data

MDPI

*Article*

# Analysis of Government Policy Sentiment Regarding Vacation during the COVID-19 Pandemic Using the Bidirectional Encoder Representation from Transformers (BERT)

**Intan Nurma Yulita [1,*], Victor Wijaya [2], Rudi Rosadi [2], Indra Sarathan [3], Yusa Djuyandi [4] and Anton Satria Prabuwono [5]**

[1] Research Center for Artificial Intelligence and Big Data, Universitas Padjadjaran, Bandung 40132, Indonesia
[2] Department of Computer Science, Faculty of Mathematics and Natural Sciences, Universitas Padjadjaran, Sumedang 45363, Indonesia
[3] Faculty of Cultural Sciences, Universitas Padjadjaran, Sumedang 45363, Indonesia
[4] Faculty of Social and Political Science, Universitas Padjadjaran, Sumedang 45363, Indonesia
[5] Faculty of Computing and Information Technology in Rabigh, King Abdulaziz University, Rabigh 21911, Saudi Arabia
* Correspondence: intan.nurma@unpad.ac.id

**Abstract:** To address the COVID-19 situation in Indonesia, the Indonesian government has adopted a number of policies. One of them is a vacation-related policy. Government measures with regard to this vacation policy have produced a wide range of viewpoints in society, which have been extensively shared on social media, including YouTube. However, there has not been any computerized system developed to date that can assess people's social media reactions. Therefore, this paper provides a sentiment analysis application to this government policy by employing a bidirectional encoder representation from transformers (BERT) approach. The study method began with data collecting, data labeling, data preprocessing, BERT model training, and model evaluation. This study created a new dataset for this topic. The data were collected from the comments section of YouTube, and were categorized into three categories: positive, neutral, and negative. This research yielded an F-score of 84.33%. Another contribution from this study regards the methodology for processing sentiment analysis in Indonesian. In addition, the model was created as an application using the Python programming language and the Flask framework. The government can learn the extent to which the public accepts the policies that have been implemented by utilizing this research.

**Keywords:** sentiment analysis; bidirectional encoder representation from transformers; government policy; COVID-19

check for updates

**Citation:** Yulita, I.N.; Wijaya, V.; Rosadi, R.; Sarathan, I.; Djuyandi, Y.; Prabuwono, A.S. Analysis of Government Policy Sentiment Regarding Vacation during the COVID-19 Pandemic Using the Bidirectional Encoder Representation from Transformers (BERT). *Data* **2023**, *8*, 46. https://doi.org/10.3390/data8030046

Academic Editor: Hiram Calvo

Received: 31 December 2022
Revised: 4 February 2023
Accepted: 21 February 2023
Published: 23 February 2023

## 1. Introduction

The global increase in Coronavirus Disease 2019 (COVID-19) cases has continued worldwide, notably in Indonesia [1]. As a result, the Indonesian government must undertake steps to limit the rising number of cases [2]. To address the high number of confirmed positive cases, the government implemented a variety of programs to reduce the number of positive confirmed cases. One of the policy considerations raised by the administration concerned vacations [3]. This strategy was implemented as an effort to avoid the formation of new clusters as a result of people taking extended vacations. To address this issue, the government issued Circulation No. 12 of 2021 concerning the Provisions for Domestic Travel During the 2019 Coronavirus Disease (COVID-19) Pandemic [4] and Circular No. 13 of 2021 concerning the Elimination of Homecoming for Eid al-Fitr 1442 Hijri and Efforts to Control the Spread of Coronavirus Disease 2019 (COVID-19) During the Holy Month of Ramadan 1442 Hijri [5]. The government was concerned that the number of positive COVID-19 cases might continue to grow, therefore the administration took measures designed to maintain a decline in positive cases by introducing this program.

Undoubtedly, these policies attracted diverse viewpoints from the general populace. The numerous responses were generally categorized into three groups: agreeing (pro), disagreeing (contra), and indifferent. These comments were frequently disseminated in print media, online news channels, and social media. One such social media site is YouTube [6], which features a comments area where varied views, ideas, opinions, and public opinion on government policies may be expressed. YouTube is the top social media platform in Indonesia, with most of the population having access to it. As a result, YouTube may be used as a source of information to gauge the public's reaction to government efforts to fight the COVID-19 pandemic. To determine the public's reaction to this policy, a sentiment analysis needs to be used in the YouTube video comment column. For the public impact worldwide, commercial choices, and policy formation, large-scale extractions of human emotions and reactions from social media networks are vital [7].

Sentiment is a description of an emotion or an emotionally significant occurrence. It may additionally be described as an individual's viewpoint (it is generally subjective). In contrast, sentiment analysis is a method for measuring opinions, emotions, and subjectivity in written texts [8]. It is sometimes referred to as opinion mining, and a topic of research that employs natural language processing (NLP), text mining, computational linguistics, and a measurement to identify, release, evaluate, and explore emotional states and subjective data [9]. The use of sentiment analysis is employed to evaluate the speaker's disposition based on their emotional responses, which will indicate the emotional response of the speakers as they talk. It is possible to construct a system that detects and extracts text-based opinions using sentiment analysis. Currently, sentiment analysis may be used to investigate various social media opinions so as to determine an individual or group perspective on a certain topic. This evaluation is based on dialogue and evaluative discourse to examine attitudes and sentiments toward the associated brand [10,11]. In sentiment analysis, there are three key aspects to be considered, namely the topic, the polarity, and the opinion holder [12]. The topic refers to the relevance to the issue being discussed. Polarity describes the value of the opinions expressed, and whether they are favorable, negative, or neutral. The person who conveys an opinion is the opinion holder. According to the above description, it is a technique used to extract information in the form of an individual's opinion about a certain topic or event. Using sentiment analysis, the varied perspectives offered by each of these individuals will be interpreted and categorized. These categorization phases can be modified to match the aims of the research.

Machine learning is a subfield of artificial intelligence (AI) that allows machines to undertake human-like activities [13]. The existence of machine learning supports many human activities. Among the applications of machine learning is sentiment analysis [14–16]. Sentiment analysis is a science used to extract information in the form of an individual's opinion on a topic or an incident [17–19]. By using this kind of analysis, various public perspectives on a government's policy may be determined. Sentiment analysis extracts the opinions expressed by the public via social media, which is information reflecting the perspective of the community on a particular policy. Undoubtedly, the government's policy with regard to extended vacations during the COVID-19 pandemic would provoke a variety of responses among Indonesians. A portion of society agrees with and supports the government's policy to reduce the spread of COVID-19 [20]. On the other hand, there are individuals who oppose this program and there are also those who are uninterested in the topic. Using sentiment analysis to determine the public's reaction to this government program is therefore of value.

This study employs bidirectional encoder representations from transformers (BERT) to be used for sentiment analysis on this topic. It can educate robots to learn human speech patterns sentence-by-sentence, as opposed to the current method of learning word-by-word [21]. Google released the algorithm for the first time in October 2019. BERT is an NLP technique that mixes machine and human language [22]. BERT's application includes two-way learning, which increases the model learning. It is more accurate than the conventional model [23]. Consequently, the objective of this study is to automate the categorization of public sentiment toward government vacation policies during the COVID-19 pandemic. Considering that sentiment analysis has been developed to be used in English, this analysis is language dependent, so advances in processing technology in other languages cannot be applied directly to other languages. Therefore, this study also contributes to the methodology of sentiment analysis in Indonesian. It also provides a web-based application for sentiment analysis utilizing a trained BERT model. The anticipated significance of this study is that the government will be able to establish the policies taken during the COVID-19 outbreak when they are accepted by the people. Using sentiment analysis in this study will certainly make it easier to achieve this objective.

