Bidirectional Long Short-Term Memory for Analysis of Public Opinion Sentiment on Government Policy During the COVID-19 Pandemic

Intan Nurma Yulita[1*], Ahmad Faaiz Al-Auza'i[2], Anton Satria Prabuwono[3], Asep Sholahuddin[4], Firman Ardiansyah[5], Indra Sarathan[6], Yusa Djuyandi[7]

Department of Computer Science, Universitas Padjadjaran, Sumedang, Indonesia[1, 2, 4]
Faculty of Computing and Information Technology in Rabigh, King Abdulaziz University, Rabigh, Saudi Arabia[3]
Magister of Management, Institut Teknologi dan Bisnis Ahmad Dahlan Lamongan, Indonesia[5]
Faculty of Cultural Sciences, Universitas Padjadjaran, Sumedang, Indonesia[6]
Faculty of Social and Political Science, Universitas Padjadjaran, Sumedang, Indonesia[7]

Abstract—One of the initiatives adopted by the Indonesian government to combat the development of COVID-19 in Indonesia is Community Activities Restrictions Enforcement. Many public opinions emerged, both for and against this policy. There are so many comments every second that it is certainly not easy to analyze them by reading each one by one. This task necessitates computer applications. Therefore, this study was conducted to produce an application that can help analyze public sentiment on the policy through social media, namely Twitter, into three classes: positive, neutral, and negative. The method used in this research is bidirectional long short-term memory (BiLSTM), one of the algorithms of deep learning. This study trains the model using the dataset, which consists of 10,486 tweets. The model receives an f1-score of 76.67 %. Thus, the model can be used to analyze public sentiment when the same policy is enforced. It can determine public acceptance of this policy. Thus, the system created in this research can be used as evaluation material for the government to review the policy when it is implemented in the future. However, this study concentrates on how to develop the sentiment analysis system and does not examine how the community responds to government policy.

Keywords—Sentiment analysis; COVID-19; BiLSTM; deep learning; government policy

I. INTRODUCTION

There are several obstacles that must be overcome in order for Indonesia's political system to function effectively given the country's status as a growing democracy with a sizable population [1, 2]. The political choices that are made by the government have significant repercussions, both directly and indirectly, on the daily lives of the people who live in Indonesia. This encompasses fundamental concerns like as the distribution of wealth, economic equality, human rights, the maintenance of a healthy environment, and the satisfaction of fundamental needs like education and health care. As a result, the leaders and policymakers in Indonesia have a significant challenge in terms of preserving political stability and developing appropriate policies.

The public mood regarding these policies is a reflection of the sentiments, opinions, and perspectives of many individuals and groups [3]. The vast majority of these sensations may be freely communicated through social media, online news platforms, or online discussion forums, which are increasingly reflecting people's emotions in real-time [4]. This point of view should not be ignored by the government, which is the organization that is responsible for formulating and enforcing policies. They should, rather, view this sentiment data as a useful source of knowledge that may assist them in better directing their policies and should do so by capitalizing on it.

The political discourse and polarization that exists in Indonesia have a significant impact on the decision-making processes there as well [5, 6]. Analysis of public sentiment can assist in detecting recurring patterns of support for or opposition to a certain program or leader. It gives an overview of the extent to which diverse political and social ideas exist in society, which is vital for establishing an effective communication strategy and defusing political tensions.

The influence of sentiment analysis can also have an effect on how the public perceives the government in today's digital and social media-driven environment, which links the public at large [7,8]. It is much easier for governments to earn the support and trust of their constituents when they react rapidly to the sentiments and viewpoints expressed by the populace. On the other hand, apathy for public feelings or a refusal to respond to them might result in demonstrations.

As a result, sentiment analysis is not only a tool for analyzing public mood, but it is also a critical factor in the effective crafting of policies and sustaining the political stability that is required in Indonesia's ever-shifting process of democratization. The use of technology such as machine learning in sentiment analysis provides governments with more powerful and efficient tools to analyze large amounts of data and respond in a timely way. This can have a beneficial influence on the government's ability to be effective and responsive. As a result, having a grasp of and making use of sentiment analysis in the context of political policy in Indonesia is an absolute must in order to successfully navigate the political and social dynamics that will continue to emerge in the years to come.

In order to address this issue, the present study used sentiment analysis to evaluate the political strategies of the Indonesian government in response to the COVID-19 pandemic. The Indonesian government published the implementation of restrictions for community activities (PPKM) policy on July 3, 202. It is one of the government's measures to regulate community activities in order to combat the rise in COVID-19 cases in Indonesia. From July 3 until July 20, this policy—originally known as the PPKM Java-Bali emergency—was formally in effect. Then, on July 20, the president announced the extension of PPKM until the 25th and modified the designation of PPKM to level 4 during a virtual press conference. It was used to stop the spread of COVID-19 and minimize the requirement for inpatient care. The 44 areas and cities that made up the Java and Bali regions at first were the only ones to which it was applied. On July 12, however, the administration decided that it would also apply to non-Java and non-Bali areas after deliberating for more than a week. 15 cities and regencies in regions with the highest COVID-19 spread rates were covered by it outside of Java and Bali. The government set restrictions on people's mobility during the era. Residents, for example, must bring a minimum vaccination letter and the results of a PCR swab test or an antigen test indicating a negative result for COVID-19 when traveling by any mode of transportation [9].

