

# The Effect of Feature Weighting on Sentiment Analysis TikTok Application Using The RNN Classification

Rizki Nabil Aufa<sup>1\*</sup>, Sri Suryani Prasetyowati<sup>1</sup>, Yuliant Sibaroni<sup>2</sup>

<sup>1</sup>Informatics, Informatics, Telkom University, Bandung, Indonesia

<sup>2</sup>Faculty, Research Program, Name of Institution, City, Indonesia

Email: <sup>1\*</sup>rizkinabil@student.telkomuniversity.ac.id, <sup>2</sup>srisuryani@telkomuniversity.ac.id, <sup>3</sup>yuliant@telkomuniversity.ac.id

Correspondence Author Email: rizkinabil@student.telkomuniversity.ac.id

Submitted: 08/06/2023; Accepted: 29/06/2023; Published: 29/06/2023

**Abstract**—Social media is a medium used by people to express their opinions. In its development, social media has become a necessity in social life. One of the most popular social media applications since 2020 is TikTok. Short videos with an average duration of 60 seconds can entertain the community so that they don't feel isolated. There are 17 million TikTok application reviews in the Google Play store in Indonesia from various user ages. The rapid development of information and technology has led to the pros and cons of this application. Freedom of expression without specific restrictions on content publication negatively impacts the user's mentality. Based on this, sentiment analysis is very important to reveal trends in opinions about applications that are useful for the community in increasing awareness of whether the application is good before use. Proper feature weighting is required to improve the sentiment analysis results' accuracy. More optimal results can be obtained by determining the appropriate weight for different feature weighting. This study compares the TF IDF, TF RF, and Word2Vec feature weighting methods with the RNN classifier on the TikTok app review. The experiment shows that TF RF is superior to TF IDF, with successive feature weighting accuracy with TF RF of 87,6%, TF IDF of 86%, and Word2Vec of 80%. The contribution of this research lies in its exploration of different feature weighting methods to enhance sentiment analysis accuracy and provide valuable insights for decision-making processes.

**Keywords:** Sentiment Analysis; TikTok; TF-IDF; TF-RF; Word2vec; RNN

## 1. INTRODUCTION

Social media is a computer application designed to facilitate communication with others without having to meet face-to-face and as a means of having fun to reduce feelings of isolation[1]. Along with technological advances, social media has become a social need everyone must fulfill. One of the social media applications that has become popular since 2020 is TikTok. That year, the world's population was struck by a rapidly spreading and highly dangerous virus known as COVID-19. As a result, many countries implemented their respective isolation policies, enforcing lockdown measures restricting people from engaging in activities outside their homes. This included working at offices, attending schools, conducting business operations, and even limiting access to holy places for prayer. Of course, the TikTok application has a significant role during these difficult times. This pandemic has caused TikTok to experience a significant increase in popularity or a sizable spike[2].

The TikTok app is a viral technological phenomenon that Zhang Yiminy from China created. TikTok allows users to create 60-second videos and add various features such as music, sound effects, filters, stickers, and others[3]. TikTok encourages user creativity in creating entertaining videos.

According to market research on mobile sensor applications in 2021, the number of active users of TikTok social media reached 65.2 million downloads, equivalent to a 21.4% increase from the same period in the previous year[4]. TikTok app users in Indonesia increased from 2017 to 2020, reaching 315 million users from Q3 2019 to Q1 2020[5]. TikTok users in Indonesia have expanded to millennials, Gen Z, and minors[6]. Research[7] shows that as of May 24, 2023, Indonesia is the second largest TikTok app user country after the United States, with 112.97 million users. This number is only 3.52 million users, different from the total TikTok users in the United States.

Based on this data, there are many reviews of the TikTok app. The flexibility in expressing oneself through short videos and the rapid dissemination of information creates pros and cons. In its development, this application continuously evaluates restrictions on the type of content created by its users to reduce negativity and influence a person's mentality. Underage TikTok users tend to be at risk of adverse impacts such as bullying and narcissism. Regarding business, TikTok always improvises to become a new market for its users to sell their products. Data and widespread opinions in Indonesian society regarding the TikTok application need to be evaluated so that users can find out whether this application is good or not before they install it. In addition, it can provide significant benefits for the future development of the TikTok application.

