IJITEB
Vol. 6
No. 2
2024

International Journal of Information Technology and Business

http://ejournal.uksw.edu/ijiteb



Twitter Sentiment Analysis Using Natural Language Processing (NLP) Method and Long Short Term Memory (LSTM) Algorithm in the 2024 Indonesian Presidential Election

Basworo Ardi Pramono[1], April Firman Daru[2,*], Muhammad Bahrul Ulum[3]

[1-3]Faculty of Information technology and Communication, Semarang University, Indonesia

Abstract : Twitter is one of the media used by the Indonesian people to express their opinions regarding the 2024 Presidential Election. However, there is no scientific calculation that can determine the tone of public opinion regarding the 2024 presidential election. In this study, sentiment analysis was carried out on the tweets of the Indonesian people related to the 2024 Presidential Election (Pilpres 2024). The purpose of this study is to find out the opinions of Indonesian Twitter users regarding the 2024 Presidential Election using Natural Language Processing (NLP) Technology and Long Short Term Memory (LSTM) algorithms. NLP techniques are used to understand natural language and extract meaning from tweet copy, and LSTM is used to analyze the accuracy and accuracy of classification. The data used in this study was 1,004 tweets with the topic "Presidential Election", this data researchers obtained through the process of crawling using the tweet harvest library. In this study, 53.2% had positive emotions, 3.5% had neutral emotions, and 43.3% had negative emotions. 78% accuracy, 67% precision, and 67% recall.

Keywords: LSTM, NLP, Pilpres, Twitter

1. Heading

Sentiment analysis on Twitter has recently sparked scholarly attention. Twitter is a famous microblogging website where users may share their ideas and opinions. Sentiment analysis on Twitter addresses the challenge of analysing tweets based on the opinions they express [1]. The implementation of the presidential election process cannot be separated from public interaction and opinion regarding the holding of the presidential election. One of the places where people can express their opinions is through social media. One of the social media that is often used by people to write their thoughts is twitter. Although many discussions about the 2024 Presidential Election are published on the Twitter page, there is no scientific calculation that can determine the tone of public opinion regarding the 2024 Presidential Election.

Natural language processing (NLP) is gaining traction in management research due to its capacity to automatically analyse and interpret human discourse [2]. The research conducted by Novianto and Munasatiya using the object of research, namely tweets about President Jokowi using Natural Language Processing (NLP) at the pre-processing stage of tweet data to facilitate tweet recognition and achieve accuracy of 93.26% [3]. This research uses NLP to create the model.

Long short-term memory (LSTM) is a recurrent neural network with a state memory and multilayer cell structure [4]. However, a study comparing the use of deep learning in conducting sentiment analysis between NLP and LSTM found better performance results with LSTM compared to NLP [5].

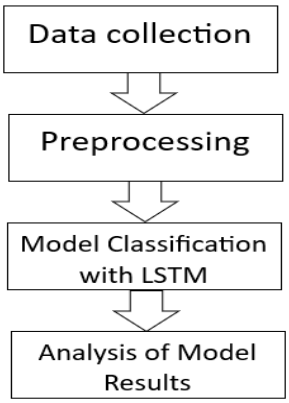
Based on the results of the above research, this study uses Natural Language Processing (NLP) techniques to process and analyze natural language data[5], and uses Long-Short Term Momory (LSTM) algorithms to analyze classification accuracy and precision [6]. The preprocessing method with NLP techniques was applied to tweet data obtained through the crawling process using the keyword "presidential election" on December 13, 2023 to February 4, 2024. The preprocessing technique consists of several stages, namely *case folding, cleaning, tokenization, filtering, and stemming* [7]. The processed data is then classified to determine positive, negative, or neutral sentiment.

Hasil analisis sentimen ini diharapkan dapat memberikan an rinci mengenai opini publik, sentiment patterns, and the dynamics of changes in public opinion ahead of the 2024 Presidential

Election. With a focus on the application of NLP and LSTM technology, this research is expected to contribute to a deeper understanding of public interaction and public opinion in the 2024 Presidential Election.

