Paper Lexicon

by Aba

Submission date: 01-Mar-2023 05:58AM (UTC-0500)
Submission ID: 2026079255
File name: Paper_SVM_LexiconBased.pdf (6.52M)
Word count: 8076
Character count: 42019

Paper Lexicon

ORIGINALITY REPORT

18%	15%	12%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Universitas International Batam Student Paper	2%
2	www.researchgate.net Internet Source	1%
3	ojs.unud.ac.id Internet Source	1%
4	www.semanticscholar.org Internet Source	1%
5	journal.lppmunindra.ac.id Internet Source	1%
6	www.ijrte.org Internet Source	1%
7	Bellatasya Unrica Nadia, Irene Anindaputri Iswanto. "Indonesian Clickbait Detection Using Improved Backpropagation Neural Network", 2021 4th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), 2021 Publication	<1%

8	Imam Suyuti, Dewi Retno Sari S. "Fine-Grained Sentiment Analysis on PeduliLindungi Application Users with Multinomial Naive Bayes-SMOTE", 2022 9th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), 2022 Publication	<1 %
9	jurnal.fikom.umi.ac.id Internet Source	<1 %
10	www.hindawi.com Internet Source	<1 %
11	ojs.unikom.ac.id Internet Source	<1 %
12	Almira Diva Sanya, Lya Hulliyyatus Suadaa. "Handling Imbalanced Dataset on Hate Speech Detection in Indonesian Online News Comments", 2022 10th International Conference on Information and Communication Technology (ICoICT), 2022 Publication	<1 %
13	Rifqatul Mukarramah, Dedy Atmajaya, Lutfi Budi Ilmawan. "Performance comparison of support vector machine (SVM) with linear kernel and polynomial kernel for multiclass sentiment analysis on twitter", ILKOM Jurnal Ilmiah, 2021	<1 %

	Publication	
14	economy.okezone.com Internet Source	<1 %
15	ejournal3.undip.ac.id Internet Source	<1 %
16	voi.id Internet Source	<1 %
17	Submitted to University of Aberdeen Student Paper	<1 %
18	ejournal.gunadarma.ac.id Internet Source	<1 %
19	thesai.org Internet Source	<1 %
20	artikel.rumah123.com Internet Source	<1 %
21	ejurnal.pdsi.or.id Internet Source	<1 %
22	ejournal.nusamandiri.ac.id Internet Source	<1 %
23	join.if.uinsgd.ac.id Internet Source	<1 %
24	kumparan.com Internet Source	<1 %
25	Lecture Notes in Computer Science, 2015.	

	Publication	<1%
26	M. Laylul Mustagfirin, Giri Wahyu Wiriasto, I Made Budi Suksmadana, Indira Puteri Kinasih. "Android-Based Short Message Service Filtering using Long Short-Term Memory Classification Model", Khazanah Informatika : Jurnal Ilmu Komputer dan Informatika, 2022 Publication	<1%
27	Submitted to Udayana University Student Paper	<1%
28	Akbar Ridwan, Hilal H. Nuha, Ramanti Dharayani. "Sentiment Analysis of Floods on Twitter Social Media Using the Naive Bayes Classifier Method with the N-Gram Feature", 2022 International Conference on Data Science and Its Applications (ICoDSA), 2022 Publication	<1%
29	Submitted to President University Student Paper	<1%
30	Submitted to Universitas Sanata Dharma Student Paper	<1%
31	conference.upnvj.ac.id Internet Source	<1%
32	jurnal.peneliti.net Internet Source	<1%

33	123dok.com Internet Source	<1%
34	ejurnal.stmik-budidarma.ac.id Internet Source	<1%
35	jurnal.iaii.or.id Internet Source	<1%
36	Submitted to University of Western Sydney Student Paper	<1%
37	ojs.serambimekkah.ac.id Internet Source	<1%
38	publikasi.mercubuana.ac.id Internet Source	<1%
39	Fiki Firmansyah, Wildan Budiawan Zulfikar, Dian Sa'adillah Maylawati, Nunik Destria Arianti et al. "Comparing Sentiment Analysis of Indonesian Presidential Election 2019 with Support Vector Machine and K-Nearest Neighbor Algorithm", 2020 6th International Conference on Computing Engineering and Design (ICCED), 2020 Publication	<1%
40	Manaarul Hidayat, Rahmat Hidayat, Dwi Otik Kurniawati. "Comparison of The Use of Bigrams and Stopword Removal for Classification Using Naive Bayes (Case Study on Sentiment Analysis of By.U Internet	<1%

Users)", 2021 International Conference on Software Engineering & Computer Systems and 4th International Conference on Computational Science and Information Management (ICSECS-ICOCSIM), 2021
Publication

41	repository.uksw.edu Internet Source	<1%
42	Submitted to Sheffield Hallam University Student Paper	<1%
43	Suwarno Suwarno. "Application of the UTAUT Model for Acceptance Analysis of COBIT Implementation in E-Learning Management with Microsoft Teams on Distance Learning in Batam City", Khazanah Informatika : Jurnal Ilmu Komputer dan Informatika, 2022 Publication	<1%
44	Bin Raies, Arwa, Hicham Mansour, Roberto Incitti, and Vladimir B. Bajic. "Combining Position Weight Matrices and Document-Term Matrix for Efficient Extraction of Associations of Methylated Genes and Diseases from Free Text", PLoS ONE, 2013. Publication	<1%
45	Abdurrahim Abdurrahim, Lailis Syafa'ah, Merinda Lestandy. "Sentiment analysis of Covid-19 vaccine tweets utilizing Naïve Bayes", AIP Publishing, 2022	<1%

Publication

46	ejournal.kresnamediapublisher.com Internet Source	<1 %
47	Erika Ramadhani, Amrullah Sidiq. "Design Thinking Method to Develop a Digital Evidence Handling Management Application", Khazanah Informatika : Jurnal Ilmu Komputer dan Informatika, 2022 Publication	<1 %
48	Rachma Indira, Warih Maharani. "Personality Detection on Social Media Twitter Using Long Short-Term Memory with Word2Vec", 2021 IEEE International Conference on Communication, Networks and Satellite (COMNETSAT), 2021 Publication	<1 %
49	ilmudata.org Internet Source	<1 %
50	Andrea Rahmadanisya, Erwin Budi Setiawan, Didit Adytia. "The Influence of Sentiment on Bank Mandiri (BMRI) Stock Movements Using Feature Expansion with Word2vec and Support Vector Machine Classification", 2022 10th International Conference on Information and Communication Technology (ICoICT), 2022 Publication	<1 %

51	Evi Dewi Sri Mulyani, Dani Rohpandi, Fityan Atqia Rahman. "Analysis Of Twitter Sentiment Using The Classification Of Naive Bayes Method About Television In Indonesia", 2019 1st International Conference on Cybernetics and Intelligent System (ICORIS), 2019 Publication	<1%
52	Luh Gede Surya Kartika, Putu Kussa Laksana Utama, I Dewa Gede Budiastawa, Komang Rinartha. "Comparison of the Sentiment Analysis Model's Code Complexity and Processing Time", Sinkron, 2023 Publication	<1%
53	Nur Restu Prayoga, Tresna Maulana Fahrudin, Made Kamisutara, Angga Rahagiyanto et al. "Unsupervised Twitter Sentiment Analysis on The Revision of Indonesian Code Law and the Anti-Corruption Law using Combination Method of Opinion Word and Agglomerative Hierarchical Clustering", EMITTER International Journal of Engineering Technology, 2020 Publication	<1%
54	Syamsul Rizal, Adiwijaya, Mahendra Dwifebri Purbolaksono. "Sentiment Analysis on Movie Review from Rotten Tomatoes Using Word2Vec and Naive Bayes", 2022 1st International Conference on Software	<1%