## 2. Related Works

As a means of connecting people and encouraging the exchange of ideas, information, and expertise, social media has created a number of online platforms. It is undeniable that social media platforms have more sway than ever before, and their prominence is on the rise [11]. Since many people use their devices and spend so much time consuming content on social media sites, these platforms are sometimes referred to as the Big Data of the world; thus, social and statistical research have concluded that they have a significant impact on users' habits. In terms of global use, YouTube, Facebook, Twitter, Instagram, and Reddit are among the top social media platforms. In spite of the vast amounts of information available on these sites, the material may have opposing impacts, including both good and negative psychological sway on users' lives [16]. It is possible that those who are addicted to social media use it to vent their frustrations and express their perspectives. Therefore, it is important to seek ways to convert these comments and postings into assets by utilizing sentiment analysis.

Studies on sentiment analysis have progressed significantly. This study has been conducted in a number of languages, including Arabic [24], Malaysian [25], Brazilian [26], Persian [27], German [28], Portuguese [29], Chinese [30,31], Urdu [32], Bengali [33], Vietnamese [34], Indonesian [35], and Lithuanian [36]. This kind of research is language dependent, since the success of technology in one language cannot be applied straight to other languages. Each language has its own characteristics, such as word formation, sentence structure, and style of language use. This is a barrier for sentiment analysis research, as each language requires a unique approach. Research produced by Xu et al. presents a cross-lingual technique to accommodate diverse languages in this research [37]. Their research advances not only language modeling but also the preprocessing techniques employed. Pradha et al. [38] proposed a method for effectively processing text input and developing an algorithm for training Support Vector Machine (SVM), Deep Learning (DL), and Naive Bayes (NB) classifiers to categorize tweets. When computing the sentiment score, they designed a system that lends greater weight to hashtags and more recently cleansed content. They compared the success of Google Now and Amazon Alexa using Twitter data. According to the findings, the stemming method is the most efficient. de Oliveira et al. [39], Sohrabi et al. [40], Alam et al. [41], and Resyanto et al. [42] also conducted research pertaining to additional text preparation.

Many classification methods have also been extensively used to address the issues presented in this research. Several traditional machine learning techniques that have been implemented include Naive Bayes [43,44], Support Vector Machine (SVM) [45], Decision Trees [46], Random Forests [47], and Regression [48]. This investigation leads to the context of Big Data as the era of social media becomes more sophisticated. The created method is also in the deep learning stage of processing [49,50]. Long Short-Term Memory [51], Convolutional Long Short-Term Memory [52–55], Bi-LSTM [56], and BERT [57–59] are some of the deep learning algorithms that have been developed.

## 3. Methodology

As a case study, this paper examines Indonesia's vacation regulations during the COVID-19 pandemic. In reaction to the impending vacation, the administration has chosen to impose this policy after examining the trend of positive COVID-19 cases. The government's plan is to reduce the spread of infectious illnesses in famous tourist destinations. The research activities were undertaken by collecting data, labeling data, processing data, and designing and building models to conduct sentiment analysis, and assessing the outcomes, as illustrated in Figure 1.

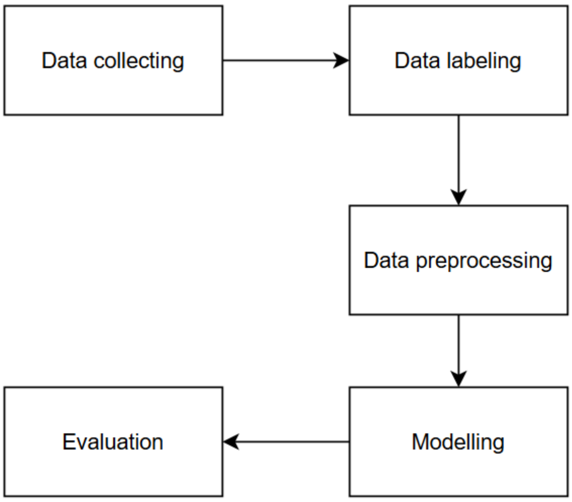


**Figure 1.** Proposed framework.

### *3.1. Data Collecting*

This study draws its data from the textual comments sections of YouTube social media videos highlighting government activities for the Eid holiday in 2021. The footage originated from user accounts of Indonesian news. Keywords linked with the 2021 Eid holiday were used to locate videos. From April to May 2021, the data were collected from the YouTube social media comments section. The selected comments are only in Indonesian. This timeframe was chosen because it corresponded with the height of public conversation around the 2021 Eid holiday policy. For this objective, we deployed the Python programming language and a scraping technique. The YouTube data were retrieved after obtaining the API key from YouTube. The procedure began with the creation of a Google Developer Console account. Google's Google API client library and httplib are the libraries used to access the YouTube API. The installation of a library is accomplished with the following command:

```
pip install –upgrade google-api-python-client google-auth-httplib2 google-auth-oauthlib
```

It was then followed by invoking the library through syntax "from googleapiclient.discovery import build". There were 10,397 replies collected in all.

### *3.2. Data Labeling*

After gathering the feedback, the data were labeled. This study built machine learning via a supervised method. A supervised technique requires ground truth. All of these collected data were individually labeled by the Indonesian Literature Study Program personnel at Padjadjaran University, Bandung, Indonesia. Assigning a sentiment category to each piece of text in the dataset was the initial stage in this labeling technique. This manual method creates the ground truths for this study. Using this fact, machine learning analyzes data to generate models. The second stage was to transform any non-standard terminology into its corresponding synonyms. The labeling step categorized the gathered data into positive, negative, and neutral categories. There were a total of 1616 positive comments, 6640 negative comments, and 2141 neutral opinions. Negative comments contained the greatest info relative to other comments. The positive comments were the fewest in number. Table 1 below depicts the data proportion with labels. In the final step, retracing the labeling process was carried out to ensure inter-rater reliability. Table 2 shows an example of the resulting data from the labeling stage.

**Table 1.** Statistics of data.

| Label | Count |
|---|---|
| Positive | 1616 |
| Negative | 6640 |
| Neutral | 2141 |
| Total | 10,397 |

**Table 2.** Example of data.

| No. | Comment | Label |
|---|---|---|
| 1 | *Saya dukung lock down -7sebelum idul fitri +7 semoga berahir secepatnya corana di Indonesia* (I support the lockdown -7D before Eid Al-Fitr +7D, hopefully corona in Indonesia will end as soon as possible) | Positive |
| 2 | *Oke saya setuju di larang mudik tapi tolong kapal dan pesawat di berntikan juga beroprasi gimana oke* (Ok I agree mudik is prohibited, but please also stop the operation of ships and planes, OK). | Negative |
| 3 | *Saya ngak mudik pakde, saya cuman kangen pengen ketemu Orang tua dan saudara* (I don't go mudik, Sir, I only miss and want to see my parents and relatives) | Neutral |

### *3.3. Text Preprocessing*

Text preprocessing is the cleaning and categorizing of the text preparation. It is challenging due to the diverse nature of the gathered data, particularly the text data. In addition, certain words do not alter the main direction of sentences [60]. It is a critical step in sentiment analysis since it increases the quality of the categorization process. It eliminates noise from phrases [61]. There are several steps to preprocessing, including the following:

1. Case folding
   Case folding is the standardization of all text inside a phrase or document [62]. In this context, lowercase or non-capitalized letters are most frequently employed.
2. Character deletion
   The obtained data contain several features that are unnecessary for the analytical technique. Certain characters, including the uniform resource locator (URL), numbers, and punctuation, are eliminated.
3. Tokenizing
   Based on the units of analysis, this process breaks down texts or phrases into individual words. This method generates word vectors, often known as a bag of words (BOW) [63].
4. Stop-word removal
   This technique is used to eliminate frequent but insignificant terms from sentences [64]. It decreases the corpus size without compromising the key information contained.
5. Stemming
   It is the process of breaking down words into their simplest versions. This approach is used to reduce the complexity of the token's words [65]. During this phase, all affixes, whether prefixes, infixes, or suffixes, will be removed from the word.

