = \log(\frac{N}{DF(w)}) \quad (1)$$

$$TF - IDF\ (w,d) = TF\ (w,d) * IDF\ (W) \quad (2)$$

Explanation:

TF-IDF (w,d) : the weight of a word throughout the document
W : word
d : document
TfF(w,d) : the frequency of a word w appears in a document
IDF (w) : inverse of DF from W
N : the total number of documents
DF (w) : number of documents containing w

2.5. K-Nearest Neighbor Implementation

After the word weighting stage with TF-IDF, the K-Nearest Neighbor was implemented by dividing 80% of the training data and 20% of the test data to calculate the distance using the Euclidean distance, calculate the Cosine Similarity distance, and calculate the Manhattan distance.

a. Cosine Similarity

Cosine Similarity can be used on spaces that have dimensions, including Euclidean space and discrete versions of Euclidean space, such as spaces where points are vectors with an integer component or a boolean component (0 or 1). In such a space, points can be thought of as directions. Then, the cosine distance between two points is the angle made by the vector to those points. This angle will be in the range of 0 to 180 degrees, regardless of the spatial dimensions [15]. Cosine Similarity is formulated as follows:

$$cosSim\ (\mathrm{x}, d_j) = \frac{\sum_{i=1}^{m} xi.d_{ji}}{\sqrt{(\sum_{i=1}^{m} {x_i}^2).(\sum_{i=0}^{m} {d_{ji}}^2)}} \quad (3)$$

Explanation:

cosSim (x,dj) = level of document similarity to a particular query
xi = the i-order term in vector for the i-order documents
dji = the i-order term in vector for i-order query
n = number of unique term in data

b. Euclidean Distance

In the Euclidean we square the distance in each dimension, add the squares, and take the positive square root defined as follows [15]:

$$distance\ (x,y) = \sqrt{\sum_{i=0}^{n-1}(x_i - y_i)^2} \quad (4)$$

explanation:
distance (x, y) : distance
xi : the i-order testing data
yi : the i-order training data

c. Manhattan Distance

Manhattan Distance is a calculation of the distance between two points x = (x1, x2, ……. Xn) and x = (y1, y2, ……. Yn) in dimensional space which adds up the distances in each dimension. Manhattan Distance has the following formula [16]:

$$distance\ (x,y) = \sum_{i=0}^{n-1} |x_i - y_i| \quad (5)$$

Explanation:
distance (x, y) : distance
xi : the i-order testing data
yi : the i-order training data

2.6. Testing

The testing of the K-Nearest Neighbor method of which each k value: 1, 3, 5, 7, 9, 11, and 13 is divided into 3 parts with comparisons of the distance calculation for Cosine Similarity, Euclidean Distance and Manhattan. The results of the accuracy of the K-Nearest Neighbor test, comparing the distances between Cosine Similarity, Euclidean Distance, and Manhattan, are known based on confusion matrix calculations. With confusion matrix calculation table, as follows:

Table 6. Confusion Matrix

Actual Class	Assigned Class	
	Positive	Negative
Positive	TP	FP
Negative	FN	TN

From the table above, it can be seen that there are categories and predictions, namely TP, FP, FN and TN. Where TP, FP, FN, and TN have the following meanings [17]:

- TP (True Positive) shows the number of test documents in category x, and true documents in category x.
- FP (False Positive) shows the number of test documents that are not in category x, and these documents should be in category x.
- FN (False Negative) shows the number of test documents in category x, and should not be in category x.
- TN (True Negative) shows the number of test documents which are not in category x, and indeed not in category x.

To find out the performance metrics, here is the calculation of precision values using the equation:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (6)$$

Recall using equations:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (7)$$

F-measure using equations:

$$\text{F} - \text{Measure} = (2 * \text{P}) * \frac{\text{R}}{\text{P} + \text{R}} \quad (8)$$

and accuracy using equations:

$$\text{accuracy} = \frac{\text{TP+TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (9)$$

