Sentiment Analysis of IMDB Movie Reviews Using Recurrent Neural Network Algorithm

Aryasuta[1], Fenina Adline Twince Tobing[2]
[1,2] Department of Informatics, Universitas Multimedia Nusantara, Tangerang, Indonesia
[1]aryasuta.saputra@student.umn.ac.id, [2]fenina.tobing@umn.ac.id

Accepted 06 June 2024
Approved 02 July 2024

***Abstract*— IMDb is a well-known platform that provides user reviews and ratings of various movies. The number of reviews found on IMDb is quite large, reaching thousands of reviews. Although a movie can have a high overall rating, it is still possible to receive negative reviews from some viewers. Therefore, the purpose of this sentiment classification system is to provide a benchmark for the level of sentiment contained in the movie, and hope that filmmakers can use this information as a reference in the development of their next movie. In this research, reviews from IMDb users are classified into two types, namely positive reviews and negative reviews. The program was created using the Python language with the LSTM (Long Short-Term Memory) classification model of the RNN (Recurrent Neural Network) algorithm. The purpose of using this algorithm is to measure the level of prediction accuracy in the classification process. The results of three test ratios, namely 60:40, 70:30, and 80:20, show that in the scenario of 80% data training and 20% data testing has better performance with the results accuracy of 96%, precision of 97%, recall of 98%, f1-score of 97%.**

***Index Terms*— Sentiment Analysis; IMDb; Python; Recurrent Neural Network**

I. Introduction

The increasing use of digital platforms is a result of the world's growing population and the changing environment. Digital platforms or more famously known as social media, are very often used to exchange opinions and share experiences about a product or service. People express their emotions directly or indirectly through language, facial expressions, gestures or writing [1]. Finally, these expressions are expressed on one of the platforms for reviewing a movie called IMDb.

Internet Movie Database (IMDb) is a website that provides a collection of information about movies, tv shows, and the cast involved in the movie or show. The majority of IMDb site users are people who want to find some information about movies based on other audience reviews [2]. Viewers who provide reviews about the movie will also provide a rating related to the movie that has been seen. Based on research that ratings and reviews by the audience can have a significant effect on film production [3]. The many forms of reviews that are scattered are sometimes very difficult for humans to distinguish a person's emotions that are actually poured out from text, speech, or facial expressions [4]. Therefore, this research will create a sentiment analysis program in order to find out someone's sentiment on the topic.

Sentiment analysis is a process that uses human language processing and computer language to extract, identify, and classify diverse opinions expressed in text format. It is one of the most important and interesting areas of research, as it can determine the success of a product from reviews and ratings on the internet [5]. Sentiment analysis is basically a classification problem that covers two fields, namely Natural Language Processing (NLP) and Machine Learning (ML). This sentiment analysis system is carried out to see a person's view whether the person's opinion shows a positive or negative side expressed towards a movie, product, and other things [6]. The level of sentiment analysis is divided into three levels, namely document level analysis, sentence level analysis, and entity and aspect level analysis. In reviews of a movie, the level of analysis used is entity and aspect level analysis, where the person's opinion will refer to only two sentiments between positive and negative [7].

There is similar research on sentiment analysis of IMDb movie reviews using the Support Vector Machine (SVM) algorithm from 5000 reviews data getting 79% accuracy, 75% precision, and 87% recall [11].

There is also other research that has been done with the recurrent neural network (RNN) algorithm to analyze the sentiment of traveloka application users as much as 5,000 data and divided by 2,500 testing data and 2,500 training data, getting an accuracy value of 87.42%, and evaluating the performance of the algorithm obtained recall 87.17%, precision 87.53%, and f-measure 87.34% [12].

On the other hand, there are researchers who use the RNN algorithm for sentiment analysis on social media called twitter with the data used as many as 1,500,000 tweets which are divided for testing by 20% and then 80% are used for training and obtain an accuracy of 80.39%, recall 83.57%, precision 78.56% [13].

Based on the background above, from previous research, the RNN algorithm has been well tested for sentiment classification. However, in this study, the algorithm will be tested using the IMDb movie reviews dataset to evaluate whether the results will be comparable to previous research. The tested data will be grouped into two categories, namely positive and negative.

II. THEORETICAL FOUNDATION

A. Sentiment Analysis

Sentiment analysis is an area of machine learning research that focuses on extracting information from textual reviews. The field of sentiment analysis is closely related to natural language programming and text mining. This analysis is used to determine the attitude of a reviewer towards various topics or the review as a whole [14]. Commenting sites have become a popular place to share emotional impressions through short texts. Emotions include happiness, sadness, anxiety, fear, and more. Gathering opinions from movie reviews can be difficult because human language is quite complex, leading to situations where positive words have negative connotations and vice versa [15].