This study employed a variety of experimental methods to discover the ideal preprocessing technique. The preprocessing steps used in this inquiry are shown in Figure 2. Trials in this study employed stemming and two distinct sets of stop-words. The stop-word removal technique keywords were extracted from a repository at the following address: https://github.com/datascienceid/stopwords-bahasa-indonesia (accessed on 18 February 2021). This study aimed to discover which of the two lists of stop-words would be more beneficial in this work. The first set of stop-words was taken directly from the aforementioned source. The second list, on the other hand, contained modified variants of already established stop-words. Numerous new stop words have been introduced, while negation terms have been eliminated. Eliminating negation words and introducing new stop words were among the modifications. Four distinct tests were undertaken in this work.

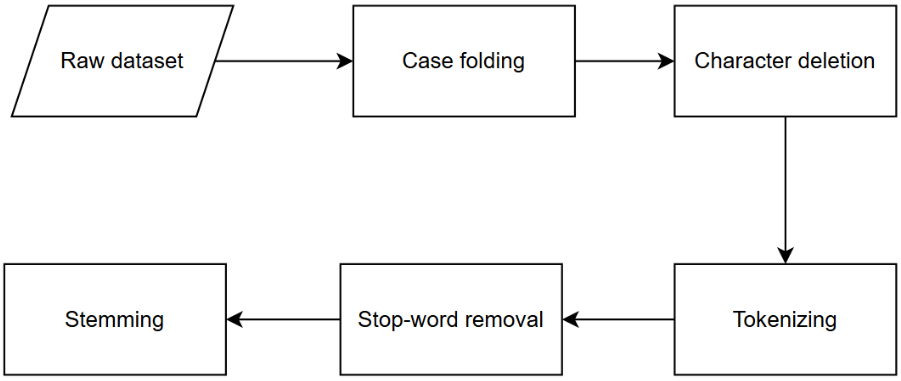


**Figure 2.** Preprocessing.

### *3.4. Modelling*

The BERT approach is based on transformers, which examine contextual relationships between words in a text [66]. Language translation techniques were first modeled after transformers. Transformers consist of two components: an encoder and a decoder. Each has a stack of six identical levels, while every stack has an identical architecture.

- Encoders
  The transformer component responsible for receiving instructions is the encoder. In an encoder layers, there are two sorts of sub-layers: self-attention and feed-forward neural networks. Self-attention can aid the encoder in paying attention to other words in the text input, in addition to focusing on the word being encoded directly. The output of the self-attention layer will be routed through a feed-forward neural network before reaching the subsequent layer.
- Decoders
  The decoder is a transformer component whose function is to create output predictions. Encoders and decoders are distinguished by the number of stacked layers. The decoder consists of three layers: self-attention, the feed-forward neural network, and attention. Self-attention and feed-forward neural networks fulfill a role analogous to that of encoders. In the meantime, the decoder benefits from the presence of the attention layer between the two.

The transformer theory is merely based on the notion of self-awareness. Self-attention enables the model to perceive the context, as opposed to only the overall word order. Using this information may determine the probable links between a given word and others in the language. The component called a feed-forward neural network (FFNN) generates predictions for the next word in a text [67]. Without consideration of the output, an FFNN will produce illogical data predictions. Randomly generated word predictions are contextually connected to the current phrase using a combination of FFNN and self-attention. The self-attention and FFNN outputs of both the encoder and the decoder are processed by the add & norm layer, which also contains the residual structure and layer normalization.

The encoder and decoder operations are described below [68].

1. The transformer will convert each entered word into a token representation. The tokenization of data will be adjusted to align with the language of the model. The embedding process is then utilized to convert each token into a vector sequence. Each of these vectors is comprised of 512 bytes. This operation is exclusive to the start of

the encoding phase, since the output of this initial encoder will function as the input for the next encoder.

2. Next, the vector series will enter the FFNN and self-attention phases. Throughout this operation, the input vector will be parsed into a query vector, a key vector, and a value vector. These three vectors are founded upon the multiplication of embeddings. In contrast to the encoder and decoder, which have 512 dimensions apiece, each of these three additional vectors has 64 dimensions.
3. Step two of self-awareness involves the making of a tally. This rating is determined by doing a dot product on the query and key vectors. Given that the square root of the dimension of each vector is eight, the total score will be divided by eight. After completion, the Softmax function will be utilized to calculate the final score. This function converts the score to a positive integer and the total to 1 when called. The result of the Softmax function will then be multiplied by the value vector. After combining the values in the vector, the outcome is the self-attention output. In the next phase of FFNN, this output will be utilized.
4. The next output will enter the decoder when all encoders have completed processing. In all practical ways, the decoder approach is similar to the encoding procedure. However, the decoder has a layer of attention between the self-attention and FFNN stages. A decoder search for significant words is facilitated by an attention layer. The decoder will produce a vector of floating-point values as its final output. The job of converting numbers into words requires the inclusion of two layers, specifically the final linear layer and the Softmax layer. Finally, we have a completely interconnected linear layer that yields vector logits—a vector collection of possible words. The Softmax layer is responsible for translating logit vectors into probabilities when everything else fails. The procedure's ultimate output will have the highest probability.

The model was developed utilizing deep learning techniques and draws influence from a variety of research models (such as ELMo, OpenAI GPT, and Transformer) [69]. The BERT model was founded based on the transformer language paradigm. Unlike traditional models that only learn in a single direction, this one is capable of self-improvement. The model can differentiate between right-to-left and left-to-right direction orders. Since it employs bidirectional learning, it can process text sequences from both left to right and right to left directions. In addition, this technique is a trained language model that simultaneously considers left and right word contexts. Several natural language processing (NLP) models, such as sentiment analysis, can benefit from BERT's capacity to improve their own findings. It is superior to split-training models because it can extract more context information [70]. It has been applied to a broad variety of NLP challenges in practice. BERT may perform tasks such as text encoding, similarity retrieval, summarization, question answering, and phrase categorization [66]. These multiple duties should be taught to the model with as few adjustments to the model as possible. Fine-tuning is the process of training a model to do this specific task.

The model is also built as a multilayered, bidirectional transformer, similar to an actual transformer. However, there is a minor discrepancy in that it solely utilizes the encoder component. Despite the absence of a decoder, it gets excellent performance, which enables the encoder to produce more stacks, more hidden sizes, and model training on enormous data sets, all of which contribute to this outcome. It may be built using either the BERTBASE or BERTLARGE major architectural models. BERTBASE is comprised of a total of 12 encoder layers, 12 self-attention heads, 768 hidden parameters, and 110 million parameters. However, BERTLARGE has 340 million parameters, 16 self-attention heads, a hidden size of 1024 bytes, and 24 encoder layers. Tokenization will be the initial stage in the model's text processing. The rules include introducing a special token at the beginning and end of each phrase, inserting a padding token to ensure that sentences are always the same length, and then employing an attention mask to differentiate between the padding token and the original token. Classifier tokens, also known as classifier token sets (CLS), are added at the beginning of a set to signal the classification phase [71]. Separation tokens, often known as [SEP]s, are utilized to indicate the conclusion of a sentence.

There are two steps to the BERT procedure: pre-training and tuning. During pre-training, which is conducted on a massive corpus of unlabeled text, the model will perform several tasks, including masked language modeling and sentence prediction. The fundamental objective of this basic training step is to develop a model capable of recognizing the context of a language. After receiving preliminary instruction, the model may be customized for specific occupations. Models are fine-tuned by executing them on training parameters and then adjusting them using labeled data [72]. The input representation is utilized to represent the input as a phrase on the token sequence, enabling it to perform a number of tasks. Each of these tokens will be represented with three distinct embeddings. There are three types of embedding: token embedding, segment embedding, and positional embedding. They provide the input representation for BERT. The model uses these three symbols in a variety of ways [73].