The enactment of this policy caused controversy among the people of Indonesia. Some people supported this policy, but not a few people opposed it for certain reasons. Of course, the government issued this policy for the welfare of society. If it is discovered that the policy causes social unrest, the government must respond. The public's opinion of a policy can be seen through the comments that appear in the mass media [10]. One of them is Twitter, a social media site. There are so many comments every second that it is certainly not easy to analyze them by reading each one by one. This task necessitates computer applications. This application can facilitate the analysis of the sentiment that appears in all Twitter comments on this policy automatically and quickly. The computer analysis of textual opinions, attitudes, and emotions is known as sentiment analysis, also referred to as opinion mining [11]. Natural language processing, text analysis, computational linguistics, and biometrics are used to systematically identify, extract, quantify, and study emotional states and subjective information [12]. It is widely used for textual data to help businesses or organizations understand customers' needs based on their opinions of a product or service. The government needs this study's sentiment analysis to help it understand how the general public feels about PPKM policies.

Different techniques can be used to analyze sentiment [13]. One of the deep learning algorithms that are most frequently used is long short-term memory (LSTM) [14]. It is more accurate than the recurrent neural network (RNN) approach for problems involving long texts. It was developed to solve the long-term memory issue brought on by fading gradients while using RNNs to process massive sequential data [15]. When a gradient has a very low value or is extremely close to zero, the network weight remains constant, indicating that the training process cannot be sustained. However, the LSTM algorithm still has some drawbacks. The standard LSTM only considers

the previous context. Despite the fact that text identification is seen not only in one direction from the previous context but also in the next context [16]. To address this issue, this work employs the bidirectional long short-term memory (BiLSTM) method. Based on the foregoing, it develops the BiLSTM model to automatically categorize the public sentiment toward PPKM policies during the COVID-19 pandemic. The data came from social media users' tweets on PPKM. This model is a fundamental component of computer algorithms that automatically categorize data tweets into three groups. Thus, there is no need to read each remark individually. Since every second there are so many comments. This categorization is reliant on language hence the success of an application cannot be directly applied if a different language is used. This application's outcomes are supposed to let the government observe the response of the Indonesian people to policies created more quickly and readily, so that they may be examined and utilized as material to consider when establishing a policy in the future. This study concentrates on how to develop the system and does not examine how the community responds to government policy. A second contribution of this study is the generation of a novel dataset that is unique to the Indonesian language for the PPKM topic.

II. RELATED WORKS

Sentiments are the underlying feelings, attitudes, assessments, or emotions underpinning an individual's viewpoint. Sentiment orientation might be positive, negative, or neutral. A sentiment target sometimes referred to as an opinion target, is the entity or feature of the entity about which the feeling is conveyed. The computational study of attitudes, opinions, judgments, and feelings concerning things and their features that are expressed in text is known as sentiment analysis. In general, sentiment analysis commonly uses two approaches: the rule-based/lexicon-based approach [17] and the machine learning/supervised learning approach [18]. By concentrating on the structure and features of the social network, sentiment analysis, and dangers including spam, bots, false news, and hate speech, Antonakaki et al. provide an effort to map the current study themes in Twitter [19]. Best practices for data sampling and access are also presented, in addition to Twitter's fundamental data model. This overview provides a foundation for the application of computational methods like Graph Sampling, Natural Language Processing, and Machine Learning in various fields.

COVID-19-related sentiment analysis research has also been extensively explored in recent years. Ridwan et al. collected English tweets that talked about COVID-19 and were geolocated as "Singapore." [20]. The system is based on the VADER lexicon-based classifier and the emotions from the pre-trained recurrent neural networks to find correlations between real-world events and changes in sentiment over the whole time period. The results of the sentiment analysis showed that about half of the tweets in the dataset were positive, and about a quarter of the tweets were negative or neutral. Topic modeling also showed that most of the talk about COVID-19 on Twitter was about staying home and that these conversations were mostly positive, which contributed to the overall positive mood during the study period. Overall, the results showed that the community supported the steps taken

by the Singapore government during COVID-19. This was clear from the positive comments on Twitter.

Sattar et al. use Twitter data to figure out how people feel about different COVID-19 vaccines [21]. It used the Twitter API and the Python library Tweepy to get about 1.2 million tweets. The tweets were collected from April 10 to May 17, 2021. It only got English tweets, and we used NLTK to do more data analysis. It builds the forecasting model classifier with well-known machine learning regression algorithms: Support Vector Machine (SVM) for regression, k-Nearest Neighbor (KNN), Linear Regression (LR), Random Forest (RF), M5 model tree, Gaussian process for regression, and Multi-layer Perceptron (MLP). Even though some of the vaccines have side effects, we find that the general public is more positive than negative about them. The model for predicting vaccinations says that by the end of July 2021, 62.44% of the population will have had at least one dose of vaccine and 48% will be fully vaccinated. The prediction model says that 73.53% of adults will be partially vaccinated on Independence Day (4 July 2021). The findings give a way to measure how the public talks about the COVID-19 vaccination and how to live a healthy life during the pandemic. It classifies tweets into different emotions like inspired, happy, annoyed, sad, angry, afraid, etc.

Many other studies related to sentiment analysis in English have been carried out, such as in English [22], Arabic [23], Italian [24], and Hindi [25]. Moreover, Intan NY et al. did COVID-19-related research in Indonesia. The research addresses the COVID-19 problem in Indonesia, for policies related to vacations [26]. Government initiatives regarding this vacation program have resulted in a diversity of societal perspectives. Using a bidirectional encoder representation from transformers (BERT) technique, they apply sentiment analysis to this government policy. This research produced a new dataset on this subject. The data was taken from the YouTube comments area and classified into three categories: favorable, neutral, and negative. This study generated an F-score of 84.33 percent. Using Twitter data, more COVID-related studies conducted in Indonesia have been conducted [27, 28].