2. Research Method

This research has several stages that need to be completed properly. The stages of research carried out include tweet data collection, preprocessing, labeling, model classification with LSTM and analysis of model results. The research flow chart can be seen in the picture below.



Figurel 1. Relselarch Stagels Flow Chart

Data Collelcting

In the initial stage, what was done was to collect tweet data with searches for the yword "presidential election" taken using the*et harvest library*. *Thetweet harvest library* can help us more easily get tweet data [8] .

Preprosessing

Preprocessing is a very important step in sentiment analysis, data that has been collected needs to be preprocessed so as to produce clean data that is used to make *word vectors* and sentiment classification more accurate [7]. In the preprocessing stage in this study, the author will use NLP (Natural Language Processing). The steps taken are:

1. *Case Folding* is the process of converting all letters into lowercase, it is necessary to homogenize the text [7].
2. *Cleaning* is the process of cleaning tweets of unnecessary words to reduce noise. Omitted words can be HTML characters, keywords, *hashtags*, usernames, *urls*, and *emails*[9].
3. *Tokenizing* is the process of cutting sentences into words for word, this is done by cutting with reference to each space in a sentence [7].
4. *Filtering* is the stage of retrieving important words from token results. Stoplist/stopwords are non-descriptive words that can be discarded in a *bag-of-words approach* [9].
5. *Steamming* is the stage of finding the root word of each filtered word. At this stage, the process of returning various word formations into basic meanings [9].

Labeling

With this labeling, the data will be processed automatically and the score value is calculated based on *the lexicon* dictionary. Lexicon dictionary is a classification method that uses a dictionary containing opinion words to determine a positive and negative class sentiment from a text data [10].

A set of words that have gone through the text preprocessing stage needs to be labeled so that the set of words can be processed and counted. The calculation of the score will be carried out on the number detected based on the lexicon word dictionary. The formula for calculating the sentiment score is by subtracting the number of positive words by the number of negative words [11].

Score = (number of positive words) – (number of negative words) (1) After calculating the sentiment score based on the lexicon dictionary, the next step is to check the sentiment labeling results based on the score obtained. Sentiment labeling is done in three classes of sentiment, namely negative, neutral, and positive sentiments. Where the positive class has a score of >
1, the negative class has a score of < 0, and the neutral class has a score = 0.

Model Classification with LSTM

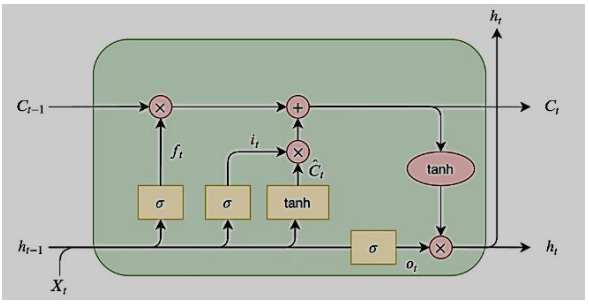


Figure 1. *Cell LSTM*

LSTM is a type of *Recurrent Neural Network (RNN).* LSTM itself can find hidden layers of each cell and is designed to store previous cell information. The LSTM method is used by classifying data long-term by storing it in memory cells [6].
At this stage, a model is created using the LSTM algorithm. In the model calling, the previously improved dataset is divided into two parts, namely training data and testing data. Below is an example of a manual calculation example from LSTM:
['indonesia', 'prestasi', 'bangga', 'bangsa', 'gelar', 'pilpres']
x = [23, 37, 92, 93, 195, 2]
w = 0,5

$_{f,\ i,\ o}$ = [1, 0.5 , 0.5]

Forgate Gate
ft = ($_f$ [h_{t-1}, $_t$] + $_f$) (2)
ft = ([0,5][0] +[0,5][23] + [1])
ft = 0,9999