Engineering and Information Technology (ICoSEIT), 2022
Publication

55	Taufikur Rahman, Fenty Eka Muzayyana Agustin, Nurul Faizah Rozy. "Normalization of Unstructured Indonesian Tweet Text For Presidential Candidates Sentiment Analysis", 2019 7th International Conference on Cyber and IT Service Management (CITSM), 2019 Publication	<1 %
56	Utomo Pujianto, Agusta Rakhmat Taufani, Luis Devvi Ratna Kus Anggraini, Deni Sutaji. "Comparative Analysis of Bagging and Boosting Algorithms on the Classification of the Popularity of Educational-themed Youtube Videos", 2021 7th International Conference on Electrical, Electronics and Information Engineering (ICEEIE), 2021 Publication	<1 %
57	publisher.uthm.edu.my Internet Source	<1 %
58	www.e-informatyka.pl Internet Source	<1 %
59	www.image.ece.ntua.gr Internet Source	<1 %
60	www.mikroskil.ac.id Internet Source	<1 %

61	Setio Basuki, Yufis Azhar, Agus Eko Minarno, Christian Sri Kusuma Aditya, Fauzi Dwi Setiawan Sumadi, Ardiansah Ilham Ramadhan. "Detection of Reference Topics and Suggestions using Latent Dirichlet Allocation (LDA)", 2019 12th International Conference on Information & Communication Technology and System (ICTS), 2019 Publication	<1%
62	garuda.ristekbrin.go.id Internet Source	<1%
63	Majid Rahardi, Afrig Aminuddin, Ferian Fauzi Abdulloh, Rizky Adhi Nugroho. "Sentiment Analysis of Covid-19 Vaccination using Support Vector Machine in Indonesia", International Journal of Advanced Computer Science and Applications, 2022 Publication	<1%

khazanah informatika

Jurnal Ilmu Komputer dan Informatika

Combination of Support Vector Machine and Lexicon-Based Algorithm in Twitter Sentiment Analysis

Rindu Hafil Muhammadi *, Tri Ginanjar Laksana, Amalia Beladinna Arifa
Informatics Engineering
Institut Teknologi Telkom Purwokerto
Purwokerto
*rinduhafilmuhammadi@gmail.com

Abstract - Data from the Ministry of Civil Works and Public Housing (Kementrian PUPR) in 2019 shows that around 81 million millennials do not own houses. Government Regulation Number 25 of 2020 on the Implementation of Public Housing Savings, commonly called PP 25 Tapera 2020, is one of the government's efforts to ensure that Indonesian people can afford houses. Tapera is a deposit of workers for house financing, which is refundable after the term expires. Immediately after enaction, there were many public responses regarding the ordinance. We investigate public sentiments commenting on the regulation and use Support Vector Machine (SVM) for the study since it has a good level of accuracy. It also requires labels and training data. To speed up labeling, we use the lexicon-based method. The issue in the lexicon-based lies in the dictionary component as the most significant factor. Therefore, it is possible to update the dictionary automatically by combining lexicon-based and SVM. The SVM approach can contribute to lexicon-based, and lexicon-based can help label datasets on SVM to produce good accuracy. The research begins with collecting data from Twitter, preprocessing raw and unstructured data into ready-to-use data, labeling the data with lexicon-based, weighting with TF-IDF, processing using SVM, and evaluating algorithm performance model with a confusion matrix. The results showed that the combination of lexicon-based and SVM worked well. Lexicon-based managed to label 519 tweet data. SVM managed to get an accuracy value of 81.73% with the RBF kernel function. Another test with a Sigmoid kernel attains the highest precision at 78.68%. The RBF kernel has the highest recall result with a value of 81.73%. Then, the F1-score for both the RBF kernel and Sigmoid is 79.60%.

Keywords: sentiment analysis, tapera, public housing, lexicon-based, confusion matrix

Article info: *submitted July 20, 2021, revised September 20, 2021, accepted October 1, 2021.*

1. Introduction

Based on data from the Ministry of Public Works and Public Housing (Kementrian PUPR) in 2019, there were around 81 million millennial generations (or equivalent to 30% of Indonesia's population) who did not own homes [1]. In other data sources, there are about 11 million households (KRT) who have an income below Rp. 8 million who do not own a house and this number is equivalent to 15% of the total 71 million KRT. For the top 10 provinces with the percentage of households who do not own a house, Jakarta is at the top. The illustration is that two out of five households in the capital city do not yet own a house, the percentage is indeed large. Of the 3 million households in Jakarta, 41 percent do not own a house [2].

From this data, the government is trying to make Indonesian people have a place to live. One of them is through the making of Government Regulation Number 25 of 2020 concerning the Implementation of Public Housing Savings or commonly called PP 25 Tapera of 2020. Tapera is a term deposit by workers for housing financing, or it can also be returned after the time expires. The amount of this savings is 3% of the salary received by workers with a division of 0.5% from the employer or in this case the company and 2.5% from the workers themselves [3].

There have been many responses from the public regarding this regulation. Their responses varied, ranging from those who rejected this regulation, those who supported it, or those who were neutral. Twitter social media is one of the media for delivering public feedback regarding the implementation of this regulation. From their responses, the sentiments of the community regarding the implementation of this regulation can be seen. It makes a suitable research model for sentiment analysis, such as using Naïve Bayes, Support Vector Machine (SVM), KNN, and so on. From several sentiment analysis studies,

the algorithm that has a better level is the Support Vector Machine (SVM) [4].

SVM has better results as evidenced in several previous studies, including research on Sentiment Analysis on Tourism Objects in Central Java. The results of this study conclude that SVM has a higher accuracy result of approximately 10% than Naïve Bayes [5]. Other research on Comparative Analysis of Accuracy and Processing Time in SVM and KNN Algorithms shows SVM is slightly superior in accuracy compared to KNN [6]. Based on the research examples above, SVM has good accuracy results in the sentiment analysis process. However, the implementation of SVM in addition to requiring training data also requires labels and usually the data to be tested is data in large quantities so manual labeling will take a long time [7].

Based on the previous explanation, there needs to be a method to label, using opinion words or sentiment words, which words express a positive sentiment or negative sentiment. One of the labeling methods that can be used is the lexicon-based method. Lexicon-based is a method to be able to determine sentiment or polarity of opinion through several functions of opinion words in documents or sentences based on the lexicon dictionary [8]. Dictionaries are an important component in systems that use lexicon-based methods. The difficulty that occurs in the lexicon-based method lies in determining or updating the dictionary by humans. For that, it can be done by updating the dictionary automatically by combining lexicon-based with machine learning [9].

The above notion becomes the basis for the formation of a combination of SVM and lexicon-based. SVM approach can contribute to lexicon-based methods, and lexicon-based can help label datasets in SVM. This combination is carried out sequentially, using lexicon-based to determine the sentiment value, and the lexicon result data is used as labeling data for the Support Vector Machine. In other words, this combination makes lexicon-based a means to transfer learning to SVM in the hope that the combination of methods and algorithms can contribute to the sentiment labeling process to get good accuracy. [10].

2. Method

The research method used to conduct this research is to apply a combination of lexicon-based and support vector machines, where lexicon-based as a data labeling method with sentiment values and labeling results data is used by SVM. The dataset used comes from Twitter regarding the implementation of PP Tapera No. 25 the Year 2020.

Figure 1 shows the stages of the research carried out, including the stages of data collection that produces research datasets, then from the research datasets, preprocessing data is carried out, then followed by the data labeling process with lexicon-based, then the TF-IDF weighting process, processing with SVM, and the last is the evaluation process using the confusion matrix.

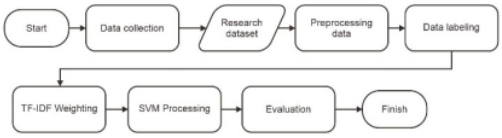

Figure 1. Research Steps

a. Data collection

The dataset used in this study is a dataset originating from Twitter response regarding PP Tapera No. 25 of 2020. The data collection method used is by using the help of Octoparse tools. After getting the Twitter URL about the dataset, then the URL is entered into octoparse. After entering the URL, next set up the data pagination process which has a function to extract data through different pages. The data extraction process will run and will be completed when the data is considered exhausted. Figure 2 is the dataset that was successfully collected in the data collection process. Later what will be used is only the Tweet column.