Conducting application reviews manual is a time-consuming task. At the same time, sentiment analysis can speed up review assessment as it is processed with natural language processing through machine learning, making it more efficient. Therefore, the role of sentiment analysis is needed. Sentiment analysis is a branch of Text Mining used to analyze text opinions[8].

In sentiment analysis, using feature weighting is very important in the process. Feature weighting can maximize the performance and accuracy of a classification model. The feature weighting study was carried out by[9], using the TF IDF method, the TF RF method, and the Weighted Inverse Document Frequency (WIDF). This study implements the word weighting method with the Naïve Bayes classification approach. A comparison of the three methods has a fairly good accuracy value. The TF RF feature weighting method produces better accuracy than the other two methods,

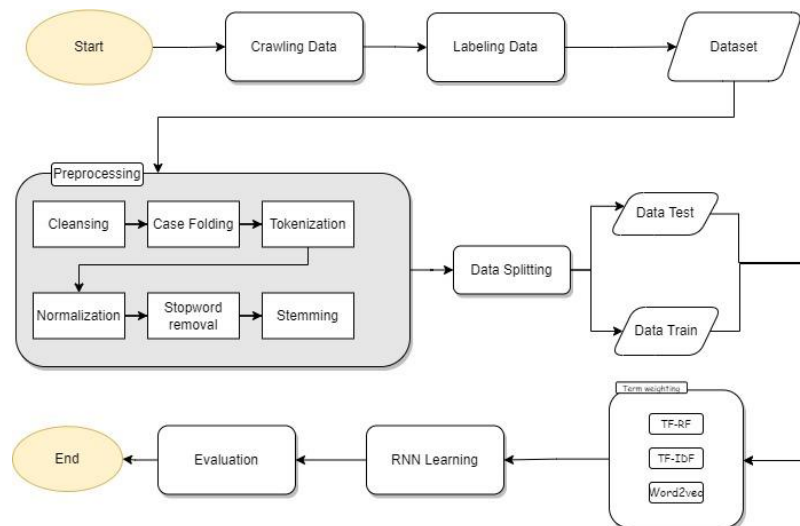
with an accuracy of 98,67%. Research[10] examines the effect of the Word2vec and TF IDF weighting features on the accuracy of the RNN classification model used for sentiment analysis of the COVID-19 vaccine. The results showed that the Word2vec weighting resulted in an accuracy of 53%, while an accuracy of 51% with the TF IDF weighting method.

Wahyudi et al[11] have carried out a sentiment analysis on a classification based on reviewing aspects of the Tiktok application using the RNN-LSTM method. The research was carried out by adding the word feature embed BERT in the pre-training model. The labels used for assessing three aspects are feature aspects, business aspects, and content aspects. The highest accuracy results are 95% on the assessment of business aspects. Word weighting in Text Mining aims to provide value/weight to the terms contained in a document[9]. The results of the study[12] show that the accuracy of the TF IDF weighting in Sentiment Analysis of the New State Capital of Indonesia is 88,8%. In comparison, research[13] proves that the TF IDF method still has better accuracy results than TF RF. While research[10], which used the Word2Vec feature extraction in the RNN classification, produced significant numbers in increasing the model's accuracy. No combination makes performance and accuracy results from the use of three sentiment classes in RNN classification has yet to be done.

Based on several literature reviews, this research analyzes three categories of opinions spread regarding TikTok app reviews in Indonesia, whether the sentiment is positive, neutral, or negative. This research also compares the performance of TF IDF, TF RF, and Word2Vec feature weighting methods on RNN classification for TikTok app sentiment analysis. The three weighting methods are compared to determine the best performance, and has never been done in previous studies.