Input gate
i_t = ($_i$ [h_{t-1}, $_t$] + $_i$) (3)
i_t = ([0.5][0] + [0.5][23]+ [0,5])
i_t = 0,9999

Kandidat state
$_t$ = t h([h_{t-1}, $_t$] +) (4)
$_t$ = t h([0,5][0] + [0,5][23] + [0])
$_t$ = 0,9999
Cell state
C_t = f_t ∗ C_{t-1} + i_t ∗ $_t$ (5)
C_t = 0,9999 x 0 + 0,9999 x 0,9999
C_t = 0,9998
Output gate
$_t$ = ([h_{t-1}, $_t$] +) (6)
$_t$ = ([0,5][0] + [0,5][23] + [0,5])
$_t$ = 0,9999
hidden state
h_t = $_t$ ∗ t h($_t$) (7)
h_t = [0,9999] ∗ t h(0,9998)
h_t = 0,9999 x 0,7615
h_t = 0,7614

2.5. Analysis of Model Results

After doing the modeling, we then tested the model using test data to confirm the sentiment classification of the 2024 presidential election tweet data. In this est, the author uses the Confusion Matrix as a method to test the model that has been created to determine the level of accuracy, precision, and recall.

A confusion matrix is a table that states the correct classification of the number of test data and the incorrect amount of test data [12]. An example of confusion matrix for binary classification is shown in Table 1.

		Predictions	
		1	0
Aktual	1	TP	FN
	0	FP	TN

Table 1. Confusion Matrix Examples

Information :
TP *(True Positive)* = The number of documents of the correct class 1 is classified as class 1
TN *(True Negative)* = The number of documents of the correct class 0 is classified as class 0
FP (False Positive) = number of documents of class 0 incorrectly classified as class 1
FN (False Negative) = Number of documents from class 1which are incorrectly classified as class 0

From the data above, other data can be obtained that are useful for measuring the performance of a model, as follows:

1. *Accuracy* is the ratio of correct predictions (positive and negative) to the overall data [12]. The formula for calculating *accuracy* is as follows.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (8)$$

2. *Precision* is the ratio of true positive predictions compared to overall positive prediction results [12]. The formula for calculating *precision* is as follows.

$$Precision = \frac{TP}{TP+FP} \quad (9)$$

3. *Recall* is thetruepositiveratio compared to all positivedata [12]. Theformula for calculating *recall* is as follows.

$$Recall = \frac{TP}{TP+FN} \quad (10)$$

Modeling Design

As shown in Figure 3, the author made a modeling design. The following is an explanation of the process that occurs in the form of a flowchart :

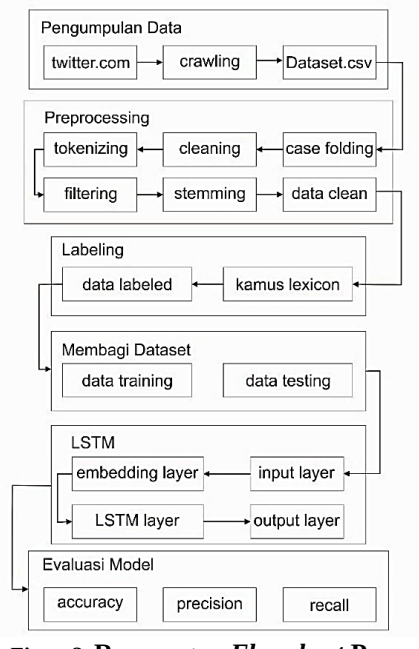


Figure2. Rancangan *Flowchart* Pemodelan LSTM

3. Results and Discussion

Data Collection

The tweet data used in this study is data obtained by searching for theyword "presidential election" taken from December 13, 2023 to February 4, 2024

using the *et harvest library*. In the ss *of crawling* tweet *data*, we got 1004 *tweet data* with the keyword "presidential election".