- Token embedding
  Initially, bypass the embedding token layer, which provides a vector representation of each input but is not utilized by later layers. The objective is to convert the input into tokens and assign distinct IDs to each token. After obtaining the ID, the succeeding token will be turned into a token representation. At this point, it will append a [CLS] token to each input and a [SEP] token to each output to serve as markers for the classification task's input representation, segmenting the provided text.
- Segment embedding
  The addition of this embedding component to the input token enables us to differentiate between sentences and establish their relative order. This distinction layer is needed due to the nature of BERT's input format. Only tokens from the first and second sentences are included in this layer.
- Positional embedding
  The positional embedding is the final phase in the tokenization process. Each input token receives an additional layer that may be used to determine its position in the sequence. Without this previous knowledge, it cannot grasp the meaning of a phrase's words. Thus, it will be able to differentiate between synonyms that appear in the same sentence.

### 3.4.1. Pre-Train

When it is being trained or pre-trained, it does two unsupervised tasks concurrently. Training is dependent upon two unique elements: the masked language model and next sentence prediction. During the learning phase, these two procedures are substituted for the decoder, because the BERT model only utilizes an encoder and no decoder [66].

- Masked language model (MLM)
  It is indisputable that the two-way representation paradigm is preferable to the one-way method. Current language models can only be trained using a single-way representation since two-way representations allow the model to re-read the same word. In order to perform two-way representation learning, the model will thus arbitrarily cover a tiny portion of the input, which refers to this educational approach. The encoder of the transformer does not know in advance which words to predict or replace, which is one of the primary advantages of using this model. This constrains the model to continue reflecting the distributional context on each input token. A small quantity of random word substitution has no impact on the comprehension phase of the model.
- Next sentence prediction
  The BERT model learning technique accepts sentence pairs as valid input. The aim of the model is to identify whether the second sentence is a continuation of the first. During training, two inputs will be utilized by the model. The initial 50% of the input consists of two phrases, the second of which is a continuation of the first. The remaining 50% is replaced with random sentences selected from the corpus.

### 3.4.2. Fine Tuning

Transfer learning, commonly referred to as fine-tuning, is the application of the information obtained from training one model to another. The new tuning approach is adaptable to the requirements of specific applications, such as text classification. BERT will utilize the [CLS] token set exclusively when performing classification tasks on texts [57]. Because a single token can represent the full phrase, the sentence similarity is the therefore the method through which it alters the [CLS] token's value based on the current dataset. The following actions are conducted during the tuning phase:

- When a user enters text, it is tokenized beforehand. According to the BERT dictionary, the input text is tokenized and augmented with specialized tokens. If the original word does not include tokens, BERT will break it down into smaller words that do.
- The token is translated into an embedding format following tokenization.
- Throughout the learning phase, the encoder applies the transformers' steps.
- In the final layer, only the [CLS] token's output is used for classification.

### 3.4.3. BERT's Hyperparameter

The choice of hyperparameter has such a dramatic impact on the model's output that the entire machine-learning procedure is extremely sensitive to it. Therefore, analysis is needed to determine the optimal hyperparameter, which is as follows:

- Batch size
  This determines how many samples must be processed before the system's default parameters may be modified. The batch size is the mechanism through which information is delivered uniformly across the whole model. A total of 100 iterations would be necessary, for instance, to uniformly disperse 3000 data throughout a model with a batch size of 30. When the batch size is bigger, the reliability of the results increases [35].
- Epoch
  The iteration count of an algorithm over the dataset of the research is controlled by the hyperparameter epoch, which denotes an epoch of time. Each training data sample corresponds to one epoch, during which the model parameters are modified depending on the newly acquired information. By dividing the entire quantity of data by the batch size, the number of iterations in a single epoch may be determined. If there are 3000 observations and the batch size is set to 30, then 100 iterations will be performed in a single epoch.
- Learning rate
  The value of this learning rate often falls within the range of 0 and 1. The model's learning rate must strike a balance between being too high and too low because it will have an effect on the model's efficiency. Training time can be reduced if there is a higher learning rate. This is due to the decreased accuracy experienced during training.

This study employed both the IndoBERT pre-train and BERTBASE post-train models. As IndoBERT is a natural language processing model based on Indonesian, it was therefore used in this study. A dataset consisting of approximately four million Indonesian words was utilized to train the IndoBERT pre-train model. A transformers library was required to perform research with IndoBERT as a pre-train model. Huggingface's Transformers Python package includes a vast quantity of BERT-ready models. The data were split after the word embedding procedure was complete. Meanwhile, the model was trained, overfitting was minimized through validation, and the trained model's performance was assessed through testing. This study utilized a total of twelve encoders for BERTBASE. On each of these encoders, a self-attention mechanism and a feed-forward neural network were implemented. The input was transformed in the first encoder using a word embedding procedure that was distinct from that of the other encoders. The BERT tokenizer carries out the word embedding stage. Figure 3 depicts the BERT word embedding process flowchart. BERT performed classification using a fully connected layer coupled with a

Softmax function. In its own encoder, BERT only utilized the vector output of the [CLS] token. During this step of classification, BERT modified the dataset's weights based on the degree of similarity between the most recent [CLS] token and the existing tokens. This is what is referred to as sentence similarity. The logit score was determined by evaluating two sentences. These logs were processed using Softmax to yield probabilities, whose total was 1. As the output, the probability value with the greatest value will be chosen. It was used to generate probabilities from these logits vectors. This was the training phase of the model, and Adam was used to alter the emphasis of the input sentence. This study proposed the following value to achieve the optimal hyperparameters:

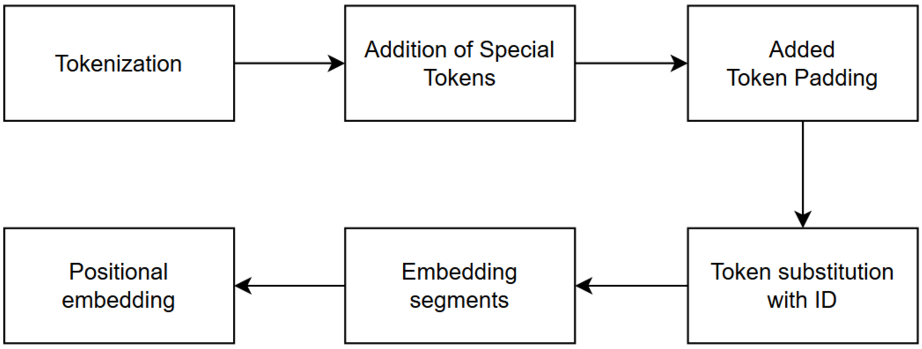


**Figure 3.** Word embedding.

- Batch size: 16, 32
- Epoch: 2, 3, 4
- Learning rate: 2e-5, 3e-5, 5e-5

### *3.5. Evaluation*

Confusion matrix evaluation was used for this investigation as a machine learning assessment tool. It was utilized to construct performance metrics evaluating the model's quality. Examples of performance measurements include precision, recall, and F-score [74].

- Precision is the degree to which observed data correspond to those predicted by a model.
- Recall is the model's ability to locate previously stored data. It is the fraction of true positive predictions relative to all true positive data.
- F-score, often known as the f-measure, is the average ratio of a measure's recall to its precision.

The assessment process began with actions taken to find the optimal model's hyperparameters in advance. Next, this study selected the best model based on precision, recall, and F-score. The higher the value of precision, recall, and F-score means the better the model. Therefore, a model is deemed superior if these three parameter values are greater than all others.

## **4. Results and Discussion**

At this stage of testing, four hyperparameters of the sentiment analysis model were assessed. This examination was divided into two sections: the preprocessing method and the BERT hyperparameter examination. This study determined which preprocessing strategy might be the most successful by administering several tests. Regarding the BERT hyperparameter, three factors were evaluated: the batch size, epoch, and learning rate.

### *4.1. Preprocessing Data Analysis*

This study performed four distinct test combinations to identify the ideal preprocessing strategy. This number was derived by first employing the two stop-word lists used in the study, and then determining whether or not to utilize a stemming phase. Note that the amended version of the first list of stop words acts as the second list. The outcomes of the investigation that verified the preprocessing approach are presented in Table 3. The greatest results were obtained when the second list of stop-words was used without stemming during preprocessing. It indicated that the BERT approach for sentiment analysis might be impacted by the inclusion or exclusion of certain phrases. When stemming was deactivated, both sets of results increased because the affix affected the BERT token. During the embedding process, the ID of the BERT token within the embedded word might be adjusted by stemming. In addition, there were infix words, repetition words, and words that melt upon meeting affixes that had not been adequately addressed in the employed technique, rendering stemming extremely hazardous when used on Indonesian words for this work.