B. Text Mining

Text mining is a method of finding patterns in unstructured text and is done automatically by a computer to find useful information for certain purposes [16]. Through the text mining process, it is possible to see how a person's opinion or opinion on a topic will be classified into two or more classes [17]. The process of this text will require a document preprocessing method, where this process will separate the whole text only to be analyzed in order to facilitate the sentiment classification process. There are several text mining methods that can handle problems, including classification, clustering, information extraction, information retrieval [18].

C. Classification Method

Classification method is a process that aims to develop a model or function that is able to understand and distinguish between concepts or classes that exist in unlabeled data [19]. In data classification, there are two stages that must be carried out, consisting of training using the dataset (training data) and testing using the data to be tested (testing data). The training data used is data that already has a class label. The difference between classification and clustering is that classification requires a data training process and requires data that already has a class label, while clustering does not require a class label because the class label already exists [20].

D. Algoritma Recurrent Neural Network

RNN is one type of neural network family in the deep learning category because the data is processed automatically without defining features [21]. This algorithm is applicable in Natural Language Processing (NLP) in the form of speech recognition, music synthesis, and text. The calculation of hidden state (St) and output (Ot) in the RNN algorithm for the tth step can be formulated as follows [22].

$$S_t = f(U_{xt} + W_{st-1})$$

$$O_t = softmaxV_{st}$$

E. Confusion Matrix

Confusion matrix serves to display the identification results between correctly predicted data sets and incorrectly predicted data, then the results will be compared with the actual facts [23]. The matrix calculation is done based on the true class and predicted class, with the basis as shown below.

		True Class	
		Positif	Negatif
Predicted Class	Positif	TP	FP
	Negatif	FN	TN

Fig 1. Confusion Matrix Classification

- TP: True Positive (number of correct predictions of positive classes)
- TN: True Negative (number of correct predictions of negative classes)
- FN: False Negative (original positive class predicted negative)
- FP: False Positive (original negative class, predicted positive)

1) Accuracy

This matrix is used to indicate the extent to which the model can correctly predict the class. Although this method is widely used, there are drawbacks in the interpretation (notion) of the results, especially when used on unbalanced data. Unbalanced data can lead to incorrect interpretations and needs to be addressed with caution. The matrix value is obtained from the following equation [24]:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

2) Precision

This indicator is used to measure the accuracy of the positive class prediction results with the true positive class. Although this indicator cannot clearly describe the number of correct predictions of the actual positive class, it gives an idea of how accurate the model is in identifying positive cases. The matrix value is obtained from the following equation [25]:

$$Precision = \frac{TP}{TP + FP}$$

3) Recall

This indicator aims to illustrate the extent to which the predicted positive class is correct with the actual positive class. However, this indicator does not explicitly describe how well the actual positive class prediction results are correct. The matrix value is obtained from the following equation [26]:

$$Recall = \frac{TP}{TP + FN}$$

4) F1-Score

This indicator aims to overcome the weakness in evaluating the performance of positive classes by taking into account precision and recall values in a balanced manner through the calculation of harmonic mean. The previous two indicators that focused only on positive classes meant that f-score could not provide a specific assessment of negative classes. However, all these drawbacks can be overcome by implementing a weighted version to consider all classes and their distributions. The metric value can be calculated using the following equation [27]:

$$f - score = 2 * \frac{precision * recall}{precision + recall} = 2 * \frac{2\,TP}{2\,TP + FP + FN}$$

III. METHODOLOGY AND SYSTEM DESIGN

The flow design of the recurrent neural network algorithm classification system for sentiment analysis of IMDb movie reviews will be described in the form of a flowchart, which consists of the main flowchart, preprocessing flowchart, and classification flowchart.

A. Main Flowchart

The main flowchart of the sentiment analysis system using recurrent neural network is shown in Figure 2. This diagram illustrates several stages performed in the system. The first stage is importing the library needed for this research. Next, import the dataset file in csv format. After that, the preprocessing stage is carried out to manage text data into structured sentences that can be processed by the program. The preprocessing process involves several steps, such as text cleaning, case adjustment, tokenization, removal of stopwords, and stemming. After the preprocessing stage is complete, sentiment labeling is performed on each review, which consists of two labels, namely positive and negative. The sentiment labeling process is done automatically using the Text Blob library. The last stage is classification using the Recurrent Neural Network algorithm with the LSTM (Long short-term memory) model. This model is used to predict the probability of each classification on reviews that have been labeled with sentiment before.

