Eduvest – Journal of Universal Studies
Volume 4 Number 05, May, 2024
p- ISSN 2775-3735- e-ISSN 2775-3727

SENTIMENT ANALYSIS OF FLO APPLICATIONS FOR WOMEN'S NEEDS USING THE CNN AND LSTM ALGORITHMS

Alifhia Dhiya Herlia[1], Miftah Andriansyah[2]
[1,2] Program Studi Magister Manajemen Sistem Informasi, Universitas Gunadarma, Indonesia
Email: alifhia121099@gmail.com

ABSTRACT

One application that helps women is an application that can track menstrual schedules, plan pregnancy and track pregnancy schedules until the estimated time of delivery. An example of an application that is widely used on the Google Play Store is FLO. FLO application that has been downloaded more than 100 million times and has been reviewed as many as 2 million reviews on the Google Play Store seen on January 12, 2023. Sentiment Analysis is an ongoing research field in the field of text mining and also the computational treatment of opinions, sentiments and subjectivity text that can be used as an evaluation of an application. The method chosen in this sentiment analysis research is CNN and RNN with Long Short Term Memory (LSTM) variants. In this study, the data used to carry out sentiment analysis is review data in text form. The results of sentiment analysis with the research object of FLO application reviews were 12,000 review data selected by country, namely Indonesia, which had more positive reviews, followed by neutral reviews and finally negative reviews. In the training data obtained, accuracy and loss by doing three epochs, namely 20, 50, 100 on the CNN and LSTM algorithms are good enough and not overfitting. Data testing is also carried out using confusion matrix and classification report, based on the two algorithm comparisons, it can be seen that the superior one is using the LSTM algorithm, with accuracy of 92.67% and 93% respectively, while the CNN accuracy results are 84.08% and 84% respectively.

KEYWORDS *Flo, Sentiment, Convolutional Neural Network, Recurrent Neural Network, Long Short-Term Memory*

How to cite:	Alifhia Dhiya Herlia, Miftah Andriansyah. (2024). Sentiment Analysis Of Flo Applications For Women's Needs Using The CNN And LSTM Algorithms. *Journal Eduvest. 4* (5): 4061-4078
E-ISSN:	2775-3727
Published by:	https://greenpublisher.id/

INTRODUCTION

The use of technology in Indonesia today is widespread among both women and men. Women are also considered a significant part of society that extensively uses mobile applications. A mobile application is a type of application software designed to operate on mobile devices such as smartphones or tablets. Mobile applications often provide services similar to those accessed on PCs. These applications are generally small, individual software units with limited functionality. The use of such application software was initially popularized by Apple Inc. through the App Store, which offered thousands of applications for the iPhone, iPad, and iPod Touch. Applications are divided into two main categories: native apps and web apps. Native apps are created for a specific mobile operating system, usually iOS or Android. Native apps enjoy better performance and a smoother user interface (UI) and typically undergo a much stricter development and quality assurance process before release (Al-adwan, 2020).

One application that helps women is one that can track menstrual schedules, plan pregnancies, and monitor pregnancy schedules until the estimated time of delivery. An example of a widely used app on the Google Play Store is FLO. As of January 12, 2023, the FLO app has been downloaded over 100 million times and has received 2 million reviews on the Google Play Store. The FLO app was released on April 12, 2016, with the aim of helping women worldwide track their menstrual schedules, and it automatically calculates the estimated start date of the next period. Period tracking and fertility apps have gained popularity in recent years, often marketed as tools for self-empowerment through self-knowledge (Kressbach, 2021). Besides tracking menstrual schedules, the FLO app can also be used for pregnancy planning as it can monitor ovulation periods, and pregnant women using the FLO app can determine the estimated delivery date along with related pregnancy articles that are very helpful. The FLO app is widely used, as noted in an article on Kompas.com, which recommends FLO for monitoring menstrual cycles and fertility (Utami, 2021).

Sentiment analysis has been an active research topic for a decade, and its popularity has increased with the emergence of diverse online opinion resources (Sánchez-Rada & Iglesias, 2019). Sentiment analysis is a field of research within text mining and involves the computational treatment of opinions, sentiments, and subjectivity in text (Medhat, Hassan and Korashy, 2014). The analysis performed in sentiment analysis automatically assesses customer feedback, such as survey responses and social media conversations. Furthermore, sentiment analysis allows a brand to understand what customers like or dislike, enabling the brand to adjust products and services to meet customer needs (Xu et al., 2019). With advances in science and technology, algorithms for analyzing text have also rapidly improved. The use of advanced artificial intelligence techniques can be an effective tool for conducting in-depth research, particularly in sentiment analysis (Paputungan and Jacobus, 2021).

The methods chosen for this sentiment analysis research are CNN and RNN with the Long Short Term Memory (LSTM) variant. The selection of CNN and RNN with the LSTM variant for this sentiment analysis is to compare which algorithm will provide better accuracy. Convolutional Neural Network (CNN) is an automated design algorithm and an artificial neural network method that is also used in radiology and has become dominant in several computer vision tasks. CNN learns the spatial hierarchy of features through backpropagation using several building blocks, namely fully connected layers, convolutional layers, and pooling layers (Yamashita et al., 2018). The final layer of the CNN architecture uses a classification layer to provide the final classification output (Li et al., 2021). CNN is generally used for image classification with 2D CNN or video classification with 3D CNN (Budiman and Abadi, 2023)

Recurrent Neural Network (RNN) is essentially a type of neural network where the output from previous steps is fed into the current step, with the hidden state being the main and most important feature of RNN as it remembers some information about the sequence (Sudharsan & Ganesh, 2022). LSTM is a development of the deep learning RNN method (Purnasiwi et al., 2023). LSTM's advantage is its ability to handle long-term dependencies better, avoiding the long-term dependency problem (Le et al., 2019). This is due to its ability to remember information for a long time, store information in long sequences, and model complex sequential data very efficiently (Zhao et al., 2020).