User	Date	Tweet
Kementerian PUPR	29-Mar-19	Masa jabatan Komisioner dan Deputi Komisioner ...
Paula	12-Jun-20	Peraturan Pemerintah yang ditandatangaini \n@j...
Radio PRFM 107,5 News Channel	4-Jun-20	Sampaikan opini dan komentar anda terkait PP T...
detikcom	17-Jun-20	Deputi Komisioner bidang Pemanfaatan Dana BP T...
Titik Terang	5-Jun-20	Kementerian PUPR mengatakan pemerintah ingin m...
...	...	...
Save Palestine	18-Jun-20	Selurhh Pekerja wajib jadi anggota TAPERA... D...
pi	5-Jun-20	Barusan cek lagi, ada batasnya max Rp 12 juta ...
Dr. Rizal Ramli	6-Jun-20	Mulai belajar bikin kebun hidrophonik utk 120 ...
???? ?I?? ??????	6-Jun-20	Klo punya hobi yg menghasilkan income diem² ba...
Ceu Umay	7-Jun-20	Bisa aja yg bikin ginian nih, tp menurut gw em...

Figure 2. Research Dataset

b. Data Preprocessing

The data that has been obtained, will previously go through a preprocessing process which has the aim of changing data that is still rough, unstructured, and has a lot of noise, into data that is ready to be processed [11] The preprocessing process applied consists of 5 stages which are depicted in Figure 3.

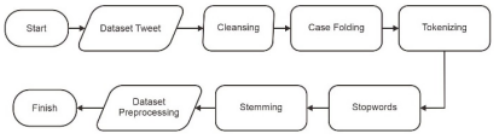

Figure 3. Preprocessing Steps

The first preprocessing stage is cleansing, the cleansing stage functions to remove punctuation characters, as well as characters that are not needed. Next is case folding to change the text of various tweet data consisting of lowercase and uppercase letters into all lowercase letters. Next, we have the tokenizing stage to change or break sentences into word-for-word parts. The stopwords stage has a function to remove words that are considered general and have no

meaning. Finally, the stemming stage is to change all the words into the basic form of the word, and the dataset is considered ready to be used for the research process.

c. Data Labeling

After performing the data preprocessing process, the next step is the data labeling process. Data labeling is needed to transfer learning to SVM, and since manual data labeling takes a long time, a lexicon-based method is needed. Lexicon-based labeling is a method to determine sentiment or polarity of opinion through several functions of opinion words in documents or sentences [8]. Lexicon-based labeling is used to give positive or negative weight to each word that appears based on the lexicon or a compiled dictionary that stores words with positive and negative sentiments [12].

The lexicon labeling process requires a dictionary as a reference to calculate the polarity of opinion or sentiment. The dictionaries used in this study are compiled based on small dictionaries provided by evanmertua34's github repository with links https://github.com/evanmartua34/Twitter-COVID19-Indonesia-Sentimen-Analysis---Lexicon-Based. These small dictionaries consist of a small dictionary of positive and negative lexicon from Inset (Indonesia Sentimen Lexicon) belonging to Fajri Koto, and Gemala Y. Rahmaningtyas), then there is a small dictionary of lexicon sentiment words, then there is also a small dictionary of lexicon swear words which contains the words - swear words, as well as other small dictionaries. Later all the small dictionaries will be combined into a main lexicon dictionary for the data labeling process. The stages of making the main lexicon dictionary are described in Figure 4.

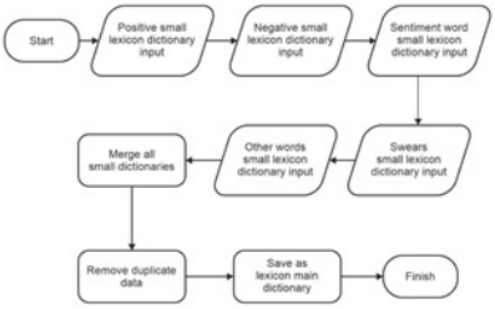

Figure 4. Making a Lexicon Based Dictionary

The next step is the process of labeling the dataset with the created lexicon dictionary. The labeling process is done by checking the sentiment words or opinion words contained in each tweet data, then the weight is calculated based on the lexicon dictionary. The results of the calculation of the value/weight of sentiment or opinion words in each tweet data are accumulated to see the results of labeling the tweet data. If the accumulated weight value in each tweet data is more than 0 (> 0), then the tweet is labeled positive. If the accumulated result is less than 0 (<0), the tweet is labeled negative. Meanwhile, a tweet is consedered neutral, if the accumulated results are equal to 0. More or less, the equation is as follows [13]:

$$\text{if} \begin{cases} \sum \text{k Score (k)} > 0 \text{ then positive} \\ \sum \text{k Score (k)} < 0 \text{ then negative} \\ \sum \text{k Score (k)} = 0 \text{ then neutral} \end{cases} \quad (1)$$

The data labeling process with the lexicon dictionary is shown in Figure 5.

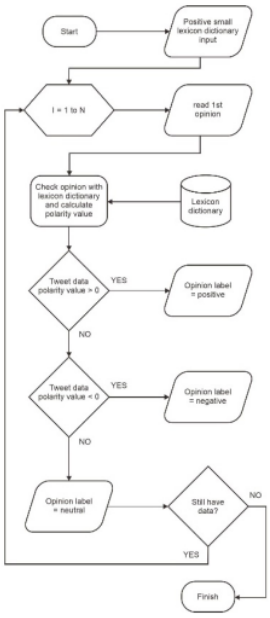

Figure 5. Data Labeling Process

d. Term Weighting

After carrying out the data labeling process, the next step is term weighting process with TF-IDF. Because basically the machine can only process numbers, then each word in the document will be given a weight or frequency value, and then the result of this weighting process is a vector value, later the vector value will be entered into a vector space for later use for the testing process algorithm after this [14].

The weighting process with TF-IDF is shown in Figure 6. The dataset that has gone through the data labeling process will then be divided into 80% composition for training, and 20% for testing. The next is to calculate the TF or Term-Frequency value based on the number of occurrences of each word in the document. Then, calculate the DF or Document-Frequency, which is to calculate the value of the number of documents that have terms, then calculate the value of N to count the entire number of existing documents. Next, calculate the Inverse Document Frequency from DF and N. After the TF and IDF values are obtained, then we do the calculation [15]. The results of this calculation will produce a word weight. The TF-IDF results from the data train will become a learning model using SVM and the learning model is tested by the testing data.

The TF-IDF calculation equation is shown through the equation below:

$$TF\begin{cases}1+\log_{10}(f_{t,d}), & f_{t,d} > 0\\ 0, & f_{t,d} = 0\end{cases} \quad (2)$$

$$IDF = \log_{10}\frac{N}{DF_t} \quad (3)$$

$$W_{t,d} = TF \cdot IDF \quad (4)$$

with:
TF = Word weight of each document
$f_{t,d}$ = Number of occurrences of term in document
IDF = Inverse weights in DF documents
DF = Number of documents containing Term
$W_{t,d}$ = TF-IDF Weighting

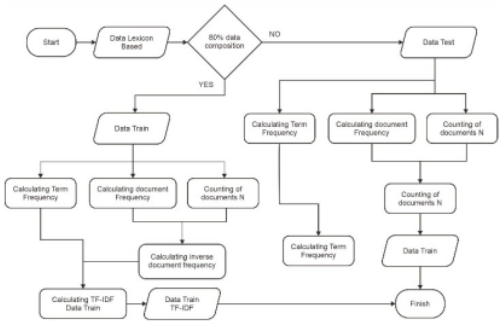

Figure 6. TF-IDF Weighting Steps

e. SVM Processing

After going through the weighting process, the next is to do sentiment analysis with SVM. SVM or Support Vector Machine is one of the algorithms with supervised learning category for data classification needs [16]. The working concept of SVM is to find the best hyperplane that separates two classes in the input space. Maximum Marginal Hyperplane calculation is used to obtain the best hyperplane in separating two classes [17].