## 2. RESEARCH METHODOLOGY

### 2.1 Architecture System



**Figure 1.** Flowchart of the Architecture build system

Figure 1 shows the process of comparing the three feature weighting methods. It starts with crawling the data, then automatically labeling using a few lines of Python code and manually being validated by humans. After labeling, we preprocess the data to make it more manageable for the model. Then, the dataset undergoes the feature weighting stage, where we validate it using k-fold cross-validation and compares the three weighting methods. The weighted data enters the RNN Classification stage, which learns the data into positive, neutral, or negative sentiments. Then, the last stage is the evaluation of each experiment.

### 2.2 Data Crawling

The data for this study were obtained by crawling 5,000 reviews of the TikTok application, from April 5 to April 15, 2023, on the Google Play Store website. The data crawling process is performed using Python and the Google-play scraper library, with an API service specifically designed to dump data from the Google Play Store.

### 2.3 Data Labelling

The results of the previous data collection were then labeled manually. The data were grouped into three labels/classes: positive, neutral, and negative.



**Table 1.** Data Labelling

Review	Label
Tolong ya tiktok apk terbanyak yg didownload manusia, sebagai manusiawi yg skrg tiktok jangan membuat konten atau live yg TDK senonoh, krna tiktok skrg di gunaiin dari umur yg 18 kebawah sudah tau, tolong di banned kalo live yg begitu sangat sangat TDK senonoh jatuh apk ini klo membiarkan mereka membuat konten bebas mengandung xxx hanya utk give aja, lebih diperhatikan ya tiktok please Bagus sih cuma bintang 3 aja dulu soalnya filter AI manga gak bisa di pake	Negative
Belanja di TIK TOK Alhamdulillah banyak promonya ,barang" nya juga bagus" kurirnya ramah..the best buat Tik Tok	Neutral
	Positive

Table 1 shows that labeling is done with two validation stages, automatically and manually. The automatic labeling stage involves automating a Python script code to evaluate the rating column of the dataset. Ratings below 3 are categorized as negative, a rating of 3 is considered neutral, and ratings above 3 are classified as positive. After the automatic labeling stage, the manual labeling stage is conducted to revalidate the rating variables that contain neutral values. This step ensures that the retrieved text data matches the assigned label values, reducing technical errors in the code and subjectivity in the labeling process.

**2.4 Preprocessing Data**

Most of the research related to sentiment analysis focuses on text. One of the important stages in text analysis in natural language processing (NLP) is the pre-processing stage[14]. Pre-processing is a process that functions to process data from a collection of texts that are still biased and unstructured into text data that has been normalized so that it has a good quality[1].

a. Data Cleaning

Data cleaning is the initial step in data preprocessing, where a dataset's errors, inconsistencies, and inaccuracies are identified and corrected or removed. The primary goal of data cleaning is to ensure that the data is suitable for analysis or modeling purposes. It involves handling missing data, eliminating duplicates, and correcting data format or structure. Data cleaning prepares the dataset to be accurate, consistent, and ready for further analysis or modeling.

**Table 2.** Data Cleaning

Actual Data Review	Cleaned data
Belanja di TIK TOK Alhamdulillah banyak promonya ,barang" nya juga bagus" kurirnya ramah..the best buat Tik Tok	Belanja di TIK TOK Alhamdulillah banyak promonya barang nya juga bagus kurirnya ramah the best buat Tik Tok

Based on table 2, the data is cleaned by removing punctuation marks, numbers, and special characters. This helps to eliminate any unwanted noise or irrelevant symbols from the text[15].

b. Case Folding

The next step is case folding. This stage makes all letters the same[16], as an example of making all letters lowercase by using lowercase[17].

**Table 3.** Case Folding

Cleaned data	Case folding
Belanja di TIK TOK Alhamdulillah banyak promonya barang nya juga bagus kurirnya ramah the best buat Tik Tok	belanja di tik tok alhamdulillah banyak promonya barang nya juga bagus kurirnya ramah the best buat tik tok

Based on table 3, the visible data still has some capitalized words, and this process makes each word lowercase.

c. Tokenization

At the tokenization stage, each word is separated from the sentence[18]. This process breaks down the text into individual tokens or words, allowing for further analysis and processing.