Based on the above explanation, the researcher is very interested in using sentiment analysis on the FLO application for women's needs by using two algorithms for comparison, CNN and RNN with the LSTM variant. The reason the researcher chose this topic is that the FLO app is quite popular for women's needs, and the researcher has been using the FLO app since 2017 until this research was conducted. Meanwhile, both sentiment analysis methods are chosen because they have high accuracy, with the CNN algorithm by Yuliska et al. achieving the highest accuracy of 98%, and the RNN with the LSTM variant by Wahyudi D. and Sibaroni Y. achieving the highest accuracy of 95% (Wahyudi & Sibaroni, 2022; Yuliska et al., 2021). Therefore, the researcher compares these two algorithms to determine which sentiment analysis is more accurate for the FLO app, a comparison that has not been made by previous researchers.

To ensure this research remains focused on its topic, the researcher sets the following limitations: the researcher uses the FLO application on the Google Play Store with reviews from Indonesian citizens, and the researcher uses CNN and RNN with the LSTM variant algorithms. The goal of this research is to compare the results of sentiment analysis of the FLO app between the CNN and LSTM methods, which can later be used as a reference for future research. Additionally, this research can serve as a consideration for developers and system managers to choose a more accurate sentiment analysis algorithm to better develop their systems according to consumer needs.

RESEARCH METHOD

The subject of this research is the opinions or sentiments of FLO application users towards the app, gathered from reviews on the Google Play Store. In this study, the data used for sentiment analysis is in the form of text reviews. Additionally, the data collection is limited to the period from August 5, 2016, to January 11, 2023, with a total of 12,000 reviews. The collected data is then split into training and testing datasets. The ratio used is 80% for training and 20% for testing, resulting in 9,600 training data and 2,400 testing data.

The software used in this research is Google Collaboratory with Python programming language, which will render the code into visual assets (Nelson & Hoover, 2020). The operating system used is macOS 12.5, with hardware specifications including an Apple M2 chip and 8GB RAM. The research methodology is illustrated in Figure 1.

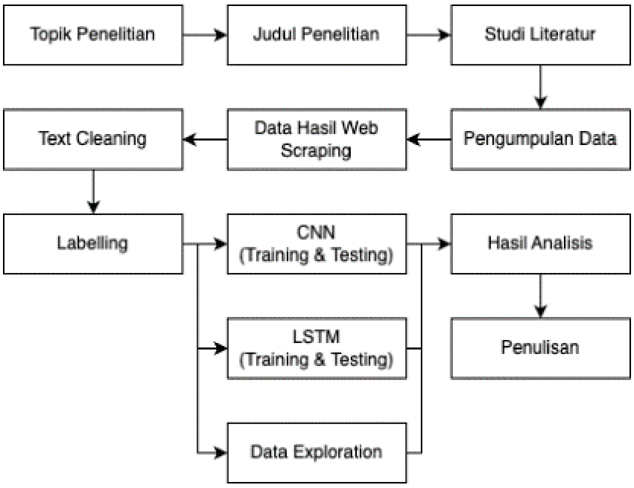


Figure 1. Research Stages

In this research stage, the basic architectural models of the two algorithms are used. CNN provides an optimal architecture for uncovering and learning key features in images and time-series data. Its three main types of layers are the convolutional layer, pooling layer, and fully connected (FC) layer. The convolutional layer is the first layer of the convolutional network. The complete architecture of CNN can be seen in Figure 2.

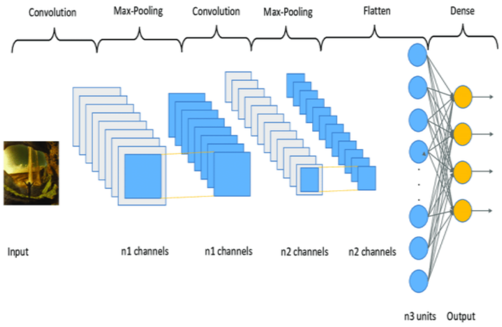


Figure 2. CNN Architecture
Source: García-Ordás et al., 2020, page 5

On the other hand, LSTM is an artificial recurrent neural network (RNN) architecture used in deep learning. LSTM has three gates that control the flow of information in and out of the memory cell: forget gate, input gate, input modulation gate, and output gate. The forget gate is responsible for discarding irrelevant information that is no longer needed by the system. The LSTM architecture is shown in Figure 3.

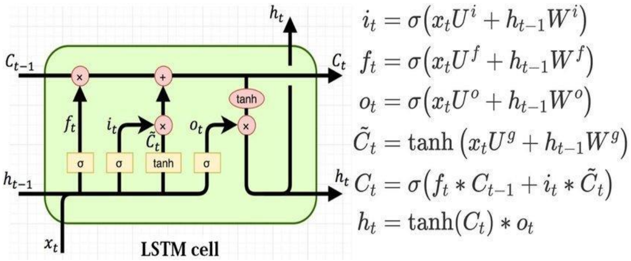

h_t
C_{t-1}
C_t
f_t
i_t
$\tilde{C}_t$
o_t
tanh
σ
h_{t-1}
h_t
x_t
LSTM cell
$i_t = \sigma(x_t U^i + h_{t-1} W^i)$
$f_t = \sigma(x_t U^f + h_{t-1} W^f)$
$o_t = \sigma(x_t U^o + h_{t-1} W^o)$
$\tilde{C}_t = \tanh(x_t U^g + h_{t-1} W^g)$
$C_t = \sigma(f_t * C_{t-1} + i_t * \tilde{C}_t)$
$h_t = \tanh(C_t) * o_t$