The SVM illustration can be seen in the equation as shown in Figure 7.

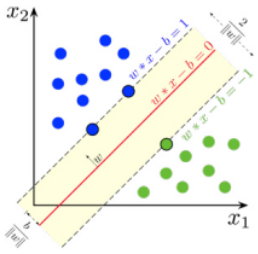

Figure 7. Hyperplane Support Vector Machine [26]

In obtaining the hyperplane, the following equation can be used:

$$(w \cdot x_i + b) = 0 \quad (5)$$

x_i is *tuple* and is class label with i = 1... . N, $x_i \in Rd$ and $y_i \in \{-1,1\}$. The equation is as follows [18]:

$$f(x_i)\begin{cases}(w \cdot x_i + b) \le 1, y_i = -1\\ (w \cdot x_i + b) \ge 1, y_i = 1\end{cases} \quad (6)$$

Although it is explained that the hyperplane separates two classes in the input space, SVM has been developed to be able to overcome the problem of more than two classes or non-linear classifiers by using the kernel trick concept [19]. By using the kernel trick, it can make it easier to solve non-linear problems by entering data in a high-dimension space [20]. The equations of each SVM kernel function can be seen in Table 1.

Table 1. Kernel Equation

Kernels	Equality
Linear	$K(X_i, X_j) = X_i^T X_j$
RBF	$K(X_i, X_j) = \exp\left(-\gamma\lvert X_i - X_j\rvert^2\right), \gamma > 0$
Polynomial	$K(X_i, X_j) = (\gamma \cdot X_i^T X_j + r)^d, \gamma > 0$
Sigmoid	$K(X_i, X_j) = \tanh(\gamma \cdot X_i^T X_j + r)$

Each kernel function in SVM has certain parameters that are used for the testing process in this research. The parameters used are described in Table 2. These parameters are general parameters that are often used. The C value is the most common for all SVM kernels. The gamma parameter (γ) is used to determine the level of proximity between two points, making it easier to find a hyperplane separator that is consistent with the data. The degree (d) parameter has a function to help map data from the input space to a higher dimension space in the feature space, so that in the new dimension a consistent hyperplane can be found. The parameter coef0 serves as an independent value [21].

Table 2. SVM Kernel Parameters

Kernels	Parameter			
	C	*Gamma ()*	*Degree (d)*	*Coef0*
Linear	√	×	×	×
RBF	√	√	×	×
Polynomial	√	√	√	√
Sigmoid	√	√	×	√

The processing stage with the Support Vector Machine (SVM) in this study is described in Figure 8. The data that has passed the weighting process with TF-IDF, then entering the process of determining the parameters used for each kernel function. Then carry out the process

of testing the data train with kernel functions that already have certain parameters. Testing the data train produces a learning model that is used by the test data. The result of the test data is a model accuracy result of each SVM kernel function.

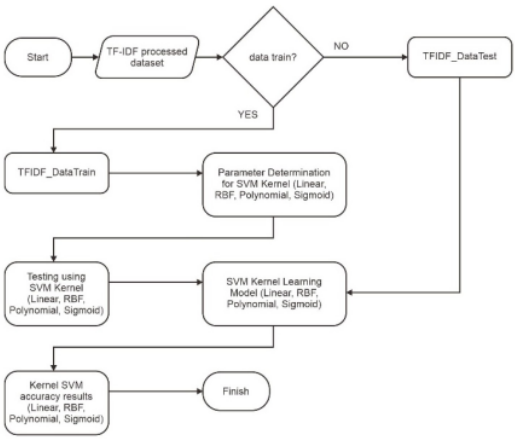

Figure 8. SVM Processing Steps

f. Evaluation

The evaluation process is carried out with a confusion matrix to test how well the classification algorithm performance model is built or used [22]. The confusion matrix provides a comparative information between the actual classification results or the actual results and predictions [23]. In general, the confusion matrix has a 2X2 structure for binary classification, which is shown in Table 3.

Table 3. Confusion Matrix 2X2

Actual	Prediction	
	Positive	***Negative***
Positive	TP (True Positive)	FP (False Positive)
Negative	FN (False Negative)	TN (True Negative)

For non-binary classification needs, it is adjusted to the number of labels or classes used. Because, in this study, the classification of 3 labels or 3 classes consisting of negative, neutral, and positive was applied, the confusion matrix structure is as shown in Table 4.

Table 4. Confusion Matrix 3X3

Actual	Prediction		
	Negative	**Neutral**	**Positive**
Negative	TNt (True Negative)	FNe (False Neutral)	FP (False Positive)
Neutral	FNt (False Negative)	TNe (True Neutral)	FP (False Positive)
Positive	FNt (False Negative)	FNe (False Neutral)	TP (True Positive)

The confusion matrix is also the basis for calculating accuracy, precision, recall, and f1-score values from the test. The accuracy value shows the correct prediction results with actual conditions. Precision shows the accuracy or accuracy of the test results. Recall shows the value to measure the proportion of the results of the correct value identified. F1-Score is the result of precision and recall [24]. The formula for determining the values above is shown below:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (7)$$

$$Precision = \frac{TP}{TP + FP} \quad (8)$$

$$Recall = \frac{TP}{TP + FN} \quad (9)$$

$$F1 - Score = 2\frac{precision \times recall}{precision + recall} \quad (10)$$

Information:

TP (True Positive) = Indicates a positive result detected correctly

TN (True Negative) = Indicates a correctly detected negative result

FP (False Positive) = Shows negative results detected positive

FN (False Negative) = Shows positive results detected negative

3. Result

The results of the study contain a description of each stage or process carried out in this study.

a. Data Preprocessing

The tweet dataset used in this study is unstructured, coarse, and still has a lot of characters that can cause noise. Thus, it is necessary to carry out a data preprocessing process to make the data ready to be processed. The preprocessing stage carried out is the cleansing process to remove punctuation characters, as well as characters that are not needed. The results of the cleansing process can be seen in Table 5.

Table 5. Cleansing Process Results

Before	After
Masa jabatan Komisioner dan Deputi Komisioner Badan Pengelola Tabungan Perumahan Rakyat adalah selama lima tahun sejak ditetapkannya Keputusan Presiden. #BPTapera #Tapera #SejutaRumah #InfrastrukturUntukIndonesiaMaju #PUPRSigapMembangunNegeri	Masa jabatan Komisioner dan Deputi Komisioner Badan Pengelola Tabungan Perumahan Rakyat adalah selama lima tahun sejak ditetapkannya Keputusan Presiden BPTapera Tapera SejutaRumah InfrastrukturUntukIndonesiaMaju PUPRSigapMembangunNegeri

The case folding process stage is performed to change all characters consisting of uppercase and lowercase letters

into whole lowercase letters. The results of case folding can be seen in Table 6.

Table 6. Case Folding Process Results

Before	After
Masa jabatan Komisioner dan Deputi Komisioner Badan Pengelola Tabungan Perumahan Rakyat adalah selama lima tahun sejak ditetapkannya Keputusan Presiden BPTapera Tapera SejutaRumah InfrastrukturUntukIndonesiaMaju PUPRSigapMembangunNegeri	masa jabatan komisioner dan deputi komisioner badan pengelola tabungan perumahan rakyat adalah selama lima tahun sejak ditetapkannya keputusan presiden bptapera tapera sejutarumah infrastrukturuntukindonesiamaju puprsigapmembangunnegeri

Then, the tokenizing process is carried out to change or break a sentence into word-for-word parts. Table 7 gives the results of the tokenizing process.