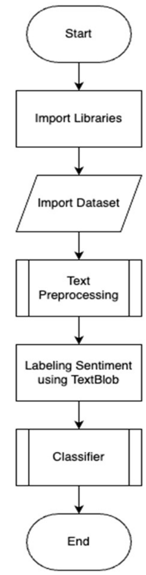

Fig 2. *Main Flowchart*

B. Preprocessing Flowchart

Figure 3 illustrates the preprocessing stage used to convert raw data collected from various sources into more structured information. The purpose of this process is to prepare the data so that it can be processed more efficiently and accurately in the next processing stage. The preprocessing process itself requires several steps, including the following:

1) Cleaning Text

This is a common process that is required during data processing, which is the removal of unnecessary symbols and punctuation marks in text data. This process includes filtering where non-letters or numbers are replaced with spaces. Then it removes excess spaces due to the replacement of special characters or numbers that have become spaces, and removes words that only consist of 1 or 2 characters, for example "the", "a", "is".

2) Case Folding

This is the stage of equalizing the size of letters from the letters "a" to "z", for example at the beginning of a normal word starting with a capital letter, it will be changed to lowercase letters for all words.

3) Tokenizing

Tokenization is the process of breaking sentences into individual word units. So the review sentences will be separated by the comma delimiter ",". An example of the tokenizing process, "Absolutely loved this film" becomes the tokens "Absolutely", "loved", "this", "film".

4) Stop words

This process is a step to remove common words that are considered irrelevant or known as stop words. Examples of these stop words include "the", "a", "an", "in", "of", and so on. This process is intended to improve the effectiveness and quality of text analysis or modeling. The process of removing stop words is generally done after the tokenizing stage, where words have been separated into tokens which are then removed. This aims to make the words processed in the classification stage more relevant and meaningful [28].

5) Stemming

It is a step in text processing that aims to convert words into their base form or root word. For example, words like "play", "playing", and "played" will be converted into the same base form, namely "play", in the stemming process. This is done to reduce the variety of words that have the same root.

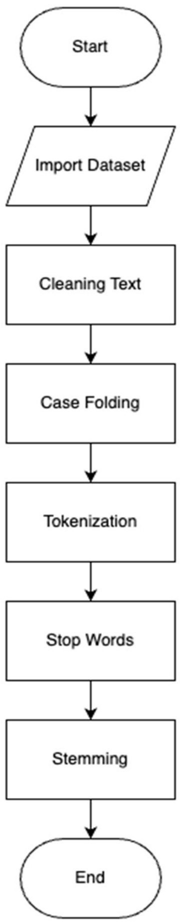

Fig 3. *Flowchart Preprocessing*

C. Classification Flowchart

Figure 4 shows the flowchart for classification using the Recurrent Neural Network algorithm. After the data set is processed at the preprocessing stage, the data is divided into two parts, namely training data and test data. There are three scenarios in separating training data and testing data, namely using a 60:40 ratio, a 70:30 ratio, and an 80:20 ratio. The train data is trained with the LSTM model of the RNN algorithm which will make predictions on the test data and match the X test and y test results to get the accuracy, precision, recall and f1-score values of the confusion matrix table model clustering.

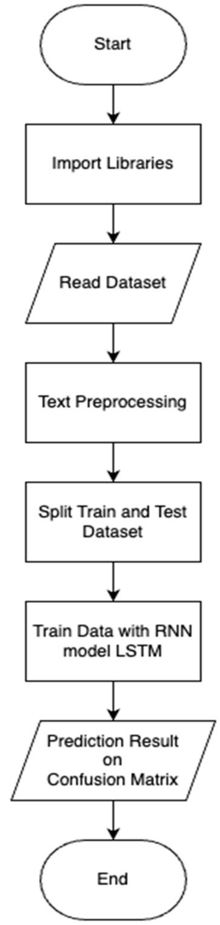

Fig 4. *Flowchart* Klasifikasi

IV. Result and Discussion

A. System Implementation

The implementation process in this research uses the Python programming language version 3.10.8. The data used are movie reviews from various genres and countries that come from an information platform called IMDb. The reviews used are 320,000 reviews data which are divided into two categories, namely positive and negative. The first process in this system will be data preprocessing, which begins with cleaning text.

```
'!"#$%&\'()*+,-./:;<=>?@[\\]^_`{|}~'
```

Fig 5. Punctuation

Figure 6 is the result of the cleaning text process created into a new column. The process removes some symbols or special characters as shown in Figure 5. The new column aims to show the results of the program work by displaying the comparison after and before the cleaning process.