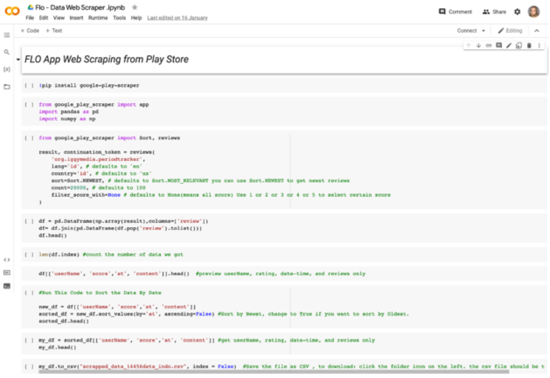
Figure 3. LSTM Architecture
Source: Varsamopoulos et al., 2019, page 4

In the final stage, the data that has undergone testing and sentiment analysis is subjected to data exploration. Data exploration involves presenting data in an easily readable visualization. The results can include graphs that show significant levels in the data source (Fahmi et al., 2020).

RESULT AND DISCUSSION

Data Collection with Word Scraping Techniques

FLO application review data is obtained by web scraping. In this study, the authors used data sources from the Google Play Store website. From the website, the required information will be taken, namely FLO application user reviews in Indonesian and English. The data to be scraped consists of several attributes, namely username, rating, date, and review. In order to see FLO application review data, it can be done by writing the keyword "FLO" in the search field on the Google Play Store website, so that the FLO application review page is obtained. Furthermore, scraping is carried out with the picture listed in Figure 4.


Flo - Data Web Scraper .ipynb
File Edit View Insert Runtime Tools Help Last edited on 16 January
Comment
Share
+ Code + Text
Connect
Editing

FLO App Web Scraping from Play Store

```
!pip install google-play-scraper
```

```
from google_play_scraper import app
import pandas as pd
import numpy as np
```

```
from google_play_scraper import Sort, reviews

result, continuation_token = reviews(
    'org.iggymedia.periodtracker',
    lang='id', # defaults to 'en'
    country='id', # defaults to 'us'
    sort=Sort.NEWEST, # defaults to Sort.MOST_RELEVANT you can use Sort.NEWEST to get newst reviews
    count=20000, # defaults to 100
    filter_score_with=None # defaults to None(means all score) Use 1 or 2 or 3 or 4 or 5 to select certain score
)
```

```
df = pd.DataFrame(np.array(result),columns=['review'])
df= df.join(pd.DataFrame(df.pop('review').tolist()))
df.head()
```

```
len(df.index) #count the number of data we got
```

```
df[['userName', 'score','at', 'content']].head()  #preview userName, rating, date-time, and reviews only
```

```
#Run This Code to Sort the Data By Date

new_df = df[['userName', 'score','at', 'content']]
sorted_df = new_df.sort_values(by='at', ascending=False) #Sort by Newst, change to True if you want to sort by Oldest.
sorted_df.head()
```

```
my_df = sorted_df[['userName', 'score','at', 'content']] #get userName, rating, date-time, and reviews only
my_df.head()
```

```
my_df.to_csv("scrapped_data_14456data_indo.csv", index = False)  #Save the file as CSV , to download: click the folder icon on the left. the csv file should be t
```

Figure 4. Web Scrapping FLO Applications

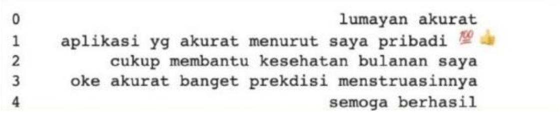
The data obtained from the results of web scraping is 14,456 reviews consisting of all reviews from August 5, 2016 to January 11, 2023. The reviews obtained also consist of Indonesian and English reviews. From the data that has been obtained, the author can proceed to the text cleaning stage before finally carrying out the labeling stage.

Text Cleaning Results

After the data was obtained from the results of web scraping, the author did text cleaning. This is because the data obtained has not been structured, so it is necessary to process changing the form of unstructured data into structured data according to needs. Text cleaning is done by uniformizing the FLO application review text, it is intended to facilitate reading and processing as well as data analysis. In addition, text cleaning prepares text into data that will undergo processing at the next stage. In carrying out the cleaning stage, several stages are carried out as follows:

1. Case Folding

At this stage of *case folding, the case* or typeface in the review is uniformized into lowercase. The following output results from the *case folding* stage in this study are shown in Figure 5:

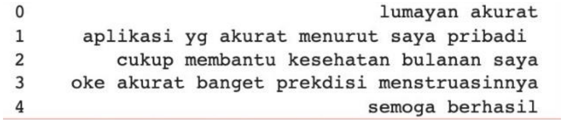


```
0                                   lumayan akurat
1     aplikasi yg akurat menurut saya pribadi 💯👍
2          cukup membantu kesehatan bulanan saya
3      oke akurat banget prekdisi menstruasinnya
4                                  semoga berhasil
```

Figure 5. Case folding results

2. *Filtering*

At the *filtering* stage, the symbols in the review are removed so that the review only contains letters as shown in Figure 6 below:

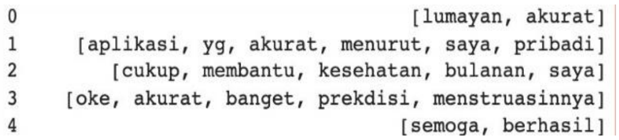


```
0                                   lumayan akurat
1       aplikasi yg akurat menurut saya pribadi
2           cukup membantu kesehatan bulanan saya
3      oke akurat banget prekdisi menstruasinnya
4                                  semoga berhasil
```

Figure 6. Filtering results

3. *Tokenization*

At the *tokenization stage,* the author performs a hyphenation of syllables from the review sentence. Here are the output results of this stage in Figure 7:

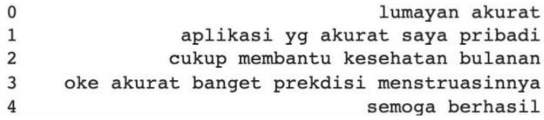


```
0                                     [lumayan, akurat]
1     [aplikasi, yg, akurat, menurut, saya, pribadi]
2        [cukup, membantu, kesehatan, bulanan, saya]
3    [oke, akurat, banget, prekdisi, menstruasinnya]
4                                    [semoga, berhasil]
```

Figure 7. Tokenization results

4. *Stopword*

At this stage, the writer discards words that are not descriptive. The following output results are shown in Figure 8:

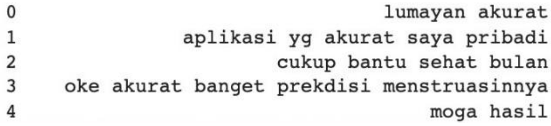


```
0                            lumayan akurat
1               aplikasi yg akurat saya pribadi
2               cukup membantu kesehatan bulanan
3    oke akurat banget prekdisi menstruasinnya
4                            semoga berhasil
```

Figure 8. Stopword results

5. *Stemming*

At the stemming stage, the author converts syllables into root words. The following is presented the output of the results of this stemming stage in Figure 9:

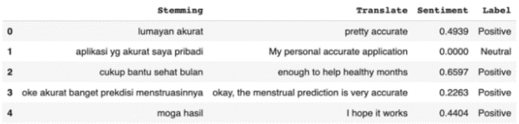


```
0                            lumayan akurat
1               aplikasi yg akurat saya pribadi
2                      cukup bantu sehat bulan
3    oke akurat banget prekdisi menstruasinnya
4                                moga hasil
```

Figure 9. Stemming results

***Sentiment Class* Labeling**

Data that has passed the web *scraping* and *text cleaning* process is categorized by the author or *labeling*. The purpose of *labelling* clean review data is to avoid *redundancy* or duplicate comment data. The labeling process is carried out to divide data into 3 classes, namely positive, neutral, and negative. *Labelling* is based on sentiment values, where sentiment values of 0 are categorized as neutral, sentiment values up to +1 are categorized as positive, and sentiment values up to -1 are categorized as negative. Here's an example of data that has been labeled a class based on its sentiment value in Figure 10::



	Stemming	Translate	Sentiment	Label
0	lumayan akurat	pretty accurate	0.4939	Positive
1	aplikasi yg akurat saya pribadi	My personal accurate application	0.0000	Neutral
2	cukup bantu sehat bulan	enough to help healthy months	0.6597	Positive
3	oke akurat banget prekdisi menstruasinnya	okay, the menstrual prediction is very accurate	0.2263	Positive
4	moga hasil	I hope it works	0.4404	Positive

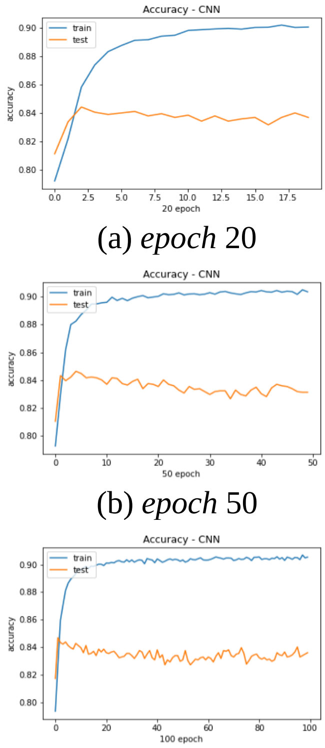
Figure 10. Example of labeling review data

From Figure 10 above, we can see examples of reviews with a sentiment value of 0.4939 given a *positive* class, reviews with a sentiment value of 0 given a *neutral* class, and reviews with a sentiment value with a negative value given a *negative class*.

CNN Algorithm Data *Training* Results

In this implementation, there is training of a *neural network* with 2 types of datasets, namely *training* and validation (*test*). *The training dataset* is the data used to change the weights of the model to train the model. In other words, these are examples of data used by the model to train itself (Afif et al., 2021). In this study, the authors used more than one *epoch*, *epochs* 20, 50 and 100. As the number of *epochs* increases, more and more *weight* changes in the *Neural*

Network and the curve curves from an inappropriate curve to align with an *overfitting curve* (Paputungan and Jacobus, 2021). The following Figure 11 presents accuracy testing data with CNNs at various *epochs*:



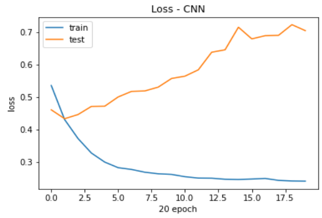
(a) *epoch* 20

(b) *epoch* 50

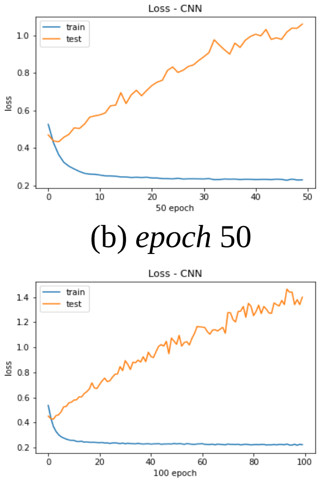
(c) *epoch* 100

Figure 11. Graph *Accuracy Training* and *Test* Some CNN Testing

In Figure 11a. number of *epoch* 20, the farthest difference is in *epoch* 16 with the difference still below 0.08. The accuracy of both generally increases from *epoch* 1 to 2, and tends to be stable from *epoch* 3 to 20. In Figure 11b the number of *epochs* 50, the farthest difference is in *epoch* 40 with the difference still below 0.08. The accuracy of both generally increases from *epoch* 1 to 2, and tends to be stable from *epoch* 3 to 50. In Figure 11c the number of *epochs* 100, the farthest difference is found in *epoch* 70 with the difference still below 0.08. The accuracy of both generally increases from *epoch* 1 to 2, and tends to be stable from *epoch* 3 to 100. Furthermore, loss testing is also shown in *the training* and *test* data shown Figure 12 with the following results:



(a) *epoch* 20



(b) *epoch* 50

(c) *epoch* 100

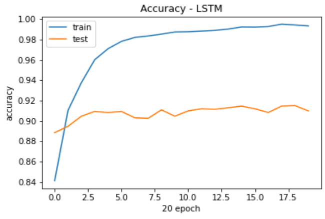
Figure 12. Chart *Loss Training* and *Test* Some CNN Testing

In Figure 12a. number of *epochs* 20, the farthest difference is in *epoch* 15 with the difference still below 0.08. *Loss* in test data tends to increase from epoch 1 to 20, while *loss* in *training data* tends to decrease from epoch 1 to 10, and stable from epoch 10 to 20. In Figure 12b. number of epochs 50, the farthest difference is in epoch 50 with an insufficiently large difference of 0.08. *Loss* in test data tends to increase from epoch 1 to 50, while *loss* in training data tends to decrease from epoch 1 to 10, and stable from epoch 10 to 50. Next in Figure 12c. number of epochs 100, the farthest difference is in epoch 100 with an insufficiently large difference of about 0.08. *Loss* in test data tends to increase from epoch 1 to 100, while *loss* in *training* data tends to decrease from epoch 1 to 10, and stable from epoch 10 to 100.

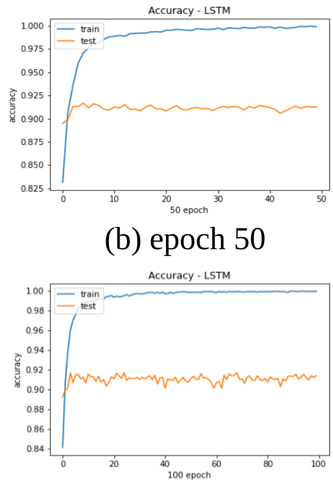
From the results of the accuracy and *loss* tests, it can be seen that the model is good enough to show the accuracy of *training data* with validation (*test*) not much different so that it does not experience *over fitting*. Furthermore, the CNN model will be tested for trust by testing with *data testing*.

LSTM Algorithm Data *Training* Results

The test was carried out the same as the previous test but used a different model, namely LSTM. The following Figure 13 presents accuracy testing data with LSTM at various epochs:



(a) epoch 20

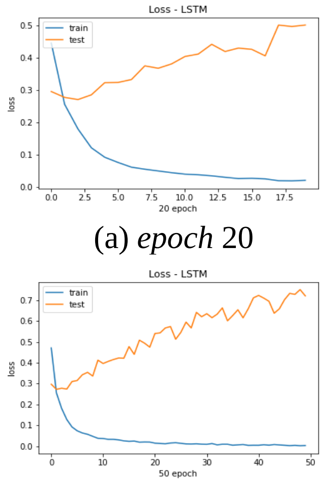


(b) epoch 50

(c) epoch 100

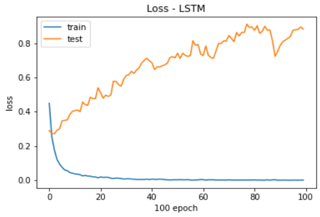
Figure 11. Graph *of Accuracy Training* and *Test* Some LSTM Tests

In Figure 13a. number of *epoch* 20, the farthest difference is in *epoch* 16 with the difference still below 0.08. The accuracy of both generally increases from *epochs* 1 to 3, and tends to be stable from *epochs* 4 to 20. In Figure 13b the number of *epochs* is 50, the farthest difference is in *epoch* 40 with the difference still below 1. The accuracy of both generally increases from *epoch* 1 to 3, and tends to be stable from *epoch* 4 to 50. In Figure 13c the number of *epochs* 100, the farthest difference is found in epoch 62 with the difference still below 0.08. The accuracy of both generally increases from *epoch* 1 to 3, and tends to be stable from *epoch* 4 to 100. Furthermore, loss testing was also tested on *training* and *test* data with the following results in Figure 14:



(a) *epoch* 20

(b) *epoch* 50



(c) *epoch* 100

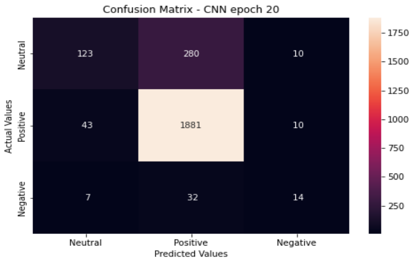
Figure 14. Chart *Loss Training* and *Test* Multiple LSTM Testing

In Figure 14a. number of *epoch* 20, the farthest difference is in *epoch* 16 with the difference still below 0.05. *Loss* in test data tends to increase from *epoch* 1 to 20, while *loss* in training data tends to decrease from *epoch* 1 to 7, and stable from *epoch* 8 to 20. In Figure 14b. number of *epoch* 50, the farthest difference is in *epoch* 49 with a difference below 0.08. *Loss* in test data tends to increase from *epoch* 1 to 50, while *loss* in *training data* tends to decrease from *epoch* 1 to 7, and stable from *epoch* 8 to 50. Furthermore, in Figure 14c. the number of *epochs* is 100, the farthest difference is in *epoch* 99 with a difference of less than 0.08. *Loss* in test data tends to increase from *epoch* 1 to 100, while *loss* in *training data* tends to decrease from *epoch* 1 to 10, and stable from *epoch* 10 to 100.