Table 7. Tokenizing Process Results

Before	After
masa jabatan komisioner dan deputi komisioner badan pengelola tabungan perumahan rakyat adalah selama lima tahun sejak ditetapkannya keputusan presiden bptapera tapera sejutarumah infrastrukturuntukindonesiamaju puprsigapmembangunnegeri	['masa', 'jabatan', 'komisioner', 'dan', 'deputi', 'komisioner', 'badan', 'pengelola', 'tabungan', 'perumahan', 'rakyat', 'adalah', 'selama', 'lima', 'tahun', 'sejak', 'ditetapkannya', 'keputusan', 'presiden', 'bptapera', 'tapera', 'sejutarumah', 'infrastrukturuntukindonesiamaju', 'puprsigapmembangunnegeri']

The next process is the stopwords process. This process serves to remove words that are considered general and have no meaning. The results of stopwords are described in Table 8.

Table 8. Stopwords Process Results

Before	After
['masa', 'jabatan', 'komisioner', 'dan', 'deputi', 'komisioner', 'badan', 'pengelola', 'tabungan', 'perumahan', 'rakyat', 'adalah', 'selama', 'lima', 'tahun', 'sejak', 'ditetapkannya', 'keputusan', 'presiden', 'bptapera', 'tapera', 'sejutarumah', 'infrastrukturuntukindonesiamaju', 'puprsigapmembangunnegeri']	['jabatan', 'komisioner', 'deputi', 'komisioner', 'badan', 'pengelola', 'tabungan', 'perumahan', 'rakyat', 'ditetapkannya', 'keputusan', 'presiden', 'bptapera', 'tapera', 'sejutarumah', 'infrastrukturuntukindonesiamaju', 'puprsigapmembangunnegeri']

The final stage is the stemming process to change all the words into the basic form of the word. Or in other cases, stemming will remove the affixes in each word. The results of stemming are shown in Table 9.

Table 9. Results of the Stemming Process

Before	After
['jabatan', 'komisioner', 'deputi', 'komisioner', 'badan', 'pengelola', 'tabungan', 'perumahan', 'rakyat', 'ditetapkannya', 'keputusan', 'presiden', 'bptapera', 'tapera', 'sejutarumah', 'infrastrukturuntukindonesiamaju', 'puprsigapmembangunnegeri']	jabat komisioner deputi komisioner badan kelola tabung rumah rakyat tetap putus presiden bptapera tapera sejutarumah infrastrukturuntukindonesiamaju puprsigapmembangunnegeri

After performing various stages of preprocessing starting from cleaning, case folding, tokenizing, stopwords, and stemming, the preprocessing dataset was obtained. Figure 9 is a dataset resulting from the data preprocessing process. From here, the dataset is ready to be used for the research process.

b. Data Labeling

The process of labeling or labeling or classing tweets is done using the lexicon-based method. The lexicon-based method in this study was carried out using the main lexicon dictionary which had been compiled from small lexicon dictionaries from evanmertua34's github repository, with the link *https://github.com/evanmartua34/Twitter-COVID19-Indonesia-Sentimen- Analysis---Lexicon-Based* . The small dictionary consists of a positive inset dictionary, a negative inset dictionary, a sentiment word dictionary, a swear words dictionary, and several other small dictionaries.

Table 10. Positive Inset Dictionary Example

Positive Inset Dictionary	
Word	***Weight***
Thank you	5
Please	2
Sorry	2

The positive inset dictionary consists of words with a weight above 0. Table 10 is an example of some words from the positive inset dictionary. It can be seen that the word 'thank you' has a weight of 5, or the word with the highest weight. Then there is the word 'please' with a weight of 2, and there is the word 'sorry' with a weight of 2.

Table 11. Negative Inset Dictionary Example

Negative Inset Dictionary	
Word	***Weight***
No	-3
Betray	-4
Do not want	-4

The negative inset dictionary consists of words with weights below 0. Table 11 is an example of some words from the negative inset dictionary. The word 'no' has a weight of -3. Then, there is the word 'betray' has a weight of -4, and the word 'don't want to' also has a weight of -4.

Table 12. Sentiment Word Dictionary Example

Word Sentiment Dictionary	
Word	***Weight***
Wonderful	3
Anarchy	-2
Graceful	-4

Sentiment word dictionary consists of sentiment words with a weight range of -5 to 5. Table 12 is an example of some words from the sentiment word dictionary. The word 'wonderful' has a weight of 3. Then, there are the words 'anarchy' and 'graceful' with a weight of -2 and -4 respectively.

Table 13. Swear Words Dictionary Example

Swear Words Dictionary	
Word	***Weight***
Stupid	-5
Curse	-5
Crazy	-5

Swear words dictionary is a small dictionary that contains swear words, so all the words in this dictionary have a weight of -5. It can be seen for example some words in Table 13, that the words 'stupid', 'curse', and 'crazy' have a weight of -5.

The existing small lexicon dictionaries are combined into one and then duplicated to become the main lexicon dictionary which consists of ±10200 words with various weights. The dataset labeling process is carried out by going through the process of calculating sentiment words by calculating them into the lexicon. Then create a data frame to store the previous bag of words and sentiment word calculations.

Table 14. Result Dataframe Dataset to Lexicon

	shake	body	tube	...	key	Spice	Sentiment
0	1	1	1	...	0	0	9
1	0	1	0	...	0	0	21
2	0	0	0	...	0	0	1
3	0	0	0	...	0	0	9
...	...	...	...	...	...	...	...
516	0	0	0	...	1	1	8
517	0	0	0	...	0	0	-7
518	0	0	0	...	0	0	2

519 rows X 634 columns

Table 14 is a dataframe structure created to accommodate the bag of words and the calculation of sentiment words in the dataset to the lexicon. The description "519 rows" indicates the number of tweet datasets that are labeled. Then, 634 is the number of sentiment words in the dataset that are calculated into the lexicon. The sentiment column shows the weight of each tweet data. It can be seen that the 3rd row tweet data has a weight of 9, then the 516th row tweet data has a weight of 8, and the 517th row tweet data has a weight of -7.

The results of all tweet datasets are given their respective weights, then converted into 3 main label forms. Tweets with a weight above 0, will be changed to label 1 or positive. Tweets with a weight below 0, are changed to the label -1 or negative. And, tweets with a weight equal to 0, are changed to 0 or neutral labels.

Table 15. Tweet Labels

Tweet Weight	Tweet Labels	Label Description	Total
Above 0 (> 0)	1	Positif	370
Under 0 (<0)	-1	Negatif	127
Equal 0 (=0)	0	Netral	22
Total Dataset			519

Table 15 shows the final results of the lexicon-based data labeling that has been used. It can be seen that, out of 519 datasets, 370 data have a positive label or 1. 127 data have a negative label or -1. And 22 data has a neutral label or 0. With this, the data labeling process has been successful and the labeling data can be used by SVM.

c. Term Weighting

The weighting is done after the data already has a label. The weighting process with TF-IDF serves to get a vector value from the dataset, later the vector value will be entered into a vector space for later use for the algorithm testing process.

The dataset that has gone through the labeling process is then divided into train data and test data with a composition of 80% versus 20%. Then, the train data and test data are carried out by the TF-IDF weighting process. The results of the TF-IDF train data and test data are described in Table 16 and Table 17.

In Table 16, the tweet column contains as many rows as the number of data in the data train. In other words, shows the index of each data. Then the term column contains the index of the feature word generated from the bag of words. The weight column contains the weight of each generated term. So, how to read from Table 16 is the data at index 0 which contains the 1811th term with the term weight 0.281070643.

In Table 17, the tweet column contains as many rows as the index of each data in the test data. Then the term column contains the index of the feature word generated

from the bag of words. The weight column contains the weight of each generated term. So, how to read from Table 17 is the data at index 0 which contains the 1777 term with a term weight of 0.480879633.

The TF-IDF results from the data train in the form of vector space will become a learning model and will be tested by data testing to find out how well the learning model is.