III. METHODOLOGY

Our case study deals with the implementation of government regulations to address the increase in COVID-19 cases in Indonesia. PPKM has gained support from a range of segments of Indonesian society. Twitter is a social media platform where the general public may express their ideas. This research was conducted through stages that included data collection from Twitter social media, data labeling, text preprocessing, data splitting, feature selection, feature extraction, modeling, and evaluation of the model that has been built. The proposed framework is illustrated in Fig. 1.
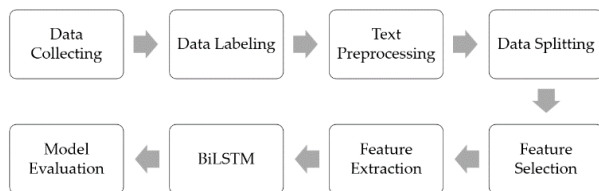

Fig. 1. Proposed framework.

A. Data Collecting

Data obtained from Twitter about the PPKM policy served as the study's subject. The unstructured, heterogeneous input is then categorized into attitudes that are positive, negative, or neutral. The procedure of data collection involved tweets posted by Twitter users between July 1 and August 19, 2021. 10,486 tweets, or 200 tweets each day for 50 days in this period make up the research data. Nevertheless, the obtained data for this study were only collected in Indonesian. Using the Python module snscrape, the scraping operation is carried out to retrieve the data. Government socialization of this policy began on July 1 and was completed on July 3, 2021. Fig. 2 depicts the peak of the topic discussion trend in Indonesia from July 4 through 10, which was the first week that this policy was put into effect.
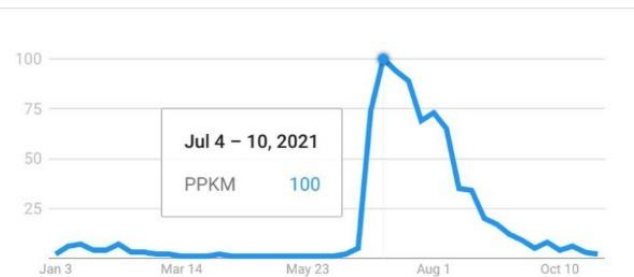

Fig. 2. Trends in PPKM policy tweets in Indonesia.

B. Data Labeling

The labeling of the collected tweet data was done manually, one tweet at a time. Several members of the Faculty of Cultural Sciences of Universitas Padjadjaran were implicated. The findings served as the foundation for this study. Negative tweets comprise profanity, expressions of hostility, criticism, wrath, fear, and despair, as well as expressions of disagreement with government-issued PPKM policies. In the meanwhile, tweets can be classified as positive if they contain terms with positive connotations, give support, demonstrate delight, encourage compliance, and comply with other PPKM criteria. If a tweet does not fall into the good or negative categories, it is labeled as neutral. In this phase, tweets that were deemed to be noise or unrelated to the opinions of Indonesians were also deleted. Its process involved manually deleting tweet data obtained from news accounts, government entities, nonprofits, and businesses. Additionally, tweets with promotions, ads, and tweets in regional languages were removed.

C. Text Preprocessing

Before the text data was processed to the next step, this stage cleaned the data of noise or modified the data's format to make the data processing easier [29]. Each language has its own characteristics hence the procedures and methodologies might differ between languages. The actions taken in this study, i.e.

- Case folding: This step transforms a word's individual letters into a single, consistent shape that may take the form of lowercase or uppercase letters [30].

- Noise reduction: During this step, the text was cleared of distracting elements including mentions, hashtags, links, numbers, symbols, punctuation, and emoticons.

- Tokenizing: This method divided the text into tokens, which were utilized to create practical features. Words in their entirety serve as the stage's tokens.

- Normalizing: In this step, the tokens were normalized to a standardized form. The Indonesian dictionary updated them with standard forms that replaced abbreviations, slang, and other non-standard forms.

- Stemming: By locating and eliminating any affixes present, this step reduced the word to its most basic form.

- Stop-word removal: In this phase, terms that were used excessively or whose definitions were irrelevant to learning are eliminated.

D. Data Splitting

This study used a 7:3 ratio to divide the dataset into training and testing data. 7,340 tweets were used for the training data, whereas 3,146 were used for the testing data. A 10 k-fold cross-validation procedure was then used to separate the training data from the validation data.

E. Feature Selection

Experiments on the minimum count threshold value were conducted during the feature selection step. The limit utilized to filter the existing characteristics based on a word's frequency of occurrence across the entire corpus was the min count value. The characteristics in question in sentiment analysis are the tokens or words that make up a phrase. At this point, irrelevant tokens were eliminated, namely those with too few occurrences

F. Feature Extraction

The process of feature extraction involves numerous initial feature changes that result in the production of new, more important features. It can be used in this situation to simplify the data representation and minimize complexity by treating each variable as a linear combination of the original input variables. The process of obtaining word lists from text data and turning them into a collection of features that the classifier may use is known as feature extraction on text [31]. Word2Vec is a method for feature extraction that appears to be an improvement over the drawbacks of the bag of words (BOW) method, which can create inaccurate models because it doesn't take into account word order or context. To create the appropriate context, it is important to take into account the arrangement of words in a document. When the same type and number of words are used in a document, the word order might yield distinct interpretations, which will impact the sentiment. Another strategy, word embedding, can be applied to get around this. It is a method of representing text in which similar-sounding words are represented similarly in a vector space [32]. In other words, it displays similar terms based on how closely connected the corpus is. It can often be divided into two categories: pre-diction-based embedding and frequency-based embedding. The count vector, TF-IDF, and co-occurrence vector techniques are the ones most frequently employed in frequency-based embedding. As for the frequently employed prediction-based approach, it is Word2Vec. It uses the context of the words around it to quantitatively identify word similarities. It makes vectors whose numerical

representations are based on things about words, like how they are used in a sentence. Also, it trains a corpus of text as input data and generates a vector list of words (embedding) as output from a model. The resulting word meanings and word embedding relationships have beneficial properties like vector arithmetic and are spatially represented [33]. It is a self-supervised learning system that gains knowledge from unlabeled input, like a group of texts. The learning technique makes use of conditional probabilities to anticipate certain phrases using some of the words they are surrounded by in a corpus of text. Word pairs that are close in proximity to one another in the text and were one-hot encoded (input, label) make up the training set.