**Table 4.** Tokenization

Case folding	Tokenization
belanja di tik tok alhamdulillah banyak promonya barang nya juga bagus kurirnya ramah the best buat tik tok	[belanja, di, tik, tok, alhamdulillah, banyak, promonya, barang, nya, juga, bagus, kurirnya, ramah, the, best, buat, tik, tok]

Based on Table 4, sentences are separated by each word. This tokenization processed by using libraries from NLTK namely nltk.tokenize.

d. Normalization

Normalization involves the process of converting abbreviated, non-standard, or misspelled words into standard words.

**Table 5.** Normalization

Tokenization	Normalization
[belanja, di, tik, tok, alhamdulillah, banyak, promonya, barang, nya, juga, bagus, kurirnya, ramah, the, best, buat, tik, tok]	[belanja, di, tik, tok, alhamdulillah, banyak, promonya, barang, nya, juga, bagus, kurirnya, ramah, the, best, buat, tik, tok]

According to Table 5, Normalization is done using a dictionary from [19]. This dictionary consists of commonly used slang words in Indonesian. The purpose of Normalization is to convert data from the dataset into a standard and formal language. But in the example above, it already uses a standard word.

e. Stopword removal

The next step is to remove stopwords, which are commonly occurring words that do not carry significant meaning or function in the context of the analysis. Examples of stopwords include articles, conjunctions, and prepositions. By removing these words, the focus is placed on more informative and content-bearing terms.

**Table 6.** Stopword removal

Normalization	Stopword removal
[belanja, di, tik, tok, alhamdulillah, banyak, promonya, barang, nya, juga, bagus, kurirnya, ramah, the, best, buat, tik, tok]	[belanja, <del>di</del> , <del>tik</del> , <del>tok</del> , alhamdulillah, <del>banyak</del> , promonya, barang, <del>nya</del> , <del>juga</del> , bagus, kurirnya, ramah, <del>the</del> , best, <del>buat</del> , <del>tik</del> , <del>tok</del> ] [belanja, promonya, barang, bagus, kurirnya, ramah, best]

The stopword is implemented in Table 6 using the NLTK stopwords library. In addition, the stopword dictionary is also added based on words that are not needed in our dataset. For example, there are the words “di”, “nya”, “the”, “buat”, “juga”, and etc.

f. Stemming

Stemming is the process of reducing words to their base or root form. It aims to remove inflectional variations and bring related words to a common base form. This helps in reducing redundancy and consolidating similar words.

**Table 7.** Stemming

Stopword removal	Stemming
[belanja, promonya, barang, bagus, kurirnya, ramah, best]	[belanja, <b>promo</b> , barang, bagus, <b>kurir</b> , ramah, best]

Table 7 demonstrates the stemming process where the word kurirnya is transformed back to its original form, 'kurir'. The stemmer used in this process comes from the Python library Sastrawi and uses the StemmerFactory method.

**2.5 Term Weighting**

After the pre-processing stage, the data enters the feature weighting stage which receiving assigned weights or values. Feature weighting aims to determine how much influence a word has in a sentence [12]. In this research, we compared and evaluated various weighting methods to determine the optimal performance. Some of the methods used are as follows:

a. Term Frequency – Inverse Document Frequency

Term Frequency – Inverse Document Frequency (TFIDF) is a process of transforming textual data into numeric data for each word or feature to be weighted [20]. Term Frequency is the frequency of occurrence of a word in each document and shows how important the word is in each document. Meanwhile, Inverse Document Frequency is the frequency of documents containing the word and shows how often the term occurs. The following TF-IDF value weights can be calculated in the following equation:

$$tf_{td}idf_t = tf_{td} \times \log \left( \frac{N}{df_t} \right) \tag{1}$$

Where variable  $tf_{td}idf_t$  represents the weight value of a word in a document, while variable  $tf_{td}$  denotes the frequency of occurrence of a word in the document, variable  $df_t$  indicates the frequency count of each word, and  $N$  represents the total number of documents. The application of the RNN classification method with TF IDF feature extraction in the sentiment analysis of the Covid-19 vaccine tweet achieved a maximum accuracy of 97% in research [21].