	review	label	cleaning
0	"Yaara Sili Sili Virah Ki Raat Ka Jalna"'Lekin...	8	yaara sili sili virah raat jalna lekin movie ...
1	Gulzar is at his best when he is telling such ...	9	gulzar his best when telling such intriiguing ...
2	I was completely mesmerized by Lekin and espec...	9	was completely mesmerized lekin and especiall...
3	Greatly enjoyed the development of the story l...	9	greatly enjoyed the development the story line...
4	The lines of time are very blurry. Past, prese...	10	the lines time are very blurry past present an...
5	This is a superb storyline and has excellent m...	10	this superb storyline and has excellent music ...
6	Criticizing is very easy. But when we see the ...	9	criticizing very easy but when see the thing w...
7	Man, where do I start? All three actresses ma...	10	man where start all three actresses make this ...
8	I'm trying to watch this movie and it would no...	2	trying watch this movie and would not play
9	Randsell Pearson's fact-based book provides th...	5	randsell pearson fact based book provides the ...

Fig 6. Cleaning Text

Figure 7 displays the results of the tokenizing process, which is a stage where the sentence will be separated into word units using a delimiter in the form of "," (comma). The sentence that will be used in the tokenizing process is taken from the cleaning column. After performing the process, the results are then entered into a new column named Tokenization.

cleaning	Tokenization
yaara sili sili virah raat jalna lekin movie ...	[, yaara, sili, sili, virah, raat, jalna, leki...
gulzar his best when telling such intriiguing ...	[gulzar, his, best, when, telling, such, intri...
was completely mesmerized lekin and especiall...	[, was, completely, mesmerized, lekin, and, es...
greatly enjoyed the development the story line...	[greatly, enjoyed, the, development, the, stor...
the lines time are very blurry past present an...	[the, lines, time, are, very, blurry, past, pr...
this superb storyline and has excellent music ...	[this, superb, storyline, and, has, excellent,...
criticizing very easy but when see the thing w...	[criticizing, very, easy, but, when, see, the,...
man where start all three actresses make this ...	[man, where, start, all, three, actresses, mak...
trying watch this movie and would not play	[, trying, watch, this, movie, and, would, not...
randsell pearson fact based book provides the ...	[randsell, pearson, fact, based, book, provide...

Fig 7. Tokenization

Figure 8 shows the result of the process of removing irrelevant words, commonly referred to as stopwords. This stopword process utilizes the NTLK (Natural Language Toolkit) library to assist in the removal of conjunctions such as "the", "was", "were", and other similar words. This stage aims to improve the performance of the analysis by removing common words that usually do not provide significant sentence changes. The words to be processed are taken from the Tokenization column. After the process, the results are then entered into a new column to verify the program's success in carrying out the process.

Tokenization	Stopwords
[, yaara, sili, sili, virah, raat, jalna, leki...	[, yaara, sili, sili, virah, raat, jalna, leki...
[gulzar, his, best, when, telling, such, intri...	[gulzar, best, telling, intriiguing, story, ea...
[, was, completely, mesmerized, lekin, and, es...	[, completely, mesmerized, lekin, especially, ...
[greatly, enjoyed, the, development, the, stor...	[greatly, enjoyed, development, story, line, m...
[the, lines, time, are, very, blurry, past, pr...	[lines, time, blurry, past, present, future, m...
[this, superb, storyline, and, has, excellent,...	[superb, storyline, excellent, music, set, bac...
[criticizing, very, easy, but, when, see, the,...	[criticizing, easy, see, thing, clarity, maybe...
[man, where, start, all, three, actresses, mak...	[man, start, three, actresses, make, movie, ge...
[, trying, watch, this, movie, and, would, not...	[, trying, watch, movie, would, play]
[randsell, pearson, fact, based, book, provide...	[randsell, pearson, fact, based, book, provide...

Fig 8. Stopwords

Figure 9 displays the results of the stemming process that has been entered into a new column. The stemming process is a process that will change from colloquial words to basic words, for example "directed: direct", "responsibility: response". The stemming process again uses the NTLK library because this library is one that is quite popular for use in natural language processing provided by Python. The words that will be processed in the stemming stage are taken from the results of the stopwords process.