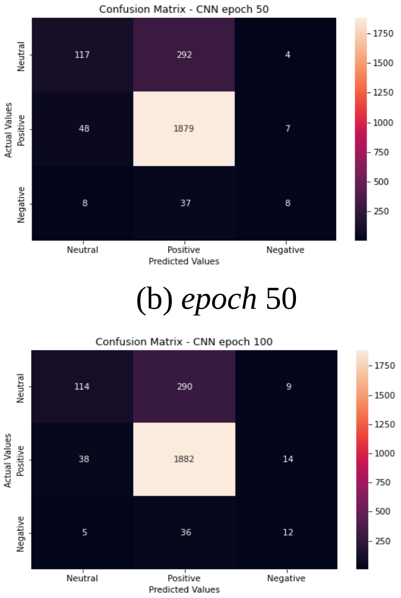
From the results of accuracy and *loss tests*, it can be seen that the LSTM model is good enough by showing the accuracy of training data with validation (*test*) not much different so that it does not experience *over fitting*.

Evaluate Results With *Confusion Matrix*

The results of *training* tests and *tests* that have been analyzed in the previous sub-chapter, then evaluation of the results using *the confusion matrix method* is *carried out.Confusion matrix* is a matrix or table that functions to determine how accurate the classification process is (Yan et al., 2022). *The confusion matrix* was tested on each model with a different number of *epochs*. The following Figure 15 presents *the confusion matrix* on the CNN model:



(a) *epoch* 20



(b) *epoch* 50

(c) *epoch* 100

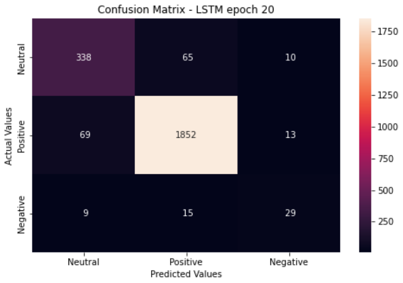
Figure 15. *Confusion Matrix* Multiple CNN Testing

The following data from the calculation of *accuracy, precision* and *recall* are summarized in Table 1:

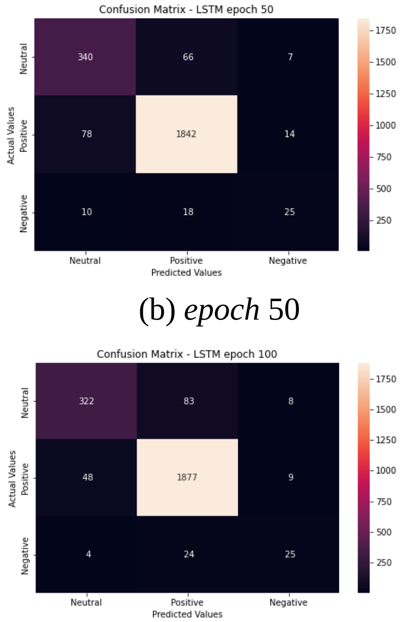
Tabel 1. Hasil Analisis *Confusion Matrix* CNN

	Accuracy	***Recall***	***Precision***
***Epoch* 20**	84,08%	51,15%	66,02%
***Epoch* 50**	83,50%	46,86%	64,94%
***Epoch* 100**	83,67%	49,19%	64,04%

The following confusion matrix is presented in the LSTM model in Figure 16:



(a) *epoch* 20



(b) *epoch* 50

(c) *epoch* 100

Figure 16. *Confusion Matrix* Multiple LSTM Testing

The following data from the calculation of *accuracy*, *precision* and *recall* are summarized in Table 2:

Table 2. LSTM *Confusion Matrix* Analysis Results

	Accuracy	***Recall***	***Precision***
***Epoch* 20**	92,46%	77,44%	77,63%
***Epoch* 50**	91,96%	74,91%	76,48%
***Epoch* 100**	92,67%	74,06%	80,08%

Furthermore, the two models are compared so that the following graph is produced in Figure 17:

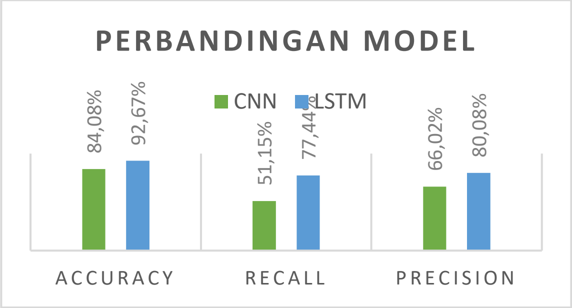


Figure 17. Comparison of CNN and LSTM Models with *Confusion Matrix*

From the graph, it can be seen that the LSTM model is far superior to the CNN model ranging from *accuracy*, *precision* to *recall* values. Thus, it can be concluded that the use of the LSTM method produces greater *accuracy, precision and recall* values and can be interpreted the results of sentiment analysis with the LSTM method to produce predictions that are good enough to find out the sentiment of FLO application reviews on *Google Playstore*. In addition, sentiment analysis using the LSTM method produces better results than the CNN method.

Evaluation of Results with Classification Report

Classification reports are used to measure the quality of predictions from classification algorithms. In this report, you can see the number of correct and incorrect predictions. From the results of the report, it can be seen that the accuracy value in the CNN model ranges from 0.83-0.84 or can be used as a percentage, which is 83%-84%. The value is quite high and it can be interpreted that the CNN model is quite accurate in analyzing sentiment in this study. Furthermore, there are also precision, recall, and F1-Score results with various values. The macro average and weighted average values are also presented in all three. The macro average calculates the matrix freely for each class and then takes the average, while the weighted average calculates the average by taking into account the weights in each data.