Skip-gram architecture was used in this study since, according to research; it performs better [34]. During training, the skip-gram algorithm could be used to predict the word context of the input word by examining the words nearby. The size of the dimensions and the size of the window serve as the Word2vec hyperparameters that would be optimized. 50, 150, and 300 are the study's test dimensions, whereas 2, 5, 10, and 14 are the study's test window sizes.

G. BiLSTM

The RNN technique, which is utilized in the field of deep learning, has undergone development in the form of LSTM. It was intended to alleviate the vanishing and exploding gradient problems that occur when RNNs are employed to handle extensive sequential input. The cells can learn to recognize substantial input with the aid of the input gate, store it for a long time with the aid of the forget gate, learn to hold onto it for however long is necessary with the assistance of the forget gate, and then learn to delete it when needed. This explains why the algorithm is capable of picking up persistent patterns in time series, lengthy texts, and other data [35].

A memory cell with three gates requires the following gates to control the flow of information:

- Forget gate to decide what data should be removed from the memory cell or not,

- Input gate, which decides what data should be entered into the memory cell,

- Output gates are used to decide the output based on the input and memory cell.

LSTM's first step is to decide what data will be eliminated from the cell state. The choice is made by the sigmoid activation function that takes place in the forget gate layer. The forget gate will take input from ht-1 and xt and output a value between 0 and 1. While 0 indicates that the information will be stored from Ct-1, and 1 indicates that it will be removed.

The information that will be kept in the cell state must then be decided. It is split into two sections. The input gate layer, also known as the sigmoid activation function, first decides what data will be updated. Following that, the vector Ct of new candidate values that can be added to the state is created by the tanh activation function. The outcomes of the sigmoid and tanh activation functions will be merged in the following step, and their values will be utilized to update the state. The subsequent step is to transform the prior cell state, Ct-1, into the new cell

state, Ct. At this point, the value of the new candidate will be added to the prior state after being multiplied by the output from the input gate, and the output from the forget gate. At this stage, processes such as the removal of unimportant information and the addition of significant actual information into the cell state, as defined in the preceding step, occur [36]. The filtered cell state determines the output outcomes. The sigmoid activation function will initially mix input and data from the preceding hidden layer to determine which portion of the cell state should be created. The output from the sigmoid is then multiplied by the output from the tanh activation function (to obtain a value between -1 and 1), yielding only the preset information. Each time step of the LSTM cell's overall process provides two outputs: Ot, which serves as the time step's output and is used as a hidden state, and Ct, which creates a new cell state and is passed on to the following time step.

BiLSTM, a variant of the LSTM algorithm, tries to observe a certain sequence from front to back or back to front. The network creates a context for each character in the text by utilizing their past and future. The goal is to maximize the utilization of an input sequence by stepping through the input step time both forward and backward. In order to create two layers side by side using this design, the first layer is duplicated repeatedly. The input sequence is then fed into the first layer, and the second layer receives a reverse copy of the input sequence. This strategy was created a while back as a general strategy to enhance the functionality of recurrent neural networks [37]. the forward layer and backward layer's information flow. It is typically employed when the order of the data is a factor. It is justified in the context of voice recognition since research suggests that rather than a one-way linear interpretation, humans perceive what is spoken in the context of complete speech. Although it was initially designed for speech recognition, the bidirectional is now a crucial component of LSTM sequence prediction as a method to enhance model performance.

The BiLSTM model neural network layer in this study can be illustrated in Fig. 3. This study's BiLSTM model was developed using the Keras module, which may be obtained at https://keras.io (accessed on 18 February 2020). The first layer was the embedding layer whose embedding matrix values were obtained from the Word2vec model. The embedding matrix resulting from Word2vec was used as the initialization value of the embedding layer's weight which was then updated during the training process. Input on this layer was in the form of tweets which were represented in word index sequences. The sequence had a maximum length of 65 words. Tweet data consisting of less than 65 words was a padded 0 in front of it, while data tweets that were more than 65 words will be truncated so that the length of each tweet sequence is the same. When a shape tuple has a none dimension, it indicates that the network is open to inputs of any dimension.

A 65 x 150-pixel matrix representing each piece of data was the embedding layer's output which is from the feature extraction stage. The maximum length for each data point was 65, and the maximum length for each word's vector representation was 150. The matrix served as the BiLSTM layer's input. The forward LSTM layer and the back-ward LSTM layer are the two LSTM layers that make up the