**Table 3.** Study of Data Preprocessing.

| Preprocessing | | Precision | Recall | F-Score |
|---|---|---|---|---|
| Stopword List | Stemming | | | |
| 1 | Yes | 80.67% | 78.33% | 79.67% |
| 1 | No | 80.33% | 78.00% | 83.33% |
| 2 | Yes | 80.00% | 80.33% | 81.67% |
| 2 | No | 83.67% | 85.00% | 84.33% |

### *4.2. Analysis of Batch Size*

In this study, determining the ideal batch size required two distinct figures. It employed a predefined set of hyperparameter values, including a learning rate of 2e-5 and a set of three epochs. The information presented in Table 4 suggests that 32 was the optimal number of batch sizes. Increasing the batch size would render the model more resilient. An epoch delivered a more even distribution of data. Therefore, 32 batch sizes were superior to 16 batch sizes.

**Table 4.** Batch Size Test Results.

| Batch Size | Precision | Recall | F-Score |
|---|---|---|---|
| 16 | 81.00% | 81.33% | 81.33% |
| 32 | 83.67% | 85.00% | 84.33% |

### *4.3. Analysis of Learning Rate*

Table 5 demonstrates that the best learning rate was 2e-5. Using a lesser learning value resulted in a superior and more stable model. Due to the progressive nature of weight-change methods, a smaller number would produce a more accurate amount of model adjustment. Consequently, the model training took more time.

**Table 5.** Learning Rate Test Results.

| Learning Rate | Batch Size | Precision | Recall | F-Score |
|---|---|---|---|---|
| 2e-5 | 32 | 83.67% | 85.00% | 84.33% |
| 3e-5 | 32 | 81.00% | 80.33% | 80.67% |
| 5e-5 | 32 | 78.67% | 77.00% | 77.67% |

### *4.4. Analysis of Epoch*

This study employed four distinct values to estimate the optimal epoch. This analysis implied that three epochs were the ideal duration for this investigation, as shown in Table 6. In the BERT framework, a low epoch value did not always result in mediocre model training. Due to the fact that the BERT model was a pre-train model, the training process was primarily focused on the stage of fine-tuning. Therefore, the BERT model might be trained with reduced epoch values.

**Table 6.** Epoch Tests.

| Epoch | Learning Rate | Batch Size | Precision | Recall | F-Score |
|---|---|---|---|---|---|
| 2 | 2.00E-05 | 32 | 79.33% | 82.00% | 82.33% |
| 3 | 2.00E-05 | 32 | 83.67% | 85.00% | 84.33% |
| 4 | 2.00E-05 | 32 | 83.33% | 83.67% | 83.67% |
| 10 | 2.00E-05 | 32 | 84.00% | 82.33% | 83.00% |

### *4.5. Final Evaluation*

Extensive testing demonstrated that the greatest results could be achieved with the second stop-word list without stemming, a batch size of 32, a learning rate of 2e-5, and a total of three epochs. When all conditions were optimal, the precision was 83.67%, the recall was 85%, and the f-score was 84.33%. Table 7 depicts the resulted confusion matrix model. There were 284 pieces of negative news that were properly predicted to be negative, 25 pieces of negative news that were accurately predicted to be neutral, and seven pieces of negative news that were accurately predicted to be positive. Furthermore, there were 20 predictions for negative neutral data, 68 for neutral data, and 7 for positive neutral data. In the positive row, seven positive data were projected to be negative, four were predicted to be neutral, and the remaining 78 were predicted to be positive. The outcomes of our computations of precision, recall, and F-score based on the confusion matrix are displayed in Table 8.

**Table 7.** Confusion Matrix.

| Actual Class | Prediction | | |
|---|---|---|---|
| | Negative | Neutral | Positive |
| Negative | 284 | 25 | 7 |
| Neutral | 20 | 68 | 7 |
| Positive | 7 | 4 | 78 |

**Table 8.** Evaluation Results.

| | Precision | Recall | F-Score |
|---|---|---|---|
| Negative | 91% | 90% | 91% |
| Neutral | 75% | 77% | 76% |
| Positive | 85% | 88% | 86% |

The obtained precision and recall percentages of negative sentiment were 91% and 90%, respectively. Consequently, when the F-score was computed from the two variables, a result of 91% for the negative sentiment was produced. The resulting precision value for the neutral sentiment, on the other hand, was 75%. It achieved a 77% recall rate for a neutral sentiment using this strategy. The sum of these two scores was 76%, showing an F-score of this opinion. The calculated precision for positive sentiment was 85%. Consequently, 88% of individuals recall having a positive view. This indicates that an F-score for optimism of 86% might be obtained using these two figures. The negative sentiment F-score was 91%, which was bigger than the neutral and positive sentiment F-scores. It can be concluded that predictive data models were more successful when dealing with negative sentiment when put into practice.

### *4.6. Comparison to Other Algorithms*

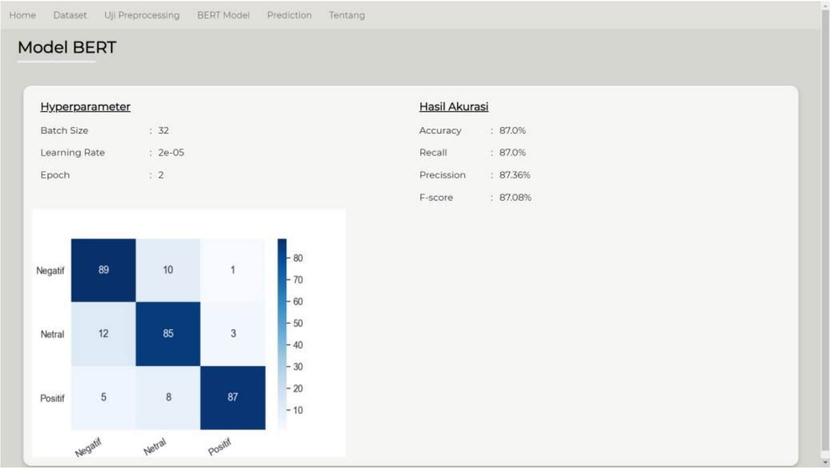
Table 9 displays the results of comparing the BERT model to the naive Bayes, SVM, and LSTM. According to the study's test data, the BERT algorithm produced the best model when compared to the other three algorithms. It demonstrates that, for the considered data set, the BERT algorithm surpassed its competitors in terms of sentiment analysis. The power of the BERT algorithm could not be separated from its two-way learning implementation, which enabled the extraction of a bigger number of context features.

**Table 9.** Comparison Results.

| No. | Algorithms | Precision | Recall | F-Score |
|---|---|---|---|---|
| 1 | BERT | 83.67% | 85.00% | 84.33% |
| 2 | Naive Bayes | 74.47% | 76.22% | 74.10% |
| 3 | SVM | 81.00% | 81.00% | 81.00% |
| 4 | LSTM | 81.27% | 82.74% | 82.00% |

### *4.7. Application Implementation*

The web-based application that contains the sentiment analysis system was developed using the flask web framework in Python. Users can input hyperparameter values that will be used for training BERT models provided by the program. In addition to the confusion matrix and numbers for accuracy, precision, recall, and f-score displayed in Figure 4, the viewer may examine the BERT model assessment results. In addition, Figure 5 displays the interface for user prediction. Users can submit a sentiment statement and the program will predict its label and display the results to the user. The application uses the Indonesian language to make it easier for Indonesian people to use it.