Stopwords	Stemming
[, yaara, sili, sili, virah, raat, jalna, leki...	[, yaara, sili, sili, virah, raat, jalna, leki...
[gulzar, best, telling, intriiguing, story, ea...	[gulzar, best, tell, intriigu, stori, eas, per...
[, completely, mesmerized, lekin, especially, ...	[, complet, mesmer, lekin, especi, castl, dimp...
[greatly, enjoyed, development, story, line, m...	[greatli, enjoy, develop, stori, line, music, ...
[lines, time, blurry, past, present, future, m...	[line, time, blurri, past, present, futur, mer...
[superb, storyline, excellent, music, set, bac...	[superb, storylin, excel, music, set, backgrou...
[criticizing, easy, see, thing, clarity, maybe...	[critic, easi, see, thing, clariti, mayb, thin...
[man, start, three, actresses, make, movie, ge...	[man, start, three, actress, make, movi, gem, ...
[, trying, watch, movie, would, play]	[, tri, watch, movi, would, play]
[randsell, pearson, fact, based, book, provide...	[randsel, pearson, fact, base, book, provid, b...

Fig 9. Stemming

Figure 10 displays the results of the sentence return process after going through several processing stages. In the first stage, the numbers in the sentence are removed, then punctuation marks or special characters are removed in the second stage, and whitespace is removed in the third stage. Next, the words that have been tidied up are put into a variable named "split" to be made into a sentence like the beginning in the fourth stage. In the last stage, the text is returned after running a series of previous processing operations.

Stemming
yaara sili sili virah raat jalna lekin movi be...
gulzar best tell intriigu stori eas perfect di...
complet mesmer lekin especi castl dimpl haunt ...
greatli enjoy develop stori line music least a...
line time blurri past present futur merg one a...

Fig 10. Sentence Return Process

Figure 11 is the review labeling process. Label processing will use subjectivity and polarity calculations from the TextBlob library. The processing results will be classified into two parts, for positive reviews marked with the number one (1) and negative reviews marked with the number zero (0). The determination of positive and negative comes from the polarity value, where if the value gets a number above 0 it will be categorized as positive, and if the polarity value gets a value below 0 or equal to 0 it will be categorized as negative. The categorization results are entered into a new column called Analysis.

After running all these processes, the last stage in the program will be modeling. This process includes converting tokens to numeric or numbers which will then be able to proceed to the modeling process using the LSTM model of RNN. In this process, the data will be formed into 1000 tokens, after which the 'Stemming' column will create an internal tokenizer dictionary that

ISSN 2085-4552

will map the words into numeric form. Furthermore, the LSTM(100) code will be used to determine the level of complexity and memory capacity. Sigmoid activation is useful for generating binary values for positive class probability results. This modeling will be run into several experimental scenarios.

Stemming	labelReview	subjectivity	polarity	Analysis
yaara sili sili virah raat jalna lekin movi be...	1	0.371296	0.051852	1
gulzar best tell intriigu stori eas perfect di...	1	0.557143	0.828571	1
complet mesmer lekin especi castl dimpl haunt ...	1	0.483009	0.324784	1
greatli enjoy develop stori line music least a...	1	0.486667	0.170000	1
line time blurri past present futur merg one a...	1	0.350000	0.050000	1
...	...	...	...	...
mike leigh work sever actor appear mani film j...	1	0.430556	0.063889	1
say previou fan movi said yet think mike leigh...	1	0.330000	0.255000	1
anoth one movi love much first time saw cri th...	1	0.294444	0.187778	1
mike leigh treat anoth masterpiec life sweet s...	1	0.550000	0.425000	1
one film subject sublim honestli portray peopl...	1	0.529762	0.161540	1

Fig 11. Labeling

B. *Testing*

In this section, the test results based on the three predefined scenarios will be discussed. After the test, the scenarios will be evaluated to see which scenario can provide the best classification performance.

1) *Test Scenario*

In this system, there are three scenarios that are divided based on the ratio comparison. The first scenario is 60:40, where 60% of the data is used for training and 40% of the data is used for testing. The second scenario is 70:30, where 70% of the data is used for training and 30% of the data is used for testing. While the third scenario is 80:20, with 80% of the data used for training and 20% of the data used for testing. These three scenarios are based on previous research in the division of training and testing data in sentiment analysis. While there are other variations in data sharing scenarios that may be used, these three scenarios are common choices and have been tested in the literature. All scenarios will use a dataset consisting of 320,747 total data, with 257,763 positively labeled data and 62,984 negatively labeled data. The training process is conducted using the epochs method, where the training data is shared and learned by the system. After learning, the system applies its learning results to the testing data to test the sentiment prediction capability.

2) *Test Results*

The system test results are visualized in the form of a confusion matrix table. The following is a display of the test results in the previous process, which will be evaluated to obtain accuracy, precision, recall and f1-score values.

Table 1 is the result of testing conducted on a ratio of 60% training data and 40% testing data formed into a confusion matrix.