To forecast the following model weighted average values are presented in the precision, recall, and F1-Score data in the CNN model, shown in Table 3:

Table 3 CNN Classification Report Analysis Results

Epoch	*Precision*	*Recall*	*F1-Score*
***Epoch* 20**	82%	84%	81%
***Epoch* 50**	81%	83%	80%
***Epoch* 100**	82%	84%	81%

Precision values range from 81%-82%, recall 83%-84% and F1-Score 80%-81%, thus the CNN model is quite good at predicting with all three values being quite high. The F1-Score score has a good score indicating that the classification model in this study has good precision and recall as well.

From the results of the report, it can be seen that the accuracy value in the LSTM model ranges from 92%-93% or can be used as a percentage, which is 92%-93%. The value is quite high and it can be interpreted that the LSTM model is very accurate in analyzing sentiment in this study. Furthermore, there are also precision, recall, and F1-Score results with various values to forecast models with weighted average values as follows in Table 4:

Table 4. LSTM Classification Report Analysis Results

Epoch	*Precision*	*Recall*	*F1-Score*
***Epoch* 20**	92%	92%	92%
***Epoch* 50**	92%	92%	92%
***Epoch* 100**	92%	93%	92%

The results of both models, namely CNN and LSTM in the *classification report* results are considered quite good and both are able to analyze sentiment well. Thus, the results obtained from the *classification report* are compared with the help of the graph in Figure 18 as follows:

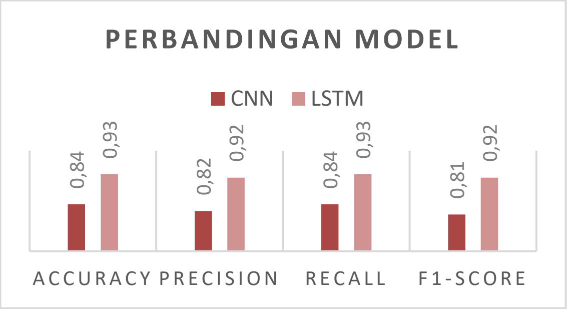


Figure 18. Comparison of CNN and LSTM Models with Classification Report

From the graph listed in Figure 16 it can be seen that the LSTM model is far superior to the CNN model starting from accuracy, precision, recall, and F1-Score. Thus, it can be concluded that the use of the LSTM method produces greater accuracy, precision, recall, and F1-Score values and can be interpreted the results of sentiment analysis with the LSTM method to produce predictions that are good enough to find out the sentiment of Flo application reviews on Google Playstore. The results are also in line with the results of the confusion matrix comparison, so it can also be interpreted that LSTM is superior to CNN with various test methods.

The results of the classification report analysis and confusion report analysis obtained do have differences, because both are evaluations of the algorithm models used and can be a comparison of the two algorithms, namely the CNN and LSTM algorithms.

Word Visualization of Each Class

Visualization of prediction results in seeing opinions or opinions contained in each class is shown after going through the *data testing stage*. The visualization of the prediction results can be seen on *wordcloud*. The following is presented the result of visualization of the prediction results in Figure 19:

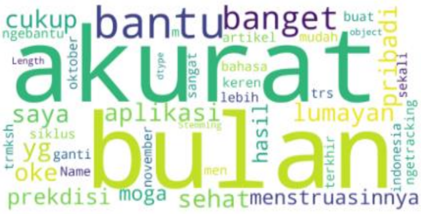


Figure 19. Wordcloud FLO Application

Furthermore, from the many reviews of the Flo application listed on the existing Google Play Store page, information that is considered the most important can be taken. The visualization for the entire data that most often appears in Figure 20 below.

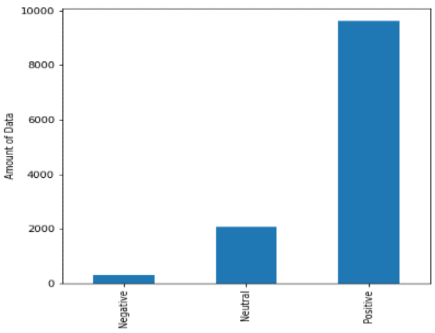


Figure 20. FLO App Review Diagram

In this study, the comparison was carried out starting with giving a rating predicate to each review. The author gives a negative rating predicate to reviews that have a rating from 1 to 2, a neutral rating predicate on reviews with a rating of 3, and a positive rating predicate on reviews that have a rating of 4 to 5. The following pie chart in Figure 21 shows the results of the rating comparison with the predictions depicted with the pie chart:

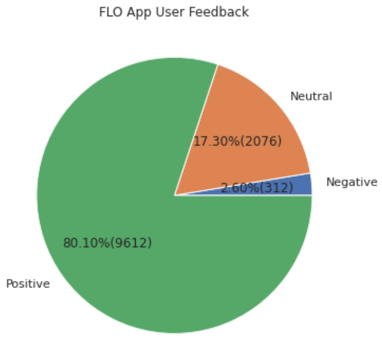


Figure 21. Pie Chart FLO App Review

From Figure 21 above, it can be shown that the number of positive reviews dominates with a percentage of 80.10% and a total of 9,612 reviews. Furthermore, in second place there are neutral reviews with a percentage of 17.30% and reviews as many as 2,076. While the number of negative reviews on the FLO application is very small with a percentage of only 2.60% and a total of 312 reviews.

While the prediction of positive: neutral: negative reviews obtained from each epoch 20, 50, and 100 consecutive from the CNN algorithm, namely 2193:173:34, 2208:173:19, and 2208:157:35. Then the LSTM algorithm was 1932:416:51, 1926:428:46, and 1984:374:42 reviews respectively.