3. RESULTS AND DISCUSSION

3.1. Data Set

At the beginning of text scrapping, a dataset containing Indonesian-language tweets said the key "psbb" was 1500 tweets. Datasets that have passed the preprocessing are selected, so that they become 1000 tweets with the following data proportions:

e-ISSN 2528-1682
p-ISSN 2527-9165

Table 7. Dataset proportions

Positive Tweet	Negative Tweet	Number
800	200	1000

3.2. Testing Process

The testing process is carried out by dividing the dataset by the same number in each counting distance into 80% training data and 20% test data. Furthermore, the implementation of K-Nearest Neighbor is carried out by calculating the Euclidean Distance, Cosine Similarity distance, and Manhattan using the python programming language with jupyter notebook. Then, the results of the comparison of tests carried out by the system with the actual classification results are shown by confusion matrix tables and performance metrics, namely precision, recall, F-measure and accuracy. Researchers used the parameters k = 1, 3, 5, 7, 9, 11, 13.

From the results of the K-Nearest Neighbor with Euclidean Distance, Cosine Similarity distance, and Manhattan distance, the highest accuracy value for K-Nearest Neighbor is obtained from Cosine Similarity distance with a value of 82% at k = 3. Whereas for K-Nearest neighbor with the Euclidean Distance has an accuracy of 81% at k = 11 and the Manhattan distance 80% at k = 5, 7, 9, 11, and 13.

Cosine Similarity has good accuracy because it has the concept of normalizing vector length and how it works to compare N-grams parallel to each other from 2 comparators [18]. In addition, Cosine Similarity is more suitable for cases that have large features because Cosine Similarity compares items with a number of features and vector lengths, in contrast to Euclidean which only approaching the same value, making it difficult to compare samples because they have the same distance but in Manhattan it can be more stable. However, if some features have a large value, then it can override other similarities [19]. The comparison of distance calculation is shown in graphic form below:

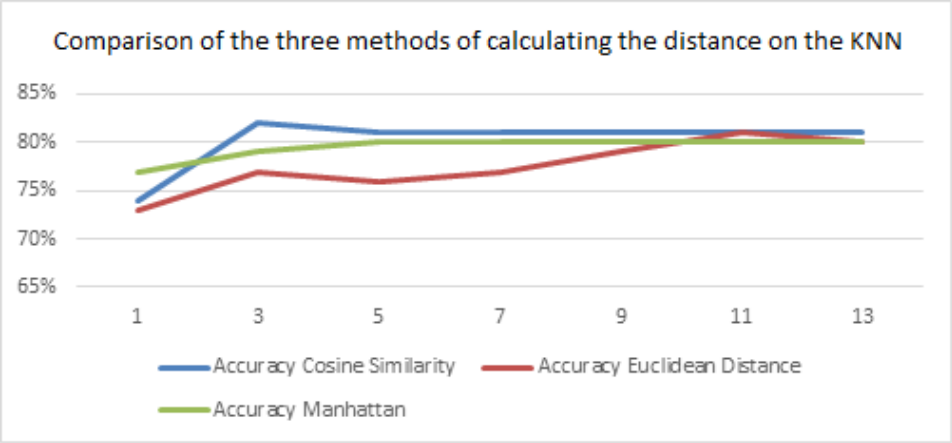

Figure 3. Graphic of the comparisons of 3 distance calculation K-Nearest Neighbor

4. CONCLUSION

To classify positive and negative sentiments on Indonesian-language tweets about large-scale social restrictions (PSBB) by comparing 3 types of distance calculations, namely: Cosine Similarity, Euclidean Distance, and Manhattan, the first thing to do is the process of data collection or tweet scrapping, then doing preprocessing data and weighting TF-IDF words then implementing K-Nearest Neighbor with Cosine Similarity distance with the highest accuracy value of 82% at k = 3, K-Nearest Neighbor with Euclidean distance with the highest accuracy value of 81% at k = 11 , K-Nearest Neighbor with the Manhattan distance with the highest accuracy of 80% at k = 5, 7, 9, 11, and 13. Thus, in this study the K-Nearest Neighbor algorithm with the Cosine Similarity distance calculation gets the highest accuracy because of how it works to make comparisons items with a number of features and a vector length.

The author realizes that this research has many shortcomings, so we expect the development of research about Large-Scale Social Restrictions in Social Media Twitter, such as:

1. multi-class classification which mean neutral label
2. add feature selection step (such as : informtion gain) or do research with balanced dataset to improve the accuracy
3. using another algorithm, such as : Support Vector Machine

4. Creating a system or application that can perform real-time sentiment analysis in order to see public sentiment in a particular area.