b. Term Frequency – Relevance Frequency

Relevance Frequency is a method made to improve from the previous tf method. This method assesses the relevance of documents as seen from the frequency of occurrence of terms or terms in related categories [22].

$$tf_{tfrf} = tf_{td} \times \log \left( 2 + \frac{b}{\max(1, c)} \right) \tag{2}$$

Where variable  $tf_{tfrf}$  represents the weight value of word  $t$ ,  $tf_{td}$  denotes the frequency of occurrence of word  $t$  in the document,  $b$  means the number of documents containing the word  $t$ , and  $c$  represents the number of documents not containing the word  $t$ . In research[22], a comparison was made between the feature weights of TF RF and TF IDF on trending topics on Twitter. The results show that the TF RF accuracy reaches 62.48% with a precision of 0.623 and a recall of 0.623. While the accuracy of TF IDF has a 0.01% value superior to TF RF.

c. Word2vec

Word embedding is a learning technique in Natural Language Processing (NLP) that converts words into vectors based on real numbers. One of the popular word embedding models is Word2vec. Word2vec represents words as vectors based on some of their features, such as the size and dimensions of the vector[23]. The researcher who popularized this method, namely Mikolov et al. There are two models in Word2vec: skip-grams and continuous bag of words (CBOW). In research[23], it is said that the skip-gram model is an efficient method for researching word vectors in large amounts of unstructured text. Meanwhile, the CBOW model predicts words based on the entire context of the word.

2.6 RNN Classification

Recurrent neural network (RNN) is a form of Artificial neural network designed to recognize and process sequential data such as sound, images, and text. RNN has good performance in classification and extraction processes[11]. The RNN classification has advantages in the data recognition process, namely having internal memory to store important information (feedback loop) from previous information, which is used to make accurate predictions. The way it works is by looping in the architectural design so that information in the past remains stored.

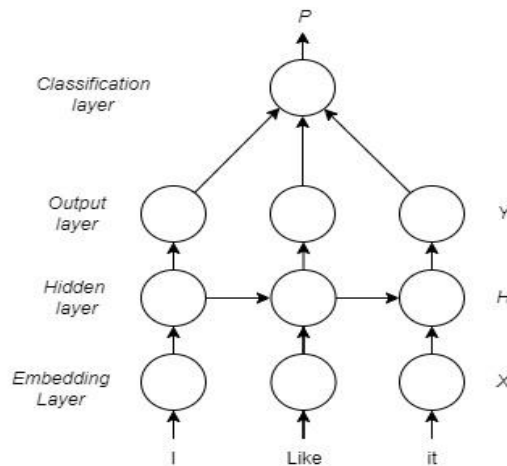


Figure 2. RNN Model

Figure 2 was taken from the research conducted by Jiajun et al., which shows that there are four layers: the embedding layer, the Hidden layer, the Output layer, and the Classification layer. Layer  $X = \{x_1, x_2, \dots, x_n\}$  as the embedding word of each word in a sentence. Layer  $H = \{h_1, h_2, \dots, h_n\}$  is a repeating layer where every node in  $H$  is a repeating unit and nodes  $h_j$  to  $j$ th are defined by the input vector  $x_j$ , and the last iteration is  $h_{j-1}$ . The output layer or  $Y = \{y_1, y_2, \dots, y_n\}$  is the output of the RNN classification results. The last layer, namely the  $P$  layer, is the logistic classification layer whose output from this layer can show positive sentiment probability values. The value of each layer can be calculated by the following equation:

$$h_1 = f(Wx_j + Uh_1 + b_h) \tag{3}$$

$$y_1 = f(Vh_j + b_y) \tag{4}$$

$$p = \frac{1}{1 + e^{-\theta Y}} \tag{5}$$

In this research, The RNN model used the TensorFlow library's Keras API. The model architecture follows a sequential structure, allowing for a linear stack of layers. The core component of the model is the SimpleRNN layer,