BiLSTM layer, which is made up of one layer altogether. The output of the BiLSTM layer has a size of 65 32 thanks to the combination of the 16 units from each of these LSTM layers with a dropout of 0.5. The GlobalMaxPool1D layer was then given the output from the BiLSTM layer. By taking the largest value in the temporal dimension, this layer was able to down-sample two-dimensional data into one dimension, producing an output with 32 dimensions. It then moves on to the layer for batch normalization. It altered the input to make it uniform. Each input variable's statistics were tracked by the layer during training and utilized to normalize the data. Through a straightforward regularization effect, it can expedite the training process and, in some situations, improve model performance.
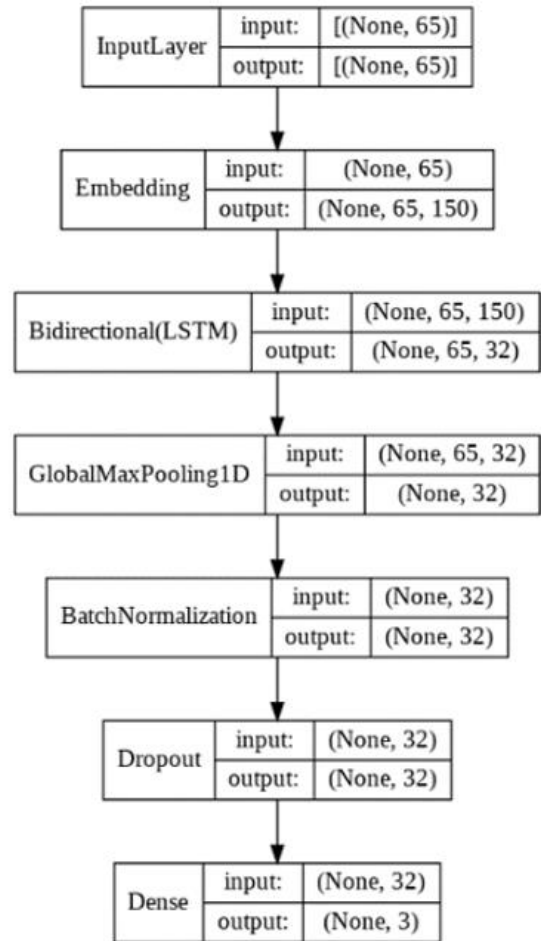

Fig. 3. BiLSTM model architecture.

TABLE I. HYPERPARAMETER VALUES

Hyperparameter	Value
Optimizer	Adam
Merge mode	Concat
Dimension size	150
Activation	Softmax
Max sequence length	65
Max epoch	50

Thus, the layer based on the dropout was a safeguard against overfitting [38], and the resulting layer was dense, fully connected, and had three units. The layer served as the output layer for the Softmax activation function, which created class probabilities from the input with values ranging from 0 419 to 1 for each class. The prediction of input data had the three classes' greatest value making the following step reasonable. It trained the model's trainable parameters defined and assembled earlier. Early stopping was used to avoid overfitting during the model training process using data by assessing the hyperparameter batch size with epoch 50. Table I displays the hyperparameter values used in this investigation. The grid search method in finding optimal hyperparameters was used, in which every combination of predetermined hyperparameters was tried to select the combination that produced the best f1-score value.

H. Model Evaluation

After the BiLSTM model had been successfully created, the next step was to evaluate the model using the confusion matrix values. The metric in this study was the f1-score. It is one of the assessment measures that is often used in sentiment analysis scores. This metric is used to quantify how well a model is able to categorize comments into various sentiment categories, such as positive, negative, or neutral. The F1-score is a metric that offers an overall view of the quality of the model classification since it combines both precision and recall into a single value. Because it strikes a good balance between precision and recall, it is an appropriate tool for use in situations in which the ratio of false positives to false negatives is a significant consideration in sentiment analysis. When identifying sentiment, a high F1 score shows that the model has a high level of precision as well as recall.

IV. RESULTS

The hyperparameters that were analyzed to create the optimal BiLSTM model were the number of BiLSTM layers, the number of BiLSTM units, batch size, dropout rate, and learning rate.

A. Analysis of the Number of Layers

With dropout rate=0.1, batch size=16, and learning rate=0.01, one and BiLSTM layers one and two were stacked against one another and evaluated in this study. Table II shows that using one BiLSTM layer as opposed to two leads in a higher f1-score. The findings of this experiment demonstrate that employing one layer of BiLSTM instead of two layers led to a higher f1-score. It is clear that the model with two layers of BiLSTM could not ensure that the model's performance would be enhanced. Because it yielded improved f1-scores, utilizing just a BiLSTM layer was sufficient for this study. The capacity of the model to learn can theoretically be increased by adding layers to the neural network. However, the model can extract more intricate patterns from the supplied data because of its expanded learning capacity. In general, this is advantageous because the model learns more and becomes better at properly predicting data. On the other side, there is a chance that the created model will be overfitted as a result. According to this criterion, the model performed well when predicting data that has been thoroughly researched but poorly when predicting data that has never been observed.

TABLE II. ANALYSIS OF THE NUMBER OF LAYERS

Number of layer	LSTM Unit	F1 score
1	16	74.49%
1	64	74.27%
2	16	72.72%
2	64	73.40%

B. Analysis of BiLSTM Unit

Table III demonstrates that an f1-score with a declining trend was obtained by in-creasing the hidden units in the LSTM layer. The model with 16 hidden units generated the model with the highest f1-score, 74.49 %. In this study, 16, 64, and 128 BiLSTM units were examined. Table III gives an overview of the test outcomes that have been evaluated. The test results demonstrate that an f1-score value with a falling trend will be produced by increasing the number of hidden units in the BiLSTM layer. The model with 16 hidden units yields the model with the highest f1-score, or 74.49 %, compared to the other models. It demonstrated that utilizing 16 hidden units in each BiLSTM layer—or 32 units in the BiLSTM layer—was sufficient for the model to extract patterns from the supplied dataset. A higher hidden unit count also did not result in a higher f1-score. It is due to the fact that a model with an excessively large number of hidden units may get overfitted, have a low f1-score in the validation data, and be overly complex.

TABLE III. RESULTS OF THE BiLSTM UNIT

Number of layers	BiLSTM Unit	F1 score
1	16	74.49%
1	64	74.27%
1	128	73.46%

C. Analysis of Dropout

The best outcomes are displayed in Table IV when a dropout of 0.5 was used. It suggests that the optimal model would consist of a BiLSTM layer, 16 LSTM units, and a dropout of 0.5, which would effectively reduce overfitting, raise the average f1-score value, and offer the best model. The model's ability to handle overfitting was less ideal due to the dropout value being too small. Because too many neurons were eliminated during the training phase due to an excessively high dropout number, the final model was subpar.