Table2. Hasil *Crawling* Data *Tweet*

No	full_text	Username
1	@zarryhendrik Habis selesai pilpres, langsung tutup akun twitter aja ðŸ˜,	Forgiveawayonli
2	Kalau punya gagasan yang matang Gak mungkin kalah terus waktu ikut Pilpres.	Shalihin_NAD
3	INDONESIA, prestasi yang membanggakan di tengah bangsa yang bersiap menggelar Pilpres 2024. https://t.co/3VgMpYKwtY	nirwarkot
4	Gong banget pilpres 2024 kali ini karna 52%nya tuh pemilih adalah pemuda	fullsun_desire
5	lagi kontestasi pilpres gini, jadi kangen sama Pak Arief Poyuono, kemana yaa?ðŸ˜,ðŸ¤”	imronmboo

Preprocessing

In the *preprocessing* stage of this study, the author used NLP *(Natural Language Processing). Preprocessing* is a step to clean *tweet* data consisting of words or symbols that are not important so that *tweet* data can be processed more easily.

a. Case Folding

Case *folding* is the process of converting all letters into lowercase, this is necessary to homogenize the text.

Table3. CaseFolding *StageResults*

No	Text CaseFolding
1	@zarryhendrik habis selesai pilpres, langsung tutup akun twitter aja ðŸ˜,
2	kalau punya gagasan yang matang gak mungkin kalah terus waktu ikut pilpres.
3	indonesia, prestasi yang membanggakan di tengah bangsa yang bersiap menggelar pilpres 2024. https://t.co/3vgmpykwty
4	gong banget pilpres 2024 kali ini karna 52%nya tuh pemilih adalah pemuda
5	lagi kontestasi pilpres gini, jadi kangen sama pak arief poyuono, kemana yaa?ðŸ˜,ðŸ¤”

b. Cleaning

The ss of clearing tweets of unnecessary words to reduce ii. Omitted words can be HTML characters, keywords, *hashtags*, *usernames*, *urls*, and *emails*.

Table4. Results of *Cleaning Stages*

No	Text Cleaning
1	habis selesai pilpres langsung tutup akun twitter aja ðŸ˜,
2	kalau punya gagasan yang matang gak mungkin kalah terus waktu ikut pilpres
3	indonesia prestasi yang membanggakan di tengah bangsa yang bersiap menggelar pilpres
4	gong banget pilpres kali ini karna nya tuh pemilih adalah pemuda
5	lagi kontestasi pilpres gini jadi kangen sama pak arief poyuono kemana yaaðŸ˜,ðŸ¤”

c. Tokenization

The ss of cutting sentences into word for word, this is doneby cutting with referenceto each space in a sentence.

Table5. Tokenization StageResults

No	Text Tokenization
1	['habis', 'selesai', 'pilpres', 'langsung', 'tutup', 'akun', 'twitter', 'aja', 'ðŸ˜,']
2	['kalau', 'punya', 'gagasan', 'yang', 'matang', 'gak', 'mungkin', 'kalah', 'terus', 'waktu', 'ikut', 'pilpres']
3	['indonesia', 'prestasi', 'yang', 'membanggakan', 'di', 'tengah', 'bangsa', 'yang', 'bersiap', 'menggelar', 'pilpres']
4	['gong', 'banget', 'pilpres', 'kali', 'ini', 'karna', 'nya', 'tuh', 'pemilih', 'adalah', 'pemuda']
5	['lagi', 'kontestasi', 'pilpres', 'gini', 'jadi', 'kangen', 'sama', 'pak', 'arief', 'poyuono', 'kemana', 'yaaðŸ˜,ðŸ¤”']

d. Filtering

Filtering is the stage of retrieving important words from token results. *Stoplist/stopwords* arenon-descriptive words that can be discarded in a *bag-of-words approach.*