Table 2 is the result of testing conducted on a ratio of 70% training data and 30% testing data formed into a confusion matrix.

TABLE I. CONFUSION MATRIX 60:40

Predicted Value	True Value	
	Positive	Negative
Positive	100734	2278
Negative	3203	22084

TABLE II. CONFUSION MATRIX 70:30

Predicted Value	True Value	
	Positive	Negative
Positive	73953	3235
Negative	1366	17617

Table 3 is the result of testing conducted on a ratio of 80% training data and 20% testing data formed into a confusion matrix.

TABLE III. Confusion Matrix 80:20

Predicted Value	True Value	
	Positive	Negative
Positive	50140	1413
Negative	1196	11401

3) *Evaluation Results*

The evaluation process will be carried out by manual calculation of each confusion matrix table, aiming to obtain and ensure the highest value of accuracy, precision, recall and f1-score. The following is a description of the manual calculation to clarify the results of the values of accuracy, precision, recall, and f1-score in the scenario of 60% training data and 40% testing data.

$$Accuracy \;= \frac{TP + TN}{TP + TN + FP + FN}$$

$$= \frac{100734 + 22084}{100734 + 3203 + 2278 + 22084}$$

$$= \frac{122818}{128299}$$

$$= 0.9572794$$

$$Precision \;= \frac{TP}{TP + FN}$$

$$= \frac{100734}{100734 + 2278}$$

$$= \frac{100734}{103012}$$

$$= 0.97788607$$

$$Recall = \frac{TP}{TP + FP}$$

$$= \frac{100734}{100734 + 3203}$$

$$= \frac{100734}{103937}$$

$$= 0.9691832$$

$$F1 - Score = \frac{2\,TP}{2\,TP + FP + FN}$$

$$= \frac{201468}{201468 + 3203 + 2278}$$

$$= \frac{201468}{206949}$$

$$= 0.97351521$$

The following is a description of manual calculations to clarify the results of the values of accuracy, precision, recall, and f1-score in the scenario of 70% training data and 30% testing data.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$= \frac{73953 + 17671}{73953 + 1366 + 3235 + 17671}$$

$$= \frac{91624}{96225}$$

$$= 0.95218498$$

$$Precision = \frac{TP}{TP + FN}$$

$$= \frac{73953}{73953 + 3235}$$

$$= \frac{73953}{77188}$$

$$= 0.95808934$$

$$Recall = \frac{TP}{TP + FP}$$

$$= \frac{73953}{73953 + 1366}$$

$$= \frac{73953}{75319}$$

$$= 0.98186381$$

$$F1 - Score = \frac{2\,TP}{2\,TP + FP + FN}$$

$$= \frac{147906}{147906 + 1366 + 3235}$$

$$= \frac{147906}{152507}$$

$$= 0.96983089$$

The following is a description of manual calculations to clarify the results of the values of accuracy, precision, recall, and f1-score in the scenario of 80% training data and 20% testing data.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$= \frac{50140 + 11401}{50140 + 1196 + 1413 + 11401}$$

$$= \frac{61541}{64150}$$

$$= 0.95932969$$

$$Precision = \frac{TP}{TP + FN}$$

$$= \frac{50140}{100734 + 1413}$$

$$= \frac{50140}{51553}$$

$$= 0.97259131$$

$$Recall = \frac{TP}{TP + FP}$$

$$= \frac{50140}{50140 + 1196}$$

$$= \frac{50140}{51336}$$

$$= 0.97670251$$

$$F1 - Score = \frac{2\,TP}{2\,TP + FP + FN}$$

$$= \frac{100280}{100280 + 1196 + 1413}$$

$$= \frac{100280}{102889}$$

$$= 0.97464258$$

The following is a table created from the results of the previous manual calculations for each scenario. By using this table, it can easily provide an overview of the performance of each scenario based on the resulting percentage value. The percentage values include accuracy, precision, recall, and f1-score.

TABLE IV. Scenario Comparison of 60:40, 70:30, 80:20 Ratio

Matrix (%)	60:40		70:30		80:20	
	Positive	Negative	Positive	Negative	Positive	Negative
Accuracy	96%		95%		96%	
Precision	98%	87%	96%	93%	97%	91%
Recall	97%	91%	98%	85%	98%	89%
F1-Score	97%	89%	97%	88%	97%	90%

In Table 4, it can be seen that the 60:40 and 80:20 ratios have the same percentage value due to rounding results. However, there is a slight difference between the 60:40 ratio value in the manual accuracy calculation and the 80:20 ratio value in the manual accuracy calculation. The difference shows that in the scenario with the 80:20 ratio, there is a 0.002 higher increase in the accuracy value compared to the 60:40 ratio.