TABLE IV. ANALYSIS OF THE DROPOUT RATE

Dropout rate	LSTM Unit	F1 score
0.1	16	74.49%
0.3	16	74.08%
0.5	16	74.57%

D. Analysis of Batch Size

According to Table V, batch size 32 produced the highest f1-score when compared to the other sizes. The batch size value was about right—it wasn't either too big or too tiny. Larger data sets result in faster model convergence, but since there are so many of them, the model finds it challenging to

identify patterns in the data. Otherwise, if it is too little, the algorithm might not take into account the true level of variance in the sampled distribution, which could lead to a noisy training process.

TABLE V.   ANALYSIS OF THE BATCH SIZE

Dropout rate	Batch size	F1 score
0.5	16	74.57%
0.5	32	75.37%
0.5	64	75.08%

E. Analysis of Learning Rate

Table VI shows that when the learning rate was set to 0.001, the model performed at its best. If the learning rate was excessively sluggish and required more repetitions, convergence happened more slowly. If the learning rate was too high, the convergence would occur quickly but the step size would be enormous and the optimal point would be exceeded. The table shows that a model with a learning rate of 0.0001 has a substantially lower f1-score.

TABLE VI.   ANALYSIS OF THE LEARNING RATE

Batch size	Learning rate	F1 score
32	0,01	75.37%
32	0,001	76.34%
32	0,0001	70.40%

F. Model Evaluation

The tests with a 10-fold cross-validation technique is presented in Sub-chapters 4.1 to 4.5. The model was then applied to the test data that had been previously separated. 3,146 data points total—including 1,510 negative class data, 1,236 neutral class data, and 400 positive class data—were utilized to evaluate the final model. Fig. 4 demonstrates that in epoch 8, the model obtained the best f1-score. The overfitting that caused the early termination mechanism to end the training process automatically in epoch 9 can be seen. It also demonstrates that there are not many differences in the curves between the training and validation sets of data, indicating that the developed model performs reasonably well. The results of the final model test on the test data shown in the confusion matrix are shown in Fig. 5.
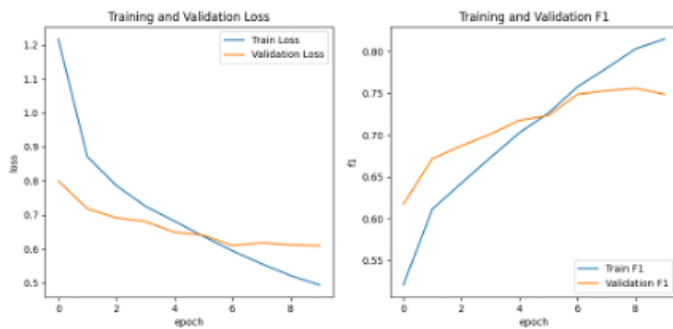

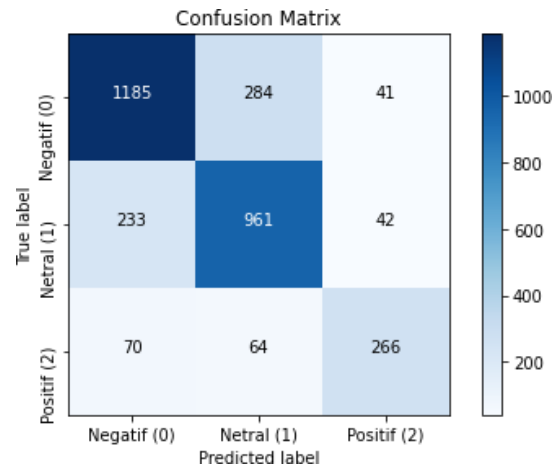Fig. 4.   Graph of F1-Score and loss.

Fig. 5.   Confusion matrix.

Also, Table VII displays the precision, recall, and f1-score for each class. The classification result shows metrics for each negative, neutral, and positive class. It is clear that negative classes frequently have metrics that are superior to those of other classes. The built-in model's performance receives a f1-score of 76.67 %.

TABLE VII.   MODEL EVALUATION RESULTS

Metric	Negative	Neutral	Positive
Precision	79.64%	73.41%	76.22%
Recall	78.48%	77.75%	66.50%
F1-score	79.05%	75.52%	71.03%

Fig. 6.   PPKM trends in November 2021.

G. Sentiment Prediction

This research focuses on the development of computer applications for PPKM-related sentiment analysis in Indonesia. The BiLSTM model was derived from the collected ground truth data in this study. By this model, various sentiment analysis-related data, specifically PPKM-related data, may be automatically classified. An example of test data is PPKM comments data collected in November 2021. In order to avoid an upsurge in COVID-19 cases in Indonesia before the year-end holidays, the government raised the PPKM level once again at the end of 2021. As a result, some Indonesians became quite interested in the PPKM subject. Fig. 6 displays a Google Trends graph of the PPKM keyword's search volume in November 2021. The graph demonstrates that on November 18, the PPKM again declared a peak. On November 17, 2021, the government declared that PPKM level 3 would once again be implemented. 31,788 tweets in total were collected. Based

on this research, there are 10,370, 13,836, and 7.58, respectively, fell into the negative, neutral, and positive categories. In November 2021, a neutral class of 43.5 percent dominated the public's opinion on the PPKM policy. It was followed by a negative class of 32.6 percent, while a small percentage of 23.9 percent expressed a favorable attitude. The majority of tweets that were categorized as neutral provide news or information about PPKM policy. The tweet content that was labeled as unfavorable typically reflects the public's dissatisfaction and frustration with the effects of PPKM's implementation. It demonstrates that the majority of Indonesians disapprove of the PPKM policy, which the government can utilize as evaluation information for reviewing current and future policies.