representing a basic recurrent neural network unit. This layer processes input sequences, each with a shape of (batch\_size, timesteps, and input\_dim). In this case, the input shape is (1, max\_length), indicating a single input sequence with a maximum length. To address overfitting, a dropout layer is implemented with a dropout rate of 0.2. This layer randomly sets a fraction of input units to 0 during training, enhancing generalization by reducing reliance on specific features. The model concludes with a dense layer, employing softmax activation for multi-class classification. With three units, this layer outputs a probability distribution over the classes. The model is compiled using the Adam optimizer, categorical cross-entropy loss function, and accuracy as the evaluation metric

### 2.7 Validation

The last step in this research is measuring the accuracy of the performance of the model that has been made or also known as validation and evaluation. The validation process involves using k-fold cross-validation, a technique that divides the data into k subsets and iterates over them[24].

**Table 8.** Confussion Matrix

Confussion matrix		Actual value		
		Positive	Neutral	Negative
Predicted value	Positive	TP	FN	FN
	Neutral	FP	TN	FP
	Negative	FN	FN	TP

Based on table 8, the evaluation of the model is carried out with the assessment parameters of the confusion matrix. The confusion matrix is a method used to assess the performance of prediction results in a classification system. Four parameters are employed in this method: true positive (TP), false negative (FN), true negative (TN), and false positive (FP). System performance is measured using accuracy based on the results of the confusion matrix.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{6}$$

Accuracy is a measure of how often a classification model correctly predicts both positive and negative instances. It is calculated by dividing the sum of correctly predicted instances (true positives and true negatives) by the total number of instances. A higher accuracy indicates a more accurate model.

## 3. RESULT AND DISCUSSION

### 3.1 Data

This research utilized 5000 reviews from the Google Play Store website in Indonesia, specifically focusing on the TikTok application. The data collection process involved utilizing the Google-play-scraper library. The reviews were categorized into three sentiment classes: positive, neutral, and negative.

**Table 9.** Data Distribution

Sentiment	Total Data
Positive	4017
Neutral	244
Negative	739

Table 9 shows 4017 reviews labeled as positive, 244 as neutral, and 739 as negative. This stage has already passed two labeling stages, automatically by a few lines of Python code and validated by humans afterward.

### 3.2 Testing scenario and testing result

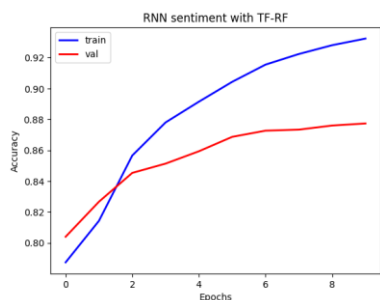
The testing method subjected the model to different term weights, namely TF IDF, TF RF, and the word2vec method used uses a pre-trained model from the Indonesian Wikipedia corpus, with the RNN classification method. This testing is performed on the same dataset. K-Fold is employed during each test to evaluate the classifier's accuracy in predicting the correct class, whether positive, neutral, or negative. Thus, the model is tested with distinct term weights three times to assess its classification performance.

Iterations were performed ten times. Experiments were also carried out by defining the hyper tuning parameters, including setting MAX\_FEATURES by 1000, training the model for ten epochs (EPOCH\_VAL), classifying data into three classes (NUM\_CLASSES), and using a BATCH\_SIZE of 32. The results of the 10-fold experiment of TF RF, TF IDF, and Word2vec Term weighting with RNN classification follow.