Model BERT

Hyperparameter

Batch Size : 32

Learning Rate : 2e-05

Epoch : 2

Hasil Akurasi

Accuracy : 87.0%

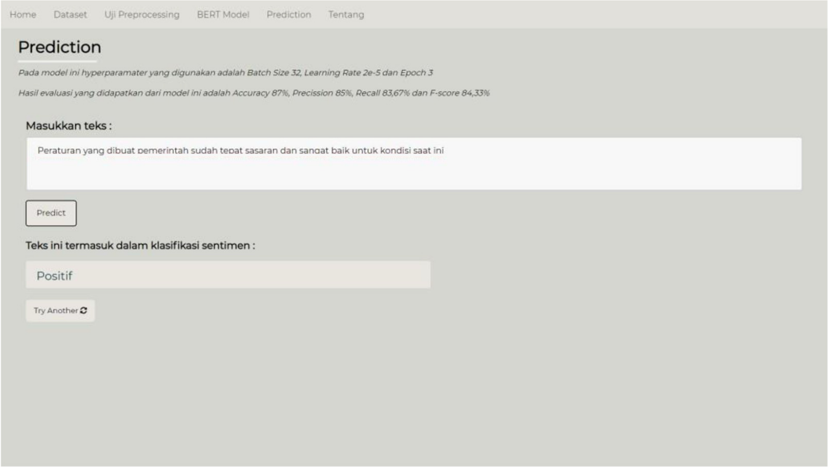
Recall : 87.0%

Precission : 87.36%

F-score : 87.08%

| | Negatif | Netral | Positif |
|---|---|---|---|
| Negatif | 89 | 10 | 1 |
| Netral | 12 | 85 | 3 |
| Positif | 5 | 8 | 87 |

**Figure 4.** Implementation of the BERT model in applications, which is presented in Indonesian here. When the model is executed, it comprises hyperparameter information (**top left**), accuracy results (**top right**), and a confusion matrix (**bottom left**).



Prediction

*Pada model ini hyperparamater yang digunakan adalah Batch Size 32, Learning Rate 2e-5 dan Epoch 3*

*Hasil evaluasi yang didapatkan dari model ini adalah Accuracy 87%, Precission 85%, Recall 83,67% dan F-score 84,33%*

Masukkan teks :

Peraturan yang dibuat pemerintah sudah tepat sasaran dan sangat baik untuk kondisi saat ini

Teks ini termasuk dalam klasifikasi sentimen :

Positif

**Figure 5.** Implementation of predictions in the application. This display in Indonesian requests that the user enter an input whose sentiment will be predicted. When the "predict" button is pressed, the system anticipates the input. The projected outcome will be displayed below.

## 5. Conclusions

Using the BERT approach, this study investigates the Indonesian people's stance on the government policy on vacations during the COVID-19 pandemic. This study created a new dataset for this topic in which the information source was obtained from news outlet stories and the comment sections of YouTube videos. The method was designed to classify public sentiment as positive, neutral, or negative. It used an epoch size of 3, a learning rate of 2e-5, and a batch size of 32 for optimal results. Using Python and the Flask framework, an application for sentiment analysis was then constructed to apply the optimal BERT model. Based on the results of testing and applying the model, the obtained F-score was 84.33%. This study is expected to become one of the tools that can be used by the Indonesian people, especially the government, to see the extent of public acceptance of government policies that have been made, especially regarding vacations. This study demonstrates that the proposed strategy is effective due to its success in classifying emotions. In addition, however, the data acquired in this study can serve as the basis for sentiment analysis research in Indonesia. Sentiment analysis is based on language. The success of technology in a particular language cannot be immediately extended to other languages. Therefore, despite the rapid development of sentiment analysis in English, this cannot be directly applied to sentiment research in Indonesian. The existence of ground truth as a result of this research would enrich the Indonesian language's collection of sentiments. Another thing that may be achieved is the correct method of processing Indonesian, which has its own distinct characteristics. However, the current ground truth has limits because it does not accept slang or regional languages, which frequently appear in YouTube comments. Therefore, sentiment analysis can be used as an effective tool when the input is supplied in formal language.

**Author Contributions:** Conceptualization, I.N.Y.; methodology, I.N.Y., V.W. and R.R.; software, V.W.; validation, I.N.Y., I.S. and Y.D.; formal analysis, I.N.Y.; investigation, I.N.Y. and V.W.; resources, V.W. and Y.D.; data curation, I.N.Y. and R.R.; writing—original draft preparation, I.N.Y., V.W. and A.S.P.; writing—review and editing, I.N.Y.; visualization, V.W.; supervision, A.S.P.; project administration, I.N.Y.; funding acquisition, I.N.Y. All authors have read and agreed to the published version of the manuscript.

**Funding:** This study was funded by the Library and Online Data Research Grant, Universitas Padjadjaran, No. Contract: 1959/UN6.3.1/PT.00/2021.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** The authors warmly appreciate the Rector, Directorate of Research and Community Service (DRPM), and Research Center for Artificial Intelligence and Big Data at Universitas Padjadjaran. We also thank the world-class professor 2021 program, Ministry of Education and Culture, Indonesia.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Setiati, S.; Azwar, M.K. COVID-19 and Indonesia. *Acta Med. Indones.* **2020**, *52*, 84–89. [PubMed]
2. Sreeramula, S.; Rahardjo, D. Estimating COVID-19 Rt in Real-time: An Indonesia health policy perspective. *Mach. Learn. Appl.* **2021**, *6*, 100136. [CrossRef] [PubMed]
3. Zainuddin, Z. The 2021 Lebaran Homecoming Prohibition Policy by the Government of Indonesia in A State Administrative Law Perspectivea. *Randwick Int. Soc. Sci. J.* **2021**, *2*, 214–224. [CrossRef]
4. Harto, B. Tinjauan Surat Edaran Satgas covid-19 Nomor 20 Tahun 2021 Atas Kebijakan Wajib Karantina Setelah Melakukan Perjalanan Internasional Pada Masa Pandemi covid-19 Dalam Perspektif Hak Asasi Manusia. *J. Smart Huk.* **2022**, *1*, 208–216.
5. Utomo, P. covid-19 versus Mudik Telaah Tentang Efektivitas Kebijakan Pelarangan Mudik Lebaran Pada Masa Pandemi Covid-19. *QISTIE* **2021**, *14*, 111–125. [CrossRef]