V. CONCLUSION

The topic of the study was data acquired from Twitter on the PPKM policy. The input is then classified as positive, negative, or neutral. The technique for collecting data involves tweets written by Twitter users between July 1 and August 19, 2021. The research data consists of 10,486 tweets, or 200 tweets every day over 50 days. This is the study's contribution. It is the creation of a new dataset that is unique to the Indonesian language in relation to the PPKM issue.

Also, this study creates the BiLSTM model to automatically classify public opinion on PPKM policies during the COVID-19 epidemic. The best model, with a final f1-score of 76.67 %. Reading each comment individually is unnecessary. Since every second, so many comments are posted. This categorization is dependent on language; hence, the success of an application cannot be directly applied when a different language is utilized. The results of this application are intended to allow the government to monitor the response of the Indonesian people to newly formed policies more quickly and easily, so that they may be analyzed and used as future policy-making material. This research focuses on developing the system and does not investigate how the community reacts to government policies. Last, this article only discusses one of the deep learning algorithms, specifically the BiLSTM method. With the rapid growth of deep learning, however, research may be conducted on the most recent algorithms with the goal of improving performance.

The focus of future work will be on developing more sophisticated pre-processing methods to improve the accuracy of public opinion sentiment on government policy during the COVID-19 pandemic, as well as exploring the application of transfer learning techniques to strengthen understanding of complex patterns in Indonesian language text data.

ACKNOWLEDGMENT

REFERENCES

[1] E. Warburton, "Deepening Polarization and Democratic Decline in Indonesia," in Political Polarization in South and Southest Asia Old Divisions, New Dangers, 2020.

[2] E. Warburton and E. Aspinall, "Explaining Indonesia's Democratic Regression," Southeast Asia, vol. 41, no. 2, 2019.

[3] X. Dong and Y. Lian, "A review of social media-based public opinion analyses: Challenges and recommendations," Technol Soc, vol. 67, 2021, doi: 10.1016/j.techsoc.2021.101724.

[4] Y. Lin, "Social media for collaborative planning: A typology of support functions and challenges," Cities, vol. 125, 2022, doi: 10.1016/j.cities.2022.103641.

[5] E. E. Supriyanto and J. Saputra, "Big Data and Artificial Intelligence in Policy Making: A Mini-Review Approach," International Journal of Advances in Social Sciences and Humanities, vol. 1, no. 2, 2022, doi: 10.56225/ijassh.v1i2.40.

[6] S. Partelow and A. O. Manlosa, "Commoning the governance: a review of literature and the integration of power," Sustain Sci, vol. 18, no. 1, 2023, doi: 10.1007/s11625-022-01191-2.

[7] P. Grover, A. K. Kar, and Y. Dwivedi, "The evolution of social media influence - A literature review and research agenda," International Journal of Information Management Data Insights, vol. 2, no. 2, 2022, doi: 10.1016/j.jjimei.2022.100116.

[8] E. Kubin and C. von Sikorski, "The role of (social) media in political polarization: a systematic review," Ann Int Commun Assoc, vol. 45, no. 3, 2021, doi: 10.1080/23808985.2021.1976070.

[9] F. Sanjaya, "A Critical Review of the RT-PCR Test Policy and Antigen Swab in Indonesia," in Proceedings of the International Conference on Communication, Policy and Social Science (InCCluSi 2022), 2022. doi: 10.2991/978-2-494069-07-7_48.

[10] S. Hansun, A. Suryadibrata, R. Nurhasanah, and J. Fitra, "Tweets Sentiment on Ppkm Policy as a Covid-19 Response in Indonesia," Indian Journal of Computer Science and Engineering, vol. 13, no. 1, 2022, doi: 10.21817/indjcse/2022/v13i1/221301302.

[11] A. Yadav and D. K. Vishwakarma, "Sentiment analysis using deep learning architectures: a review," Artif Intell Rev, vol. 53, no. 6, 2020, doi: 10.1007/s10462-019-09794-5.

[12] M. R. Hasan, M. Maliha, and M. Arifuzzaman, "Sentiment Analysis with NLP on Twitter Data," in 5th International Conference on Computer, Communication, Chemical, Materials and Electronic Engineering, IC4ME2 2019, 2019. doi: 10.1109/IC4ME247184.2019.9036670.

[13] K. Afifah, I. N. Yulita, and I. Sarathan, "Sentiment Analysis on Telemedicine App Reviews using XGBoost Classifier," in 2021 International Conference on Artificial Intelligence and Big Data Analytics, ICAIBDA 2021, 2021. doi: 10.1109/ICAIBDA53487.2021.9689762.

[14] Z. Jin, Y. Yang, and Y. Liu, "Stock closing price prediction based on sentiment analysis and LSTM," Neural Comput Appl, vol. 32, no. 13, 2020, doi: 10.1007/s00521-019-04504-2.

[15] A. Shewalkar, D. nyavanandi, and S. A. Ludwig, "Performance Evaluation of Deep neural networks Applied to Speech Recognition: Rnn, LSTM and GRU," Journal of Artificial Intelligence and Soft Computing Research, vol. 9, no. 4, 2019, doi: 10.2478/jaiscr-2019-0006.

[16] I. N. Yulita, M. I. Fanany, and A. M. Arymurthy, "Combining deep belief networks and bidirectional long short-term memory case study: Sleep stage classification," in International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), 2017. doi: 10.11591/eecsi.4.1051.

[17] A. Mitra, "Sentiment Analysis Using Machine Learning Approaches (Lexicon based on movie review dataset)," Journal of Ubiquitous Computing and Communication Technologies, vol. 2, no. 3, 2020, doi: 10.36548/jucct.2020.3.004.

[18] P. K. Jain, R. Pamula, and G. Srivastava, "A systematic literature review on machine learning applications for consumer sentiment analysis using online reviews," Computer Science Review, vol. 41. 2021. doi: 10.1016/j.cosrev.2021.100413.