Table 6. Results of Filtering Stages

No	Text Filtering
1	['habis', 'selesai', 'pilpres', 'langsung', 'tutup', 'akun', 'twitter', 'aja', 'ðŸ˜,']
2	['gagasan', 'matang', 'gak', 'kalah', 'pilpres']

3	['indonesia', 'prestasi', 'membanggakan', 'bangsa', 'menggelar', 'pilpres']
4	['gong', 'banget', 'pilpres', 'kali', 'karna', 'nya', 'tuh', 'pemilih', 'pemuda']
5	['kontestasi', 'pilpres', 'gini', 'kangen', 'arief', 'poyuono', 'kemana', 'yaaðŸ˜,ðŸ¤”"]

e. Stemming

Stemming is the process of finding the root word of each filtered word. At this stage, the process of returning various word forms into basic meanings is carried out.

Table 7. Hasil Tahapan *Stemming*

No	Text Stemming
1	['habis', 'selesai', 'pilpres', 'langsung', 'tutup', 'akun', 'twitter', 'aja', '']
2	['gagas', 'matang', 'gak', 'kalah', 'pilpres']
3	['indonesia', 'prestasi', 'bangga', 'bangsa', 'gelar', 'pilpres']
4	['gong', 'banget', 'pilpres', 'kali', 'karna', 'nya', 'tuh', 'pilih', 'pemuda']
5	['kontestasi', 'pilpres', 'gin', 'kangen', 'arief', 'poyuono', 'mana', 'yaa']

Labeling

At this stage of labeling, the data will be processed automatically, namely calculating the score value with the *xicon dictionary*. Lexicon dictionary is a classification method that uses a dictionary containing opinion words to determine a positive and negative class sentiment from a text data. For the dictionary of positive and negative words in can be from the source https://github.com/fajri91/InSet.

Table 8. Hasil Pelabelan Dengan Kamus *Lexicon*

No	text_preprocessed	score	sentimen
1	['habis', 'selesai', 'pilpres', 'langsung', 'tutup', 'akun', 'twitter', 'aja', '']	0	neutral
2	['gagas', 'matang', 'gak', 'kalah', 'pilpres']	-4	negative
3	['indonesia', 'prestasi', 'bangga', 'bangsa', 'gelar', 'pilpres']	5	positive
4	['gong', 'banget', 'pilpres', 'kali', 'karna', 'nya', 'tuh', 'pilih', 'pemuda']	0	neutral
5	['kontestasi', 'pilpres', 'gin', 'kangen', 'arief', 'poyuono', 'mana', 'yaa']	0	neutral

Model Classification with LSTM

At this stage, model creation is carried out using the LSTM algorithm. In making models, previously improved datasets are divided into two, namely training data and testing data. For *training data* 80%, while testing data is 20%, with as many as 803 *training data* and 201 testing data.

Next, create a model to train the training data so as to generatea model to predict *tweet* data into positive, neutral, or negative sentiment.

```
Layer (type)                 Output Shape              Param #
=================================================================
embedding_11 (Embedding)     (None, 46, 32)            160000

lstm_11 (LSTM)               (None, 16)                3136

dropout_11 (Dropout)         (None, 16)                0

dense_11 (Dense)             (None, 3)                 51

=================================================================
Total params: 163187 (637.45 KB)
Trainable params: 163187 (637.45 KB)
Non-trainable params: 0 (0.00 Byte)
```

Figure 3. LSTM Model Parameters

Analysis of Model Results

After making the next model, we measure the performance of the model that has been created using a *confusion matrix*. *A confusion matrix* is a tablethat states the correct classification of the amount of test data and the incorrect amount of test data. The*confusion matrix* table can be seen in Figure 5.