ACKNOWLEDGEMENTS

This research was supported/partially supported by Pelita Bangsa University. We thank our colleagues from Shanti dan Miss Antika who provided insight and expertise that greatly assisted the research, although they may not agree with all of the interpretations/conclusions of this paper.

5. REFERENCES

[1] E. Frank and I. H. Witten;, *Dara Mining; Practical Machine Learning Tools and Techniques*. United State of America: Elsevier, 2005.
[2] B. LIU, *Sentiment analysis*. USA: Cambridge University Press, 2015.
[3] We Are Social and Hootsuite, "Digital DATA OVERVEW 2019: Indonesia," *Glob. Digit. Insights*, p. 17, 2019, [Online]. Available: https://datareportal.com/reports/digital-2020-indonesia.
[4] L. Fitton, M. Gruen, and L. Poston, *Twitter for Dummies*. 2009.
[5] S. K. R. Indonesia, "Keterangan Pers Presiden RI mengenai Program Perlindungan Sosial menghadapi Dampak Pandemi COVID-19," 2020. https://setkab.go.id/program-pelindungan-sosial-menghadapi-dampak-pandemi-covid-19-31-maret-2020-di-istana-kepresidenan-bogor-provinsi-jawa-barat/ (accessed Jun. 07, 2020).
[6] P. M. P. Antonio Mucherino, Petraq J. Papajorgji, *Data Mining in Agriculture*, vol. 53, no. 9. London New York: Springer Science + Business Media, 2009.
[7] J. Samuel, G. G. M. N. Ali, M. M. Rahman, E. Esawi, and Y. Samuel, "COVID-19 public sentiment insights and machine learning for tweets classification," *Inf.*, vol. 11, no. 6, pp. 1–22, 2020, doi: 10.3390/info11060314.
[8] D. Whitenack, *using the Go Programming Language*. Packt Publishing, 2017.
[9] S. vanden Broucke and B. Baesens, *Practical Web Scraping for Data Science*. 2018.
[10] Z. C. Francesco Poldi, "Twint Project," 2018. https://github.com/twintproject (accessed Aug. 09, 2020).
[11] T. ROSA, R. PRIMARTHA, and A. WIJAYA, "Comparison of Distance Measurement Methods on K-Nearest Neighbor Algorithm For Classification," vol. 172, no. Siconian 2019, pp. 358–361, 2020, doi: 10.2991/aisr.k.200424.054.
[12] F. K. R. Mahfud, N. S. Mudawamah, and W. Hariyanto, "Sentiment Analysis of Perpustakaan Nasional Republik Indonesia Through Social Media Twitter," *Matics*, vol. 12, no. 1, p. 90, 2020, doi: 10.18860/mat.v12i1.8973.
[13] G. Miner, J. E. IV, T. Hill, R. Nisbet, D. Delen, and A. Fast, *Practical Text Mining and Statistical Analysis for Non-structured Text Data Applications*. United Kingdom: Elsevier, 2012.
[14] D. Nettleton, *Commercial Data Mining: Processing, Analysis and Modeling for Predictive Analytics Projects*. 2014.
[15] Anand Rajaraman and J. D. Ullman, *Mining of Massive Datasets*. Palo Alto, CA: Cambridge University Press, 2011.
[16] C. Sammut and G. I. Webb, *Encyclopedia of Machine Learning and Data Mining*. 2017.
[17] A. A. Puspitasari and E. Santoso, "Klasifikasi Dokumen Tumbuhan Obat Menggunakan Metode Improved k-Nearest Neighbor," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 2, pp. 486–492, 2018.
[18] O. Nurdiana, J. Jumadi, and D. Nursantika, "Perbandingan Metode Cosine Similarity Dengan Metode Jaccard Similarity Pada Aplikasi Pencarian Terjemah Al-Qur'an Dalam Bahasa Indonesia," *J. Online Inform.*, vol. 1, no. 1, p. 59, 2016, doi: 10.15575/join.v1i1.12.
[19] R. Layton, *Learning Data Mining with Python*. 2015.