V. Conclusion

Based on the test results that have been carried out, it can be concluded that the implementation of the Recurrent Neural Network algorithm with the Long Short Term Memory model in the process of classifying movie reviews from the IMDb site has been successful. Tests were conducted using movie review data from the IMDb site with a total of 320,747 reviews. Tests were carried out using several scenarios that have different ratios, namely 60:40, 70:30, and 80:20. The test results show that the scenario with a ratio of 80% training and 20% testing provides higher performance compared to other scenarios, with an accuracy of 96%, precision of 97%, recall of 98%, and f1-score of 97%.

In addition, this study also shows that the division of training data and testing data has a significant influence on accuracy results. The more data used in the training process, the better the accuracy results. Of the total 320,747 data used, 80.36% had positive sentiments, while 19.63% had negative sentiments. Sentiment analysis on IMDb movie reviews shows that positive sentiment has a higher percentage than negative sentiment.

References

[1] E. T. L. Joang Ipmawati, Kusrini, "Komparasi Teknik Klasifikasi Teks Mining Pada Analisis Sentimen," Indonesian Journal on Networking and Security, vol. 6, no. 1, pp. 28–36, 2017.

[2] S. D. A. Y. P. I. A. S. Gita Cahyani, Wiwi Widayani, "Klasifikasi Data Review IMDb Berdasarkan Analisis Sentimen Menggunakan Algoritma Support Vector Machine," JURNAL MEDIA INFORMATIKA BUDIDARMA, vol. 6, no. 3, 2022.

[3] M. A. F. Faisal Rahutomo, Pramana Yoga Saputra, "IMPLEMENTASI TWITTER SENTIMENT ANALYSIS UNTUK REVIEW FILM MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE," JURNAL INFORMATIKA POLINEMA, vol. 4, no. 2, 2018.

[4] J. R. R. A. Kashfia Sailunaz, Manmeet Dhaliwal, "Emotion detection from text and speech: a survey," Social Network Analysis and Mining, vol. 8, no. 28, 2018.

[5] S. Z. M. Md. Rakibul Haque, Salma Akter Lima, "Performance Analysis of Different Neural Networks for Sentiment Analysis on IMDb Movie Reviews," 2020.

[6] D. Oman Somantri, "Analisis Sentimen Penilaian Tempat Tujuan Wisata Kota Tegal Berbasis Text Mining," JEPIN (Jurnal Edukasi dan Penelitian Informatika), vol. 5, no. 2, pp. 191–196, 2019.

[7] N. J. Venkateswarlu Bonta, Nandhini Kumaresh, "A Comprehensive Study on Lexicon Based Approaches for Sentiment Analysis ," Asian Journal of Computer Science and Technology, vol. 8, no. S2, pp. 1–6, 2019.

[8] I. Y. B. Rosit Sanusi, Femi Dwi Astuti, "ANALISIS SENTIMEN PADA TWITTER TERHADAP PROGRAM KARTU PRA KERJA DENGAN RECURRENT NEURAL NETWORK," JIKO (Jurnal Informatika dan Komputer), vol. 5, no. 2, pp. 89–99, 2018.

[9] F. K. Muhamad Rizal Firmansyah, Ridwan Ilyas, "Klasifikasi Kalimat Ilmiah Menggunakan Recurrent Neural Network," Industrial Research Workshop and National Seminar, pp. 488–495, 2020.

[10] Y. C. O. B. Mohamed Chiny, Marouane Chihab, "LSTM, VADER and TF-IDF based Hybrid Sentiment Analysis Model," International Journal of Advanced Computer Science and Applications, vol. 12, no. 7, pp. 1097–1099, 2021.

[11] T. I. R. Nur Ghaniaviyanto Ramadhan, "Analysis Sentiment based on IMDB aspects from movie reviews using SVM)," Jurnal dan Penelitian Teknik Informatika, pp. 39–45, 2022.

[12] A. S. Lilis Kurniasari1, "Sentiment Analysis using Recurrent Neural Network," Journal of Physics: Conference Series, pp. 1–6, 2020.

[13] D. M. D. B. O. R. L. N. Sergiu Cosmin Nistor, Mircea Moca, "Building a Twitter Sentiment Analysis System with Recurrent Neural Networks," Journal of Physics: Conference Series, pp. 1–24, 2021.

[14] C. X. M. A. T. M. Zeeshan Shaukat, Abdul Ahad Zulfiqar, "Sentiment analysis on IMDB using lexicon and neural networks," 2020.