Table 16. TF-IDF Data Train Results

Tweet	Term	Weight
0	1811	0.281070643
0	1763	0.232067791
0	1728	0.571941689
0	1713	0.232067791
0	1650	0.040252079
0	1635	0.173988977
⋮	⋮	⋮
414	266	0.284576677
414	233	0.167824591
414	175	0.294715495
414	119	0.275891625
414	52	0.356111407

Table 17. TF-IDF Data Test Results

Tweet	Term	Weight
0	1777	0.480879633
0	1650	0.052862833
0	1429	0.192178494
0	1156	0.290528875
0	1108	0.32484981
0	1008	0.279480002
103	642	0.119453367
103	482	0.240375508
103	235	0.319535955
103	155	0.294302118
103	121	0.356111407

d. SVM Processing

Processing using the Support Vector Machine (SVM) focuses on the use of four kernel functions commonly used in SVM, namely the Linear Kernel, Radial Basis Function (RBF) Kernel, Polynomial Kernel, and Sigmoid Kernel.

The parameters used in the testing process are adjusted to the needs of each kernel. These parameters will determine the results of the tests in each kernel. Therefore, the hyperparameter tuning process is carried out using the grid search method to determine the optimal hyperparameter. After the input hyperparameter values are initialized, the grid search will train several prediction models that are formed through each combination of hyperparameter values. The parameter pair that produces the best accuracy obtained is the optimal parameter [25]..

Parameter testing on the kernel is done by assigning a value to each parameter used, namely degree, complexity or C, gamma, coef0, and maximum iteration or max iter. Table 18 describes the input values for the hyperparameter tuning process in each kernel.

Table 18. Input Hyperparameter Value

Kernels	Parameter	Value
Linear	C	0.001, 0.01, 0.1, 1, 1.5, 2, 2.5, 10, 20, 100
	Max Iteration	0.001, 0.01, 0.1, 1, 1.5, 2, 2.5, 10, 20, 100
RBF	C	0.001, 0.01, 0.1, 1, 1.5, 2, 2.5, 10, 20, 100
	Gamma	0.01, 0.1, 0.5, 1, 10, 20, 40, 50, 100, 1000
Polynomial	C	0.001, 0.01, 0.1, 1, 1.5, 2, 2.5, 10, 20, 100
	Degree	0.01, 0.1, 0.5, 1, 10, 20, 40, 50, 100, 1000
	Gamma	0.01, 0.1, 0.5, 1, 10, 20, 40, 50, 100, 1000
	Coef0	0.001, 0.01, 0.0, 1, 10
Sigmoid	C	0.001, 0.01, 0.1, 1, 1.5, 2, 2.5, 10, 20, 100
	Gamma	0.01, 0.1, 0.5, 1, 10, 20, 40, 50, 100, 1000
	Coef0	0.001, 0.01, 0.0, 1, 10

After initializing the input values for each hyperparameter, then testing it using a grid search to get the most optimal combination of parameters. Table 19 is the result of the combination of the best hyperparameter values in each kernel by testing it against the train data.

Table 19. Best Hyperparameter Value for Each Kernel

	Kernels	Parameter				
		C	*Max Iter*	*Gamma*	*Degree*	*Coef0*
1	Linear	2.5	100	×	×	×
2	RBF	100	×	0.1	×	×
3	Polynomial	100	×	0.1	10	1
4	Sigmoid	20	×	1	×	1

After obtaining the best combination of hyperparameter values for each kernel, testing is carried out to prove how suitable or how well the kernel tested in research is. Table 20 shows the results of the tests that have been carried out. It can be seen that the rbf kernel has the highest accuracy value with a value of 81.73%. Then, for the precision value, the sigmoid kernel has the highest value with 78.68%. The rbf kernel also has the highest recall value with 81.73%. Meanwhile, for the results

of the highest f1-score values are owned by two kernels, namely the rbf kernel and the sigmoid kernel with a value of 79.60%.

Table 20. Test Results of Each Kernel Function

	Kernels	Parameter			
		Accuracy	*Precision*	*Recall*	*F1-Score*
1	Linear	80.77	77.33	80.77	78.93
2	RBF	81.73	78.27	81.73	79.60
3	Polynomial	78.85	75.28	78.85	76.22
4	Sigmoid	80.77	78.68	80.77	79.60

e. Evaluation

The evaluation process is carried out using a confusion matrix. The confusion matrix has a function to test how well the classification algorithm's performance model is built or used. The confusion matrix also provides comparative information between the actual classification results or the actual results and predictions. The confusion matrix structure used is based on Table 4.

Table 21. Linear Kernel Confusion Matrix

Actual	Prediction		
	Negative	*Neutral*	*Positve*
Negative	18	0	9
Neutral	0	0	4
Positive	7	0	66

In Table 21, the linear kernel tests 27 tweets with negative labels, 4 tweets with neutral labels, and 73 tweets with positive labels. It is found that the linear kernel detects classes in tweet data labeled negative as much as 18 for True Negative, and as many as 9 for False Positive. Then tweet data labeled neutral can only be detected for False Positive as much as 4 data. Then, 66 positive labeled tweets were detected for True Positive and 7 for False Negative.

Table 22. Confusion Matrix Kernel RBF

Actual	Prediction		
	Negative	*Neutral*	*Positve*
Negative	17	0	10
Neutral	0	0	4
Positive	5	0	68

In Table 22, kernel rbf, tweet data labeled negative was detected as many as 17 for True Negative, and as many as 10 for False Positive. Neutral labeled data detected as many as 4 for False Positive. Positively labeled tweet data was detected for 68 True Positives and 5 False Negatives.

In Table 23, kernel polynomial, negative labeled tweet data were detected as many as 14 for True Negative, and as many as 13 for False Positive. Neutral labeled data detected as many as 4 for False Positive. Positively labeled tweet data was detected for True Positive as many as 68 and False Negative as many as 5.

Table 23. Confusion Matrix Kernel Polynomial

Actual	Prediction		
	Negative	*Neutral*	*Positve*
Negative	14	0	13
Neutral	0	0	4
Positive	5	0	68

Table 24. Confusion Matrix Sigmoid Kernel

Actual	Prediction		
	Negative	*Netral*	*Negative*
Negative	17	2	8
Neutral	1	0	3
Positive	6	0	67

In Table 24, sigmoid kernel, 17 negative labeled tweet data were detected for True Negative, 2 tweet data for False Neutral, and 8 for False Positive Neutral labeled data detected 3 for False Positive and 1 for False Negative. Positively labeled tweet data was detected for 67 True Positives and 6 False Negatives.

From the evaluation results using the confusion matrix, it can be seen that SVM and all SVM kernel functions can predict well.

After getting the rbf kernel function as the kernel with the best accuracy results, then displaying the sentiment classification with SVM on the test data from the dataset on Government Regulation No. 25 of 2020. The sentiment classification is carried out using SVM with the rbf kernel as the best kernel. The classification results are shown in Table 4.25.

Table 25. Example of Sentiment Classification Results

Final Tweet	SVM Results	Lexicon Results
Dengar tapera tabung rumah rakyat serta tapera klinikhukum	tive	Positive
pns hak rumah tapera pegawai perintah penjuru indonesia serta program tabung rumah rakyat penuh syarat manfaat milik huni lokadata id	Positive	Positive
wasekjen demokrat perintah beban rakyat rumah wasekjen beban iur rakyat wajib negara penuh uang masyarakat utk dana tapera	Positive	Negative
gaji sunat negara iur tapera gaji pegawai potong bayar pph bpjs sehat bpjs ketenagakerjaan yg bagi jht jamin pensiun jp	Positive	Neutral

From the classification results, it shows that there are still differences in the results of the SVM and the results of the lexicon. Because, there is tweet data that should have a

negative value from the lexicon label, but the SVM results show it has a positive value.