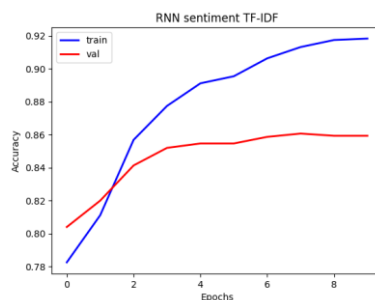
**Table 10.** 10-Fold experiments of TF RF, TF IDF and Word2vec Term weighting with RNN

Iteration (fold)	Accuracy (%)		
	TF RF	TF IDF	Word2vec
1	83,8%	86,0%	81,4%
2	84,8%	85,1%	81,7%
3	85,4%	85,4%	76,8%
4	84,5%	83,9%	79,1%
5	87,1%	86,0%	80,0%
6	86,0%	86,2%	81,9%
7	87,4%	86,8%	77,4%
8	88,8%	<b>88,8%</b>	<b>82,2%</b>
9	<b>89,1%</b>	88,5%	80,2%
10	87,1%	87,7%	80,8%
<b>Avg. Acc</b>	86,7%	86,5%	80,2%
<b>Max. Acc</b>	89,1%	88,8%	82,2%

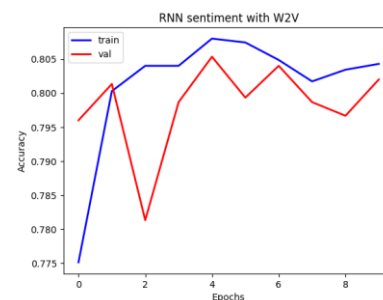
Based on Table 10, k-fold cross-validation test results shows the RNN model exhibited on the TF RF scenario performs best at iteration value = 9, achieving an average accuracy of 86,7%. The average accuracy in the TF IDF scenario with a iteration value 8 was 86,5%. In the third scenario, using the Word2vec feature weighting method with the RNN model, the experiment yielded an average accuracy of 80,2% at iteration = 8.



**Figure 3.** Accuracy graph of the Epoch value in TF RF scenario

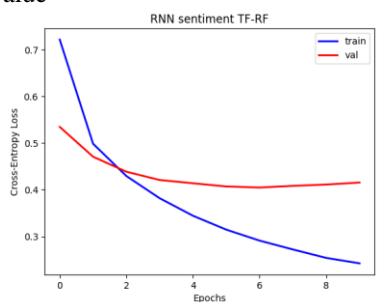


**Figure 4.** Accuracy graph of the Epoch value in TF IDF scenario

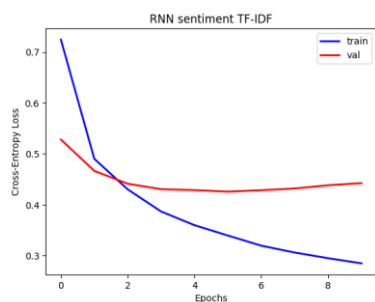


**Figure 5.** Accuracy graph of the Epoch value in Word2vec scenario

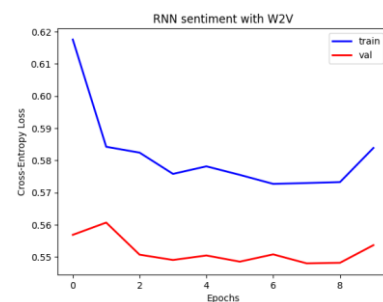
Based on Figure 3 and Figure 4, the TF RF and TF IDF model's accuracy during training scenario increased with each epoch, indicating its ability to learn patterns from the training data and generalize that knowledge to the validation data. However, the Figure 5 shows pattern on the Word2vec scenario shows the fluctuation of the accuracy value



**Figure 6.** Graph of Cross-Entropy Loss to the Epoch in TF RF scenario



**Figure 7.** Graph of Cross-Entropy Loss to the Epoch in TF IDF scenario



**Figure 8.** Graph of Cross-Entropy Loss to the Epoch in Word2vec scenario

The comparison graph of the cross-entropy loss value against the epoch value provides another visualization of the experimental results. Based on Figure 6 and Figure 7, The Cross-Entropy Loss value in TF RF and TF IDF methods is directly proportional to the running epoch value. The more the epoch iteration increases, the lower the loss value. However, Figure 8 shows the graph pattern on the Word2vec scenario is the same, showing the fluctuation of the Cross-Entropy Loss value