[15] S. M. Qaisar, "Sentiment Analysis on IMDb Movie Reviews Using Hybrid Feature Extraction Method," International Journal of Interactive Multimedia and Artificial Intelligence, vol. 5, no. 5, pp. 109–114, 2019.

[16] A. H. Fira Fathonah, "Penerapan Text Mining Analisis Sentimen Mengenai Vaksin Covid - 19 Menggunakan Metode Na¨ıve Bayes," Jurnal Sains dan Informatika, vol. 7, no. 2, pp. 155–164, 2021.

[17] M. A. Nengah Widya Utami, "TEXT MINING DALAM ANALISIS SENTIMEN PEMBELAJARAN DARING DI MASA PANDEMI COVID 19 MENGGUNAKAN

ALGORITMA K-NEAREST NEIGHBOR," JINTEKS (Jurnal Informatika Teknologi dan Sains), vol. 4, no. 2, pp. 140–148, 2022.

[18] A. F. r. O. y. P. a. De di Darwis, Eka Shintya Pratiwi, "PENERAPAN ALGORITMA SVM UNTUK ANALISIS SENTIMEN PADA DATA TWITTER KOMISI PEMBERANTASAN KORUPSI REPUBLIK INDONESIA," Jurnal Ilmiah Edutic, vol. 7, no. 1, pp. 1–11, 2020.

[19] A. R. Febie Elfaladonna, "ANALISA METODE CLASSIFICATION DECISSION TREE DAN ALGORITMA C.45 UNTUK MEMPREDIKSI PENYAKIT DIABETES DENGAN MENGGUNAKAN APLIKASI RAPID MINER," SINTECH (Science and Information Technology) Journal, vol. 2, pp. 10–17, 2019.

[20] S. M. Qaisar, "Sentiment Analysis of IMDb Movie Reviews Using Long Short-Term Memory," International Conference on Computer and Information Sciences, pp. 1–4, 2020.

[21] A. M. Gonzalo A. Ruz, Pablo A. Henr´ıquez, "Sentiment analysis of Twitter data during critical events through Bayesian networks classifiers," Industrial Research Workshop and National Seminar, vol. 116, pp. 92–104, 2020.

[22] H. Utami, "Analisis Sentimen dari Aplikasi Shopee Indonesia Menggunakan Metode Recurrent Neural Network," Indonesian Journal of Applied Statistics, vol. 3, no. 1, pp. 31–38, 2022.

[23] F. H. R. Imamah, "Twitter Sentiment Analysis of Covid-19 Using Term Weighting TF-IDF And Logistic Regresion," Information Technology International Seminar (ITIS), vol. 7, no. 1, pp. 1–17, 2020.

[24] H. S. Gientry Rachma Ditami, Eva Faja Ripanti, "Implementasi Support Vector Machine untuk Analisis Sentimen Terhadap Pengaruh Program Promosi Event Belanja pada Marketplace," (JEPIN)Jurnal Edukasi dan Penelitian Informatika, vol. 8, no. 3, pp. 508–516, 2022.

[25] L. S. Merinda Lestandy, Abdurrahim Abdurrahim, "Analisis Sentimen Tweet Vaksin COVID-19 Menggunakan Recurrent Neural Network dan Na¨ıve Bayes," JURNAL RESTI(Rekayasa Sistem dan Teknologi Informasi), vol. 5, no. 2, pp. 802–808, 2019.

[26] M. S. Michael Suhendra, Windra Swastika, "ANALISIS SENTIMEN PADA ULASAN APLIKASI VIDEO CONFERENCE MENGGUNAKAN NA¨IVE BAYES," Jurnal Ilmiah Sains Teknologi, vol. 2, no. 1, 2021.

[27] F. H. Hilda Nuraliza1, Oktariani Nurul Pratiwi, "Analisis Sentimen IMBd Film Review Dataset Menggunakan Support Vector Machine (SVM) dan Seleksi Feature Importance," Jurnal Mirai Manajemen, vol. 7, no. 1, pp. 1–17, 2022.

[28] M. N. F. M. Ulil Albab, Yohana Karuniawati P., "Optimization of the Stemming Technique on Text Preprocessing President 3 Periods Topic," Jurnal TRANSFORMATIKA, vol. 20, no. 2, pp. 1–10, 2023.

[29] P. P. K. H. R. Prof. Praveen Gujjar J, "Sentiment Analysis:Textblob For Decision Making," International Journal of Scientific Research Engineering Trends, vol. 7, no. 2, pp. 1097–1099, 2021.