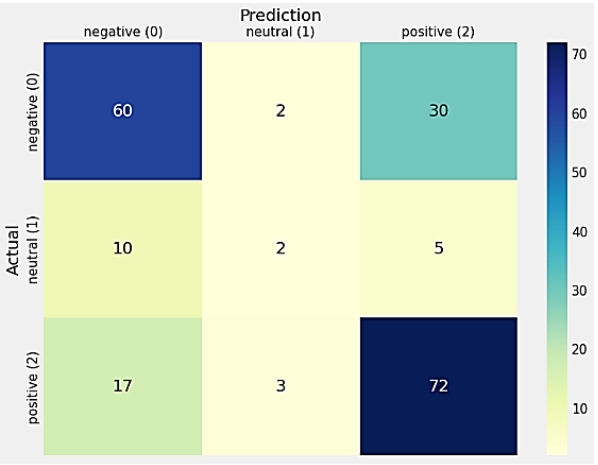


Figure 4. *Confusion Matrix*

Diketahui :

Class Positive

a. True Positive

TP = 72

b. True Negative

TN = 60+2+10+2

TN = 74

c. False Positive

FP = 30+5

FP = 35

d. False Negative

FN = 17+3

FN = 20

Class Neutral

a. True Positive

TP = 2

b. True Negative

TN = 60+30+17+72

TN = 179

c. False Positive
FP = 2+3
FP = 5
d. False Negative
FN = 10+5
FN = 15

Class Negative
a. True Positive
TP = 760
b. True Negative
TN = 2+5+3+72
TN = 82
c. False Positive
FP = 10+17
FP = 27
d. False Negative
FN = 2+30
FN = 32

- TP = (Nilai TP dijumlahkan "Class Positive, Class Neutral, Class Negative")
  TP = 72+2+60 = 134
- TN = (Nilai TN dijumlahkan "Class Positive, Class Neutral, Class Negative")
  TN = 74+179+82 = 335
- FP = (Nilai FP dijumlahkan "Class Positive, Class Neutral, Class Negative")
  FP = 35+5+27 = 67
- FN = (Nilai FN dijumlahkan "Class Positive, Class Neutral, Class Negative")
  FN = 20+15+32 = 67

Mencari Nilai Akurasi :

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \text{ x } 100\%$$

$$Accuracy = \frac{134+335}{134+335+67+67} \text{ x } 100\%$$

$$Accuracy = \frac{469}{603} \text{ x } 100\%$$

Accuracy = 0,777 x 100%
Accuracy = 78%

Finding Precision Value :

$$Precision = \frac{TP}{TP+FP} \text{ x } 100\%$$

$$Precision = \frac{134}{134+67} \text{ x } 100\%$$

$$Precision = \frac{134}{201} \text{ x } 100\%$$

Precision = 0,666 x 100%
Precision = *67%*

Mencari Nilai *Recall :*

$$Recall = \frac{TP}{TP+FN} \text{ x } 100\%$$

$$Recall = \frac{134}{134+67} \text{ x } 100\%$$

$$Recall = \frac{134}{201} \text{ x } 100\%$$

Recall = 0,666 x 100%
Recall = *67%*

From *the analysis* of Twitter sentiment towards the 2024 presidential election, it produced 53.2% positive sentiment, 3.5% neutral sentiment, and 43.3% negative sentiment which can be seen in the following digram.