[19] D. Antonakaki, P. Fragopoulou, and S. Ioannidis, "A survey of Twitter research: Data model, graph structure, sentiment analysis and attacks," Expert Syst Appl, vol. 164, 2021, doi: 10.1016/j.eswa.2020.114006.

[20] K. Mohamed Ridhwan and C. A. Hargreaves, "Leveraging Twitter data to understand public sentiment for the COVID-19 outbreak in Singapore," International Journal of Information Management Data Insights, vol. 1, no. 2, 2021, doi: 10.1016/j.jjimei.2021.100021.

[21] N. S. Sattar and S. Arifuzzaman, "Covid-19 vaccination awareness and aftermath: Public sentiment analysis on twitter data and vaccinated population prediction in the usa," Applied Sciences (Switzerland), vol. 11, no. 13, 2021, doi: 10.3390/app11136128.

[22] C. Baden, C. Pipal, M. Schoonvelde, and M. A. C. G. van der Velden, "Three Gaps in Computational Text Analysis Methods for Social Sciences: A Research Agenda," Commun Methods Meas, vol. 16, no. 1, 2022, doi: 10.1080/19312458.2021.2015574.

[23] L. Abualigah, H. E. Alfar, M. Shehab, and A. M. A. Hussein, "Sentiment Analysis in Healthcare: A Brief Review," in Studies in Computational Intelligence, vol. 874, 2020. doi: 10.1007/978-3-030-34614-0_7.

[24] R. Catelli, S. Pelosi, and M. Esposito, "Lexicon-Based vs. Bert-Based Sentiment Analysis: A Comparative Study in Italian," Electronics (Switzerland), vol. 11, no. 3, 2022, doi: 10.3390/electronics11030374.

[25] P. Chauhan, N. Sharma, and G. Sikka, "The emergence of social media data and sentiment analysis in election prediction," J Ambient Intell Humaniz Comput, vol. 12, no. 2, 2021, doi: 10.1007/s12652-020-02423-y.

[26] I. N. Yulita, V. Wijaya, R. Rosadi, I. Sarathan, Y. Djuyandi, and A. S. Prabuwono, "Analysis of Government Policy Sentiment Regarding Vacation during the COVID-19 Pandemic Using the Bidirectional Encoder Representation from Transformers (BERT)," Data (Basel), vol. 8, no. 3, 2023, doi: 10.3390/data8030046.

[27] S. H. Sahir, R. S. Ayu Ramadhana, M. F. Romadhon Marpaung, S. R. Munthe, and R. Watrianthos, "Online learning sentiment analysis during the covid-19 Indonesia pandemic using twitter data," IOP Conf Ser Mater Sci Eng, vol. 1156, no. 1, 2021, doi: 10.1088/1757-899x/1156/1/012011.

[28] M. Rahardi, A. Aminuddin, F. F. Abdulloh, and R. A. Nugroho, "Sentiment Analysis of Covid-19 Vaccination using Support Vector Machine in Indonesia," International Journal of Advanced Computer Science and Applications, vol. 13, no. 6, 2022, doi: 10.14569/IJACSA.2022.0130665.

[29] L. Hickman, S. Thapa, L. Tay, M. Cao, and P. Srinivasan, "Text Preprocessing for Text Mining in Organizational Research: Review and Recommendations," Organ Res Methods, vol. 25, no. 1, 2022, doi: 10.1177/1094428120971683.

[30] M. F. R. Abu Bakar, N. Idris, L. Shuib, and N. Khamis, "Sentiment Analysis of Noisy Malay Text: State of Art, Challenges and Future Work," IEEE Access, vol. 8, 2020, doi: 10.1109/ACCESS.2020.2968955.

[31] S. Hakak, M. Alazab, S. Khan, T. R. Gadekallu, P. K. R. Maddikunta, and W. Z. Khan, "An ensemble machine learning approach through effective feature extraction to classify fake news," Future Generation Computer Systems, vol. 117, 2021, doi: 10.1016/j.future.2020.11.022.

[32] P. G. Shivakumar and P. Georgiou, "Confusion2Vec: Towards enriching vector space word representations with representational ambiguities," PeerJ Comput Sci, vol. 2019, no. 6, 2019, doi: 10.7717/peerj-cs.195.

[33] A. Hernandez-Suarez et al., "Using twitter data to monitor natural disaster social dynamics: A recurrent neural network approach with word embeddings and kernel density estimation," Sensors (Switzerland), vol. 19, no. 7, 2019, doi: 10.3390/s19071746.

[34] F. A. O. Santos, H. T. Macedo, T. D. Bispo, and C. Zanchettin, "Morphological skip-gram: Replacing fasttext characters n-gram with morphological knowledge," Inteligencia Artificial, vol. 24, no. 67, 2021, doi: 10.4114/intartif.vol24iss67pp1-17.

[35] K. Khalil, O. Eldash, A. Kumar, and M. Bayoumi, "Economic LSTM Approach for Recurrent Neural Networks," IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 66, no. 11, 2019, doi: 10.1109/TCSII.2019.2924663.

[36] F. Landi, L. Baraldi, M. Cornia, and R. Cucchiara, "Working Memory Connections for LSTM," Neural Networks, vol. 144, 2021, doi: 10.1016/j.neunet.2021.08.030.

[37] A. Sherstinsky, "Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network," Physica D, vol. 404, 2020, doi: 10.1016/j.physd.2019.132306.

[38] M. M. Agüero-Torales, J. I. Abreu Salas, and A. G. López-Herrera, "Deep learning and multilingual sentiment analysis on social media data: An overview," Appl Soft Comput, vol. 107, 2021, doi: 10.1016/j.asoc.2021.107373.