6. Putra, S.J.; Aziz, M.A.; Gunawan, M.N. Topic Analysis of Indonesian Comment Text Using the Latent Dirichlet Alloca-tion. In Proceedings of the 9th International Conference on Cyber and IT Service Management (CITSM), Bengkulu, Indonesia, 22–23 September 2021; pp. 1–6.
7. Alamoodi, A.H.; Zaidan, B.B.; Zaidan, A.A.; Albahri, O.S.; Mohammed, K.I.; Malik, R.Q.; Alaa, M. Sentiment analysis and its applications in fighting COVID-19 and infectious diseases: A systematic review. *Expert Syst. Appl.* **2021**, *167*, 114155. [CrossRef]
8. Alsayat, A.; Elmitwally, N. A comprehensive study for Arabic sentiment analysis (challenges and applications). *Egypt. Inform. J.* **2020**, *21*, 7–12. [CrossRef]
9. Barbounaki, S.G.; Gourounti, K.; Sarantaki, A. Advances of Sentiment Analysis Applications in Obstetrics/Gynecology and Midwifery. *Mater. Socio. Med.* **2021**, *33*, 225. [CrossRef]
10. Kastrati, Z.; Dalipi, F.; Imran, A.S.; Pireva Nuci, K.; Wani, M.A. Sentiment analysis of students' feedback with NLP and deep learning: A systematic mapping study. *Appl. Sci.* **2021**, *11*, 3986. [CrossRef]
11. Wankhade, M.; Rao, A.C.S.; Kulkarni, C. A survey on sentiment analysis methods, applications, and challenges. *Artif. Intell. Rev.* **2022**, *55*, 5731–5780. [CrossRef]
12. Zucco, C.; Calabrese, B.; Agapito, G.; Guzzi, P.H.; Cannataro, M. Sentiment analysis for mining texts and social networks data: Methods and tools. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2020**, *10*, e1333. [CrossRef]
13. Alwadi, A.; Lathifa, Z. Applications of Artificial Intelligence in the Treatment of Behavioral and Mental Health Conditions. *Appl. Res. Artif. Intell. Cloud Comput.* **2022**, *5*, 1–18.
14. Abdi, A.; Shamsuddin, S.M.; Hasan, S.; Piran, J. Deep learning-based sentiment classification of evaluative text based on multi-feature fusion. *Inf. Process. Manag.* **2019**, *56*, 1245–1259. [CrossRef]
15. Zhang, L.; Wang, S.; Liu, B. Deep learning for sentiment analysis: A survey. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2018**, *8*, e1253. [CrossRef]
16. Dang, N.C.; Moreno-García, M.N.; De la Prieta, F. Sentiment analysis based on deep learning: A comparative study. *Electronics* **2020**, *9*, 483. [CrossRef]
17. Ruz, G.A.; Henríquez, P.A.; Mascareño, A. Sentiment analysis of Twitter data during critical events through Bayesian networks classifiers. *Future Gener. Comput. Syst.* **2020**, *106*, 92–104. [CrossRef]
18. Salah, Z.; Al-Ghuwairi, A.R.F.; Baarah, A.; Aloqaily, A.; Qadoumi, B.A.; Alhayek, M.; Alhijawi, B. A systematic review on opinion mining and sentiment analysis in social media. *Int. J. Bus. Inf. Syst.* **2019**, *31*, 530–554. [CrossRef]
19. Jia, Q.; Guo, Y.; Wang, G.; Barnes, S.J. Big data analytics in the fight against major public health incidents (Including COVID-19): A conceptual framework. *Int. J. Environ. Res. Public Health* **2020**, *17*, 6161. [CrossRef]
20. Djalante, R.; Nurhidayah, L.; Van Minh, H.; Phuong, N.T.N.; Mahendradhata, Y.; Trias, A.; Miller, M.A. COVID-19 and ASEAN responses: Comparative policy analysis. *Prog. Disaster Sci.* **2020**, *8*, 100129. [CrossRef]
21. Le, T.H.; Chen, H.; Babar, M.A. Deep learning for source code modeling and generation: Models, applications, and challenges. *ACM Comput. Surv.* **2020**, *53*, 1–38. [CrossRef]
22. Mridha, M.F.; Keya, A.J.; Hamid, M.A.; Monowar, M.M.; Rahman, M.S. A Comprehensive Review on Fake News Detection with Deep Learning. *IEEE Access.* **2021**, *9*, 156151–156170. [CrossRef]
23. Deepa, M.D. Bidirectional Encoder Representations from Transformers (BERT) Language Model for Sentiment Analysis task. *Turk. J. Comput. Math. Educ.* **2021**, *12*, 1708–1721.
24. Oueslati, O.; Cambria, E.; Haj Hmida, M.B.; Ounelli, H. A review of sentiment analysis research in Arabic language. *Future Gener. Comput. Syst.* **2020**, *112*, 408–430. [CrossRef]
25. Zabha, N.I.; Ayop, Z.; Anawar, S.; Hamid, E.; Abidin, Z.Z. Developing cross-lingual sentiment analysis of Malay Twitter data using lexicon-based approach. *Int. J. Adv. Comput. Sci. Appl.* **2019**, *10*, e0100146. [CrossRef]
26. Garcia, K.; Berton, L. Topic detection and sentiment analysis in Twitter content related to COVID-19 from Brazil and the USA. *Appl. Soft Comput.* **2021**, *101*, 107057. [CrossRef]
27. Dashtipour, K.; Gogate, M.; Adeel, A.; Larijani, H.; Hussain, A. Sentiment analysis of persian movie reviews using deep learning. *Entropy* **2021**, *23*, 596. [CrossRef]
28. Guhr, O.; Schumann, A.K.; Bahrmann, F.; Böhme, H.J. Training a broad-coverage german sentiment classification model for dialog systems. In Proceedings of the Twelfth Language Resources and Evaluation Conference, Marseille, France, 11–16 May 2020; pp. 1627–1632.
29. Pereira, D.A. A survey of sentiment analysis in the Portuguese language. *Artif. Intell. Rev.* **2021**, *54*, 1087–1115. [CrossRef]
30. Wu, J.; Lu, K.; Su, S.; Wang, S. Chinese micro-blog sentiment analysis based on multiple sentiment dictionaries and semantic rule sets. *IEEE Access.* **2019**, *7*, 183924–183939. [CrossRef]
31. Gan, C.; Feng, Q.; Zhang, Z. Scalable multi-channel dilated CNN–BiLSTM model with attention mechanism for Chinese textual sentiment analysis. *Future Gener. Comput. Syst.* **2021**, *118*, 297–309. [CrossRef]
32. Khan, I.U.; Khan, A.; Khan, W.; Su'ud, M.M.; Alam, M.M.; Subhan, F.; Asghar, M.Z. A review of Urdu sentiment analysis with multilingual perspective: A case of Urdu and roman Urdu language. *Computers* **2022**, *11*, 3. [CrossRef]
33. Khan, M.R.H.; Afroz, U.S.; Masum, A.K.M.; Abujar, S.; Hossain, S.A. Sentiment analysis from bengali depression dataset using machine learning. In Proceedings of the 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 1–3 July 2020; pp. 1–5.