Table 26. Difference between Predictive and Actual Results

Label	Actual Results	Predictive Results
Negative	73	82
Neutral	27	22
Positive	4	0

4. Discussion

a. Discussion of Research Results

The implementation of 5 preprocessing stages, namely the cleansing, case folding, tokenizing, stopwords, and stemming stages was successful for the need to change data that is still unstructured, still rough, and still has a lot of noise into ready-to-process data.

Then, the application of the lexicon-based method for the labeling needs of tweet data shows good results. Lexicon-based managed to label the tweet dataset with a total of 519 tweets. Lexicon-based labels tweets based on the accumulated weight of sentiment or opinion words in each tweet. The distribution of labeling values is shown in Figure 10. The largest labeling value is 30 and the smallest labeling value is -25. The results of lexicon-based labeling are used as a method to transfer learning to SVM through labeling tweet data.

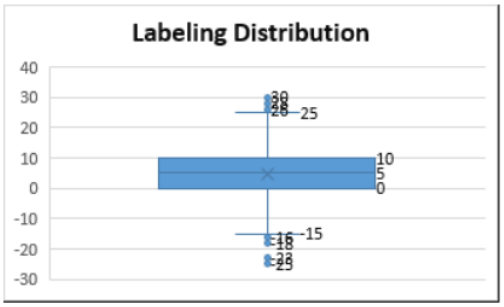

Figure 10. Distribution of Data Labeling Results

The dataset that already has a label is then weighted using TF-IDF. In the TF-IDF weighting process, the data is divided into train data and test data. The result of the TF-IDF weighting of the data train is the vector value and this vector value will be entered into the vector space to be processed using SVM. The result of vector space processing from the data train is a learning model. This learning model will be tested how well by test data. TF-IDF results from train data and test data are described in Table 16 and Table 17.

Processing using SVM is carried out based on four kernel functions, namely linear kernel, rbf kernel, polynomial kernel, and sigmoid kernel. The parameters used in each kernel are obtained through the hyperparameter tuning process using the grid search method. This method serves to get the best combination of parameters for testing in each kernel function. The best combination of parameters that have been obtained for each kernel function is in Table 19. The test results for each SVM kernel function can be seen in the graph below.

From the graph of the kernel test results, it can be seen that the rbf kernel has the best test performance when viewed from the accuracy and recall values. When viewed based on the precision value, the sigmoid kernel has the best test results. Thus, the rbf kernel is the kernel with the best performance, followed by the sigmoid kernel, then the linear kernel, and the polynomial kernel is the kernel with the lowest performance compared to other kernels in this test.

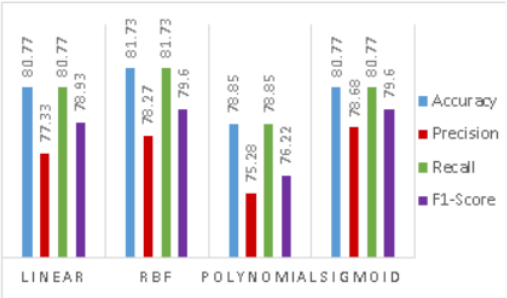

Figure 11. SVM Kernel Test Result Chart

In the evaluation, it can be seen how well the classification algorithm performance model is built based on the comparison of actual and predicted results with the confusion matrix.

Based on the results of the confusion matrix in each kernel in Tables 21 to 24, it can be seen that each kernel has a good performance for predicting tweet data with negative and positive labels. If described based on the example in Table 22, the rbf kernel succeeded in predicting 17 tweet data labeled negative correctly or True Negative, while the rbf kernel failed to predict 10 data labeled positive which should be categorized as negative, or can be called False Negative. Then, the rbf kernel, successfully predicting 68 tweet data labeled positive correctly or True Positive, also failed to predict 5 tweet data labeled negative which should be categorized on a positive label or can be called False Negative.

Although, the confusion matrix results of all kernel function tests have good performance in predicting data with negative and positive labels, however, all kernel test results are shown fail to correctly predict tweet data with neutral labels. It can be seen in the results of the confusion matrix in Tables 21 to 24, all kernels fail to predict 4 positive-labeled tweet data that should be categorized on a neutral label or can be called False Positive. In other words, all tests of the four kernel functions failed to correctly predict the neutral-labeled tweet data.

The thing that causes a failure in testing the kernel function to predict tweet data with neutral labels, is due to

an unbalanced dataset. Unbalanced dataset is a condition where there is a significant difference in the distribution of class or data labels which results in non-ideal conditions in the classification. The results of the class distribution or data labels are shown in Figure 12. Unbalanced datasets cause the kernel test results on the confusion matrix to fail or fail. Because, unbalanced dataset causes the algorithm model to give prediction results into the dominant data class or label, or in this case the data is positively labeled. This is evidenced in the results of the confusion matrix in each kernel function, where all neutral tweet data is predicted to be positive tweet data.

4.2 Comparison of Accuracy Results

As a result of the proof based on the background of using the SVM algorithm that has been described previously, a test was carried out to compare the accuracy results based on the Naïve Bayes, KNN, and SVM algorithms. SVM results are taken from all kernel tests, namely linear, rbf, polynomial, and sigmoid kernels. For input hyperparameter values in the Naïve Bayes and KNN algorithms are described in Table 25.

Table 27. Input Hyperparameters Nave Bayes and KNN Values

Algorithm	Parameter	Value
Multinomial Naïve Bayes	Alpha	1, 0.7, 0.4, 0.2, 0.1, 0.09, 0.08, 0.07, 0.06, 0.03, 0.01
KNN	n_neighbours	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30

After determining the input values for the hyperparameters for the Naïve Bayes and KNN algorithms, testing is carried out to obtain the accuracy values. Then the accuracy value is compared to determine and prove that the accuracy of SVM is the best compared to Naïve Bayes and KNN. Comparison of accuracy results is shown in the graph below.

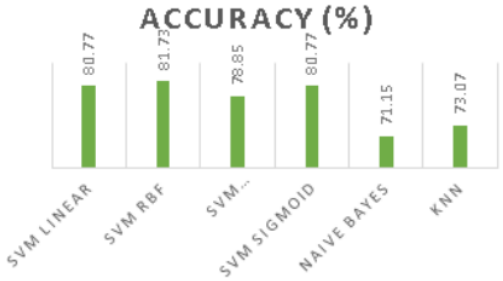

Figure 12. Results of Comparison of Accuracy of SVM, Naïve Bayes, and KNN

From the accuracy results, it can be seen that SVM is proven to have the best accuracy results. In fact, all SVM kernels have accuracy values above Naïve Bayes and KNN.

5. Conclusion

The analysis of the research results suggests that the application of the lexicon-based method for the data labeling process is successful. Lexicon-based managed to label 519 tweet data. The tweet data labels consist of 127 negative-labeled, 22 neutral, and 370 positive-labeled tweet data. The combination of lexicon-based and support vector machine algorithms also runs well. This combination goes sequentially, using lexicon-based to determine the sentiment value and the lexicon result data as labeling data for the support vector machine. This combination makes lexicon-based to transfer learning to SVM with the hope that it contributes to the sentiment labeling process to get good accuracy.

SVM was successfully implemented and supported by several methods such as preprocessing, data labeling, weighting with TF-IDF, and hyperparameter tuning with grid search. The highest accuracy results from the application of SVM with RBF kernel with an accuracy value of 81.73%. The sigmoid kernel has the highest precision at a level of 78.68%. SVM with RBF kernel attains the highest recall at 81.73%. Meanwhile, the RBF kernel and the sigmoid kernel achieve the highest F1 score at 79.6%.

Table 25 and Table 26 show discrepancies in labels by SVM and lexicon-based. SVM indicates a positive sentiment value for the negative lexicon label. However, the accuracy still shows a pretty good number at 81.73%.

Each kernel test has a pretty good result. The evaluation stage using the confusion matrix shows that each kernel function test can predict well the tweet data with negative and positive labels. However, all kernel tests show predictive failure for tweet data labeled neutral. The neutral tweet data is predicted positive. The error may happen due to the condition of unbalanced datasets. There are differences in the class distribution, or the data labels are significant. This condition makes the algorithm predict data into the dominant class.