CONCLUSION

Based on the research that has been done, sentiment analysis with the object of FLO application review research is 12000 review data selected based on country, namely Indonesia, more positive reviews followed by neutral reviews and finally negative reviews. It can be seen from the number of words used in *wordcloud*, on average using positive words compared to negative words. FLO application users, especially the Indonesian state, respond to this application to have a better impact, especially for women's needs. CNN and LSTM algorithms were used as comparisons in this study. The *training* data obtained is *accuracy and loss* by doing three epochs , namely 20, 50, 100 on the CNN and LSTM algorithms, which are good enough and not *overfitting*.

Furthermore, data testing was carried out using *confusion matrix* and *classification report*. *Confusion matrix* conducted on CNN and LSTM algorithms shows a comparison of CNN:LSTM accuracy (84.08%:92.67%), CNN:LSTM precision (66.02%:80.08%), and CNN:LSTM recall (51.15%:77.44%). *The Classification Report* conducted on CNN and LSTM algorithms shows a comparison of CNN:LSTM accuracy (84%:93%), *precision* CNN:LSTM (82%:92%), CNN:LSTM recall (84%: 93%), and *CNN:LSTM F1-Score* (81%:92%). Looking at the two algorithm comparisons, it can be seen that the superior one uses the LSTM algorithm with results above CNN.

This research is expected to be a follow-up study for those who want to use similar research. It is hoped that this research can be developed better in the future such as using more data, trying *epochs* with more variety, using other algorithms as comparisons in order to find the most accurate algorithm for sentiment analysis. Furthermore, for FLO application developers, seeing user reviews from Indonesia getting positive responses and in the future it is hoped that the FLO application can be further developed and can also use several languages, including Indonesian to better help users from Indonesia.

REFERENCES

Afif, M., Fawwaz, A., Ramadhani, K.N. and Sthevanie, F., 2021. Klasifikasi Ras pada Kucing menggunakan Algoritma Convolutional Neural Network(CNN). *Jurnal Tugas Akhir Fakultas Informatika*, 8(1), pp.715–730.

Al-adwan, A.S., 2020. What Makes Consumers Purchase Mobile Apps : Evidence from Jordan. *Journal of Theoretical and Applied Electronic Commerce Research*, 16(3), pp.562–583.

Budiman, M.R.F.I. and Abadi, F., 2023. Pemanfaatan Machine Learning Sebagai Sensor Berbasis Media Sosial Untuk Pemantauan Bencana Banjir Di Lahan Basah. *Prosiding Seminar Nasional Lingkungan Lahan Basah*, [online] 8(2), pp.94–100.

Fahmi, A., Ramadhan, I., Studi, P., Informasi, S., & Komputer, F. I. (2020). Analisis Sentiment Masyarakat Selama Bulan Ramadhan Dalam Menghadapi Pandemi COVID-19. Jurnal Informatika Dan Sistem Informasi, 1(2), 608–617.

García-ordás, M. T., Benítez-andrades, J. A., & García-rodríguez, I. (2020). Convolutional Neural Networks and Variational. Sensors, 20(4). https://doi.org/10.3390/s20041214

Kressbach, M., 2021. Period Hacks: Menstruating in the Big Data Paradigm. *Television and New Media*. https://doi.org/10.1177/1527476419886389.

Li, Z., Yang, W., Peng, S., & Liu, F. (2021). A Survey of Convolutional Neural Networks : Analysis , Applications , and Prospects. IEEE Transactions on Neural Networks and Learning Systems.

Medhat, W., Hassan, A. and Korashy, H., 2014. Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*. https://doi.org/10.1016/j.asej.2014.04.011.

Nelson, M. J., & Hoover, A. K. (2020). Notes on Using Google Colaboratory in AI Education. *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education*, 533–534.

Paputungan, C.K.N. and Jacobus, A., 2021. Sentiment Analysis of Social Media Users Using Long-Short Term Memory Method Analisis Sentimen Pengguna Sosial Media Menggunakan Metode Long Short Term Memory. *Jurnal Teknik Elektro dan Komputer*, 10(2), pp.99–106.

Purnasiwi, R.G., Kusrini and Hanafi, M., 2023. Analisis Sentimen Pada Review Produk Skincare Menggunakan Word Embedding dan Metode Long Short-Term Memory (LSTM). *Innovative: Journal Of Social Science Research,* 3(2), pp.11433–11448.

Varsamopoulos, S., & Bertels, K. (2019). Designing neural network based decoders for surface codes.

Wahyudi, D and Sibaroni, Y. (2022) ‘Deep Learning for Multi-Aspect Sentiment Analysis of TikTok App using the RNN-LSTM Method’, Building of Informatics, Technology and Science (BITS), 4(1), pp. 169-177.

Xu, G., Meng, Y., Qiu, X., Yu, Z. and Wu, X., 2019. Sentiment Analysis of Comment Texts Based on BiLSTM. *Ieee Access*, 7, pp.51522–51532. https://doi.org/10.1109/ACCESS.2019.2909919.10.47065/bits.v4i1.1665

Yamashita, R., Nishio, M., Do, R.K.G. and Togashi, K., 2018. *Convolutional neural networks: an overview and application in radiology*. *Insights into Imaging*. https://doi.org/10.1007/s13244-018-0639-9.

Yan, K., Arisandi, D., & Tony, T. (2022). Analisis Sentimen Komentar Netizen Twitter Terhadap Kesehatan Mental Masyarakat Indonesia. Jurnal Ilmu Komputer Dan Sistem Informasi, 10(1). https://doi.org/10.24912/jiksi.v10i1.17865

Zhao, J., Huang, F., Lv, J., Duan, Y., Qin, Z., Li, G. and Tian, G., 2020. Do RNN and LSTM have Long Memory ? *In International Conference on Machine Learning*, pp.11365–11375.