34. Ngoc, D.N.; Thi, T.P.; Do, P. Preprocessing improves CNN and LSTM in aspect-based sentiment analysis for Vietnamese. In Proceedings of the Fifth International Congress on Information and Communication Technology: ICICT, London, UK, 20–21 February 2020; Volume 1, pp. 175–185.
35. Afifah, K.; Yulita, I.N.; Sarathan, I. Sentiment Analysis on Telemedicine App Reviews using XGBoost Classifier. In Proceedings of the International Conference on Artificial Intelligence and Big Data Analytics, Bandung, Indonesia, 27–29 October 2021; pp. 22–27.
36. Štrimaitis, R.; Stefanovič, P.; Ramanauskaitė, S.; Slotkienė, A. Financial context news sentiment analysis for the Lithuanian language. *Appl. Sci.* **2021**, *11*, 4443. [CrossRef]
37. Xu, Y.; Cao, H.; Du, W.; Wang, W. A survey of cross-lingual sentiment analysis: Methodologies, models and evaluations. *Data Sci. Eng.* **2022**, *7*, 279–299. [CrossRef]
38. Pradha, S.; Halgamuge, M.N.; Vinh, N.T.Q. Effective text data preprocessing technique for sentiment analysis in social media data. In Proceedings of the 11th International Conference on Knowledge And Systems Engineering (KSE), Da Nang, Vietnam, 24–26 October 2019; pp. 1–8.
39. de Oliveira, D.N.; Merschmann, L.H.D.C. Joint evaluation of preprocessing tasks with classifiers for sentiment analysis in Brazilian Portuguese language. *Multimed. Tools Appl.* **2021**, *80*, 15391–15412. [CrossRef]
40. Sohrabi, M.K.; Hemmatian, F. An efficient preprocessing method for supervised sentiment analysis by converting sentences to numerical vectors: A twitter case study. *Multimed. Tools Appl.* **2019**, *78*, 24863–24882. [CrossRef]
41. Alam, S.; Yao, N. The impact of preprocessing steps on the accuracy of machine learning algorithms in sentiment analysis. *Comput. Math. Organ. Theory* **2019**, *25*, 319–335. [CrossRef]
42. Resyanto, F.; Sibaroni, Y.; Romadhony, A. Choosing the most optimum text preprocessing method for sentiment analysis: Case: iPhone Tweets. In Proceedings of the Fourth International Conference on Informatics and Computing (ICIC), Semarang, Indonesia, 16–17 October 2019; pp. 1–5.
43. Villavicencio, C.; Macrohon, J.J.; Inbaraj, X.A.; Jeng, J.H.; Hsieh, J.G. Twitter sentiment analysis towards covid-19 vaccines in the Philippines using naïve bayes. *Information* **2021**, *12*, 204. [CrossRef]
44. Li, Z.; Li, R.; Jin, G. Sentiment analysis of danmaku videos based on naïve bayes and sentiment dictionary. *IEEE Access* **2020**, *8*, 75073–75084. [CrossRef]
45. Prastyo, P.H.; Sumi, A.S.; Dian, A.W.; Permanasari, A.E. Tweets responding to the Indonesian Government's handling of COVID-19: Sentiment analysis using SVM with normalized poly kernel. *J. Inf. Syst. Eng. Bus. Intell.* **2020**, *6*, 112. [CrossRef]
46. Singh, J.; Tripathi, P. Sentiment analysis of Twitter data by making use of SVM, Random Forest and Decision Tree algorithm. In Proceedings of the 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT), Bhopal, India, 18–19 June 2021; pp. 193–198.
47. Fitri, V.A.; Andreswari, R.; Hasibuan, M.A. Sentiment analysis of social media Twitter with case of Anti-LGBT campaign in Indonesia using Naïve Bayes, decision tree, and random forest algorithm. *Procedia Comput. Sci.* **2019**, *161*, 765–772. [CrossRef]
48. Saad, S.E.; Yang, J. Twitter sentiment analysis based on ordinal regression. *IEEE Access.* **2019**, *7*, 163677–163685. [CrossRef]
49. Yadav, A.; Vishwakarma, D.K. Sentiment analysis using deep learning architectures: A review. *Artif. Intell. Rev.* **2020**, *53*, 4335–4385. [CrossRef]
50. Kaur, H.; Ahsaan, S.U.; Alankar, B.; Chang, V. A proposed sentiment analysis deep learning algorithm for analyzing COVID-19 tweets. *Inf. Syst. Front.* **2021**, *23*, 1417–1429. [CrossRef] [PubMed]
51. Zhang, Y.; Tiwari, P.; Song, D.; Mao, X.; Wang, P.; Li, X.; Pandey, H.M. Learning interaction dynamics with an interactive LSTM for conversational sentiment analysis. *Neural Netw.* **2021**, *133*, 40–56. [CrossRef] [PubMed]
52. Behera, R.K.; Jena, M.; Rath, S.K.; Misra, S. Co-LSTM: Convolutional LSTM model for sentiment analysis in social big data. *Inf. Process. Manag.* **2021**, *58*, 102435. [CrossRef]
53. Wang, J.; Yu, L.C.; Lai, K.R.; Zhang, X. Tree-structured regional CNN-LSTM model for dimensional sentiment analysis. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2019**, *28*, 581–591. [CrossRef]
54. Huang, F.; Li, X.; Yuan, C.; Zhang, S.; Zhang, J.; Qiao, S. Attention-emotion-enhanced convolutional LSTM for sentiment analysis. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 4332–4345. [CrossRef]
55. Ombabi, A.H.; Ouarda, W.; Alimi, A.M. Deep learning CNN–LSTM framework for Arabic sentiment analysis using textual information shared in social networks. *Soc. Netw. Anal. Mining.* **2020**, *10*, 53. [CrossRef]
56. Pei, Y.; Chen, S.; Ke, Z.; Silamu, W.; Guo, Q. Ab-labse: Uyghur sentiment analysis via the pre-training model with bilstm. *Appl. Sci.* **2022**, *12*, 1182. [CrossRef]
57. Azzouza, N.; Akli-Astouati, K.; Ibrahim, R. Twitterbert: Framework for twitter sentiment analysis based on pre-trained language model representations. In Proceedings of the Emerging Trends in Intelligent Computing and Informatics: Data Science, Intelligent Information Systems and Smart Computing 4, Johor, Malaysia, 22–23 September 2020; pp. 428–437.
58. Pathak, A.; Kumar, S.; Roy, P.P.; Kim, B.G. Aspect-based sentiment analysis in Hindi language by ensembling pre-trained mBERT models. *Electronics* **2021**, *10*, 2641. [CrossRef]
59. He, J.; Hu, H. MF-BERT: Multimodal fusion in pre-trained BERT for sentiment analysis. *IEEE Signal Process. Lett.* **2021**, *29*, 454–458. [CrossRef]
60. Hickman, L.; Thapa, S.; Tay, L.; Cao, M.; Srinivasan, P. Text preprocessing for text mining in organizational research: Review and recommendations. *Organ. Res. Method* **2022**, *25*, 114–146. [CrossRef]

61. Kustanto, N.S.; Yulita, I.N.; Sarathan, I. Sentiment Analysis of Indonesia's National Health Insurance Mobile Application using Naïve Bayes Algorithm. In Proceedings of the International Conference on Artificial Intelligence and Big Data Analytics, Bandung, Indonesia, 27–29 October 2021; pp. 38–42.
62. Hasanah, U.; Astuti, T.; Wahyudi, R.; Rifai, Z.; Pambudi, R.A. An experimental study of text preprocessing techniques for automatic short answer grading in Indonesian. In Proceedings of the International Conference on Information Technology, Information System and Electrical Engineering, Yogyakarta, Indonesia, 13–14 November 2018; pp. 230–234.
63. Cheligeer, C.; Huang, J.; Wu, G.; Bhuiyan, N.; Xu, Y.; Zeng, Y. Machine learning in requirements elicitation: A literature review. *AI EDAM* **2022**, *36*, e32. [CrossRef]
64. Birjali, M.; Kasri, M.; Beni-Hssane, A. A comprehensive survey on sentiment analysis: Approaches, challenges and trends. *Knowl. Syst.* **2021**, *226*, 107134. [CrossRef]
65. Asgari-Chenaghlu, M.; Feizi-Derakhshi, M.R.; Farzinvash, L.; Balafar, M.A.; Motamed, C. Topic detection and tracking techniques on Twitter: A systematic review. *Complexity* **2021**, *2021*, 8833084. [CrossRef]
66. Tay, Y.; Dehghani, M.; Bahri, D.; Metzler, D. Efficient transformers: A survey. *ACM Comput. Surv.* **2022**, *55*, 1–28. [CrossRef]
67. Jia, J.; Chen, X.; Yang, A.; He, Q.; Dai, P.; Liu, M. Link of Transformers in CV and NLP: A Brief Survey. In Proceedings of the 5th International Conference on Pattern Recognition and Artificial Intelligence (PRAI), Chengdu, China, 19–21 August 2022; pp. 735–743.
68. Nassiri, K.; Akhloufi, M. Transformer models used for text-based question answering systems. *Appl. Intell.* **2022**. [CrossRef]
69. Mars, M. From Word Embeddings to Pre-Trained Language Models: A State-of-the-Art Walkthrough. *Appl. Sci.* **2022**, *12*, 8805. [CrossRef]
70. Shah Jahan, M.; Khan, H.U.; Akbar, S.; Umar Farooq, M.; Gul, S.; Amjad, A. Bidirectional Language Modeling: A Systematic Literature Review. *Sci. Program.* **2021**. [CrossRef]
71. Acheampong, F.A.; Nunoo-Mensah, H.; Chen, W. Transformer models for text-based emotion detection: A review of BERT-based approaches. *Artif. Intell. Rev.* **2021**, *54*, 5789–5829. [CrossRef]
72. Shorten, C.; Khoshgoftaar, T.M.; Furht, B. Text data augmentation for deep learning. *J. Big Data* **2021**, *8*, 101. [CrossRef]
73. Chahal, K.S.; Grover, M.S.; Dey, K.; Shah, R.R. A hitchhiker's guide on distributed training of deep neural networks. *J. Parallel Distrib. Comput.* **2020**, *137*, 65–76. [CrossRef]
74. Maxwell, A.E.; Warner, T.A.; Guillén, L.A. Accuracy assessment in convolutional neural network-based deep learning remote sensing studies—Part 1: Literature review. *Remote Sens.* **2021**, *13*, 2450. [CrossRef]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.