References

[1] Kumparan, "Apa itu Tapera? Fakta Penting Tabungan Perumahan Rakyat, Akankah Berbuah Rumah?," kumparan, 11 06 2020. [Online]. Available: https://kumparan.com/kumparansains/apa-itu-tapera-fakta-penting-tabungan-perumahan. [Accessed 10 05 2021].

[2] Lokadata, "52,4% Kepala Rumah Tangga Milenial Belum Punya Rumah, Tapera Jadi Harapan," Rumah123.com, 29 07 2020. [Online]. Available: https://artikel.rumah123.com/52-4-kepala-rumah-tangga-milenial-belum-punya-rumah-tapera-jadi-harapan-61264. [Accessed 10 05 2021].

[3] BPK RI, "JDIH BPK RI DATABASE PERATURAN," 20 05 2020. [Online]. Available: https://peraturan.bpk.go.id/Home/Details/137950/pp-no-25-tahun-2020. [Accessed 1106 2021].

[4] F. S. Pamungkas and I. Kharisudin, "Analisis Sentimen dengan SVM, NAIVE BAYES dan KNN untuk Studi Tanggapan Masyarakat Indonesia Terhadap Pandemi Covid-19 pada Media Sosial Twitter," PRISMA, Prosiding Seminar Nasional Matematika, vol. 4, p. 633, 2020.

[5] N. D. S. E. &. S. I. Susanti, "Uji Perbandingan Akurasi Analisis Sentimen Pariwisata Menggunakan Algoritma Support Vector Machine dan Naive Bayes. 3(2), 26-33," Nusantara of Engineering, vol. 2016.

[6] M. R. A. Nasution and M. Hayaty, "Perbandingan Akurasi dan Waktu Proses Algoritma K-NN dan SVM dalam Analisis Sentimen Twitter," JURNAL INFORMATIKA, vol. p. 233, 2016.

[7] I. &. M. H. Syafei, "Analisis Kinerja Kombinasi Metode Berbasis Lexicon dan Metode Berbasis Learning pada Analisis Sentimen Twitter.," Universitas Indonesia, Depok, 2014.

[8] A. I. &. A. S. Kurniawan, "Analisis Sentimen Opini Film Menggunakan Metode Naïve Bayes dan Lexicon Based Features," Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, vol. 3, p. 8338 2019.

[9] A. Nurfalah, Adiwijaya and A. A. Suryani, "ANALISIS SENTIMEN BERBAHASA INDONESIA DENGAN PENDEKATAN LEXICON-BASED PADA MEDIA SOSIAL," JURNAL MASYARAKAT INFORMATIKA INDONESIA, vol. 2, p. 8, 2017.

[10] D. W. Seno and A. Wibowo, "Analisis Sentimen Data Twitter Tentang Pasangan Capres-Cawapres Pemilu 2019 Berbasis Metode Lexicon Dan Support Vector Machine," JURNAL ILMIAH FIFO, vol. VI, pp. 144-154, 2019.

[11] F. S. Jumeilah, "Penerapan Support Vector Machine (SVM) untuk Pengkategorian Penelitian," Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), vol. 1, p. 2020, 2017.

[12] D. W. P. Lestari, R. S. Perdana and P. P. Adikara, "Klasifikasi Video Clickbait pada YouTube Berdasarkan Analisis Sentimen Komentar Menggunakan Learning Vector Quantization (LVQ) dan Lexicon Based Features," Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, vol. 3, p. 1186, 2019.

[13] R. Arief and K. Imanuel, "ANALISIS SENTIMEN TOPIK VIRAL DESA PENARI PADA MEDIA SOSIAL TWITTER DENGAN METODE LEXICON BASED," Jurnal Ilmiah MATRIK, vol. 21, p. 245, 2019.

[14] S. H. Kusumahadi, H. Junaedi and J. Santoso, "Klasifikasi Helpdesk Menggunakan Metode Support Vector Machine," Jurnal Informatika: Jurnal Pengembangan IT (JPIT), vol. 4, p. 55, 2019.

[15] R. Melita, V. Amrizal, H. B. Suseno and T. Dirjam, "PENERAPAN METODE TERM FREQUENCY INVERSE DOCUMENT FREQUENCY (TF-IDF) DAN COSINE SIMILARITY PADA SISTEM TEMU KEMBALI INFORMASI UNTUK MENGETAHUI SYARAH HADITS BERBASIS WEB (STUDI KASUS: SYARAH UMDATIL AHKAM)," URNAL TEKNIK FORMATIKA, vol. 11, p. 157, 2018.

[16] H. C. Husada and A. S. Paramita, "Analisis Sentimen Pada Maskapai Penerbangan di Platform Twitter Menggunakan Algoritma Support Vector Machine (SVM)," TEKNIKA, vol. 10, p. 20, 2021.

[17] Y. T. Pratama, A. F. Bachtiar and N. Y. Setiawan, "Analisis Sentimen Opini Pelanggan Terhadap Aspek Pariwisata Pantai Malang Selatan Menggunakan TF-IDF dan Support Vector Machine," Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, vol. 2, p. 6246, 2018.

[18] O. H. Rahman, G. Abdillah and A. Komarudin, "Klasifikasi Ujaran Kebencian pada Media Sosial Twitter Menggunakan Support Vector Machine," JURNAL RESTI (Rekayasa Sistem dan Teknologi Informasi), vol. 5, p. 20, 2021.

[19] Y. Prayoginingsih and R. P. Kusumawardani, "Klasifikasi Data Twitter Pelanggan Berdasarkan Kategori myTelkomsel Menggunakan Metode Support Vector Machine (SVM) Studi Kasus: Telekomunikasi Selular," Jurnal Sisfo, vol. 7, p. 85, 2018.

[20] N. Fitriyah, B. Warsito and D. A. I. Maruddani, "ANALISIS SENTIMEN GOJEK PADA MEDIA SOSIAL TWITTER DENGAN KLASIFIKASI SUPPORT VECTOR MACHINE (SVM)," JURNAL GAUSSIAN, vol. 9, p. 380, 2020.

[21] I. M. Yulietha, S. A. Faraby and Adiwijaya, "KLASIFIKASI SENTIMEN REVIEW FILM MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE," e-Proceeding of Engineering, vol. 4, pp. 4747 - 4748, 2017.

[22] L. Mutawalli, M. T. A. Zaen and W. Bagye, "KLASIFIKASI TEKS SOSIAL MEDIA TWITTER MENGGUNAKAN SUPPORT VECTOR MACHINE (Studi Kasus Penusukan Wiranto)," JIRE (Jurnal Informatika & Rekayasa Elektronika), , 2, p. 46, 2019.

[23] M. I. Fikri, Y. Azhar and T. S. Sabrila, "Perbandingan Metode Naïve Bayes dan Support Vector Machine pada Analisis Sentimen Twitter," SMATIKA Jurnal, vol. 10, p. 73, 2020.

[24] L. A. Andika, P. A. N. Azizah and Respatiwulan, "Analisis Sentimen Masyarakat terhadap Hasil Quick Count Pemilihan Presiden Indonesia 2019 pada Media Sosial Twitter Menggunakan Metode Naive Bayes Classifier," Indonesian Journal of Applied Statistics, vol. 2, p. 37, 2019.

[25] E. Patriya, "IMPLEMENTASI SUPPORT VECTOR MACHINE PADA PREDIKSI HARGA SAHAM GABUNGAN (IHSG)," Jurnal Ilmiah Teknologi dan Rekayasa, vol. 25, p. 31, 2020.

[26] C. D. Garcia, "Visualizing the effect of hyperparameters on Support Vector Machines," towards data science , 8 Februari 2021. [Online]. Available: https://towardsdatascience.com/visualizing-the-effect-of-hyperparameters-on-support-vector-machines-b9eef6f7357b. [Accessed 10 07 2021].