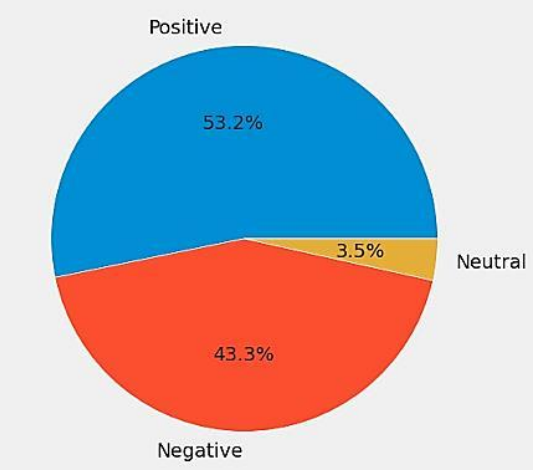


Figure 5. Analysis results in pie charts

4. Conclusion

Based on the sults of Twitter Sentiment Analysis Using *Natural Language ssing (NLP) Method* and *Long Short Term Memory (LSTM) Algorithm* in the2024 Presidential Election, theauthor can conclude that this study resulted in poor model performance by producing an *accuracy* rate of 78%, *precision* of 67% and *recall* of *67%*. This study succeeded in classifying three s of *tweet* sentiment with a combination of NLP and LSTM methods which resulted in positive sentiment of 53.2%, neutral sentiment of 3.5%, and negative sentiment of 43.3%.

This research can be developed by combining other models and increasing the number of datasets to be tested so as to produce even better accuracy scores.

References

Giachanou, A., & Crestani, F. (2016). Like it or not: A survey of Twitter sentiment analysis methods. ACM Computing Surveys (CSUR), 49(2), 1–41. https://doi.org/10.1145/2843948

Kang, Y., Cai, Z., Tan, C.-W., Huang, Q., & Liu, H. (2020). Natural language processing (NLP) in management research: A literature review. Journal of Management Analytics, 7(2), 139–172. https://doi.org/10.1080/23270012.2019.1648489

Munasatya, N., & Novianto, S. (2020). Natural language processing untuk sentimen analisis Presiden Jokowi menggunakan Multi Layer Perceptron. Techno. Com, 19(3), 237–244. https://doi.org/10.30659/techno.v19i3.237-244

Smagulova, K., & James, A. P. (2019). A survey on LSTM memristive neural network architectures and applications. The European Physical Journal

Special Topics, 228(10), 2313–2324. https://doi.org/10.1140/epjst/e2019-900149-0

Fathoniah, S., & Rozikin, C. (2022). Analisis sentimen masyarakat terhadap teroris dalam media sosial Twitter menggunakan NLP. Jurnal Ilmiah Wahana Pendidikan, 8(13), 412–419.

Astari, Y. Y., Afiyati, A., & Rozaqi, S. W. (2021). Analisis sentimen multi-class pada sosial media menggunakan metode Long Short-Term Memory (LSTM). Jurnal Linguistik Komputasional, 4(1), 8–12.

Rahman, M. Z., Sari, Y. A., & Yudistira, N. (2021). Analisis sentimen tweet COVID-19 menggunakan word embedding dan metode Long Short-Term Memory (LSTM). Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, 5(11), 5120–5127.

Satria, H. (2024, January 22). Crawl data Twitter menggunakan Tweet Harvest. helmisatria.com. https://helmisatria.com/blog/crawl-data-twitter-menggunakan-tweet-harvest

Hidayah, Y., & Sahibu, S. (2021). Algoritma multinomial naïve bayes untuk klasifikasi sentimen pemerintah terhadap penanganan Covid-19 menggunakan data Twitter. Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), 5(4), 820–826. https://doi.org/10.29207/resti.v5i4.3146

Furqan, M., Sriani, S., & Sari, S. M. (2022). Analisis sentimen menggunakan K-Nearest Neighbor terhadap new normal masa Covid-19 di Indonesia. Techno.Com, 21(1), 51–60. https://doi.org/10.33633/tc.v21i1.5446

Ismail, A. R., Bagus, R., & Hakim, F. (2023). Implementasi lexicon based untuk analisis sentimen dalam mengetahui trend wisata pantai di DI Yogyakarta berdasarkan data Twitter. Unpublished manuscript, 1(1), 37–46.

Rahim, A., Kusrini, K., & Luthfi, E. T. (2020). Convolutional neural network untuk klasifikasi penggunaan masker. Inspiration: Jurnal Teknologi Informasi dan Komunikasi, 10(2), 109. https://doi.org/10.35585/inspir.v10i2.2569