### 3.3 Discussion

The RNN model performed well with TF RF and TF IDF feature weighting methods, achieving high accuracy. However, the Word2vec method showed less optimal performance, with fluctuating accuracy and loss values. The TF RF method had a slightly higher accuracy of 87.6% compared to the TF IDF method. In terms of Accuracy and Cross-

Entropy Loss, both TF RF and TF IDF methods exhibited similar patterns, indicating no overfitting. However, the Word2vec model showed irregular fluctuations in Accuracy and Cross-Entropy Loss values, suggesting it struggled to learn the data effectively throughout the epochs. The diverse text datasets and complex word relations also influence these fluctuations. The TikTok app review dataset included words not present in the pre-trained word2vec model with Wikipedia corpus, leading to inconsistencies.

In line with research findings [8], TF IDF and TF RF comparisons also show a similar trend. The TF RF method outperforms TF IDF by a 4% margin in terms of accuracy. Both the TF IDF and TF RF methods play an important role in preventing excessive weighting of words that often appear in documents, thereby increasing the system's accuracy in determining the topics discussed in an opinion.

## 4. CONCLUSION

This study compares the performance of three feature weighting methods (TF IDF, TF RF, and Word2Vec) in RNN classification for sentiment analysis on the TikTok app. Experimental results show that TF RF and TF IDF methods achieve high accuracy, with TF RF performing slightly better. However, the Word2Vec method showed less than optimal performance, with fluctuating accuracy and loss values. It showed that the model could not learn the data optimally, as the text data contained many words that needed to have relevant values in the pre-trained Word2vec model of the Indonesian Wikipedia corpus. These findings highlight the importance of choosing the right feature weighting method for accurate sentiment analysis. The popularity of social media, particularly TikTok, has led to a significant increase in user reviews, making sentiment analysis essential for evaluating product development. Manual assessment of reviews is time-consuming, and sentiment analysis offers a more efficient and automated approach. Further research could explore alternative approaches and larger data sets to improve the analysis.

## REFERENCES

- [1] H. T. Duong and T. A. Nguyen-Thi, "A review: preprocessing techniques and data augmentation for sentiment analysis," *Comput Soc Netw*, vol. 8, no. 1, pp. 1–16, Dec. 2021, doi: 10.1186/S40649-020-00080-X/FIGURES/4.
- [2] S. Daniyah Khansa, K. Yuliaty, and S. Putri, "PENGARUH SOSIAL MEDIA TIKTOK TERHADAP GAYA HIDUP REMAJA," 2022, doi: 10.33822/jep.v5i1.3939.
- [3] J. A. Zulqornain, I. Indriati, and P. P. Adikara, "Tampilan Analisis Sentimen Tanggapan Masyarakat Aplikasi Tiktok Menggunakan Metode Naive Bayes dan Categorical Proportional Difference (CPD)," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 5, no. 7, pp. 2886–2890, Jun. 2021.
- [4] M. Arkansyah, D. Prasetyo, and N. W. R. Amina, "Utilization of Tik Tok Social Media as A Media for Promotion of Hidden Paradise Tourism in Indonesia," *SSRN Electronic Journal*, Mar. 2021, doi: 10.2139/SSRN.3830415.
- [5] J. Psikologi, D. Hidayatul Najah, A. Adi Putra, and N. Aiyuda, "Kecenderungan Narsistik dengan Intensitas Penggunaan Aplikasi Tiktok pada Mahasiswa," *Psychopolytan: Jurnal Psikologi*, vol. 5, no. 1, pp. 1–7, Aug. 2021, doi: 10.36341/PSI.V5I1.1914.
- [6] D. A. P. P. Sari and I. Kurnia, "Upaya Meningkatkan Hasil Belajar Keragaman Budaya Indonesia Melalui Tiktok pada Kelas V Sekolah Dasar," *Jurnal Basicedu*, vol. 6, no. 5, pp. 8285–8295, Jul. 2022, doi: 10.31004/BASICEDU.V6I5.3686.
- [7] C. M. Annur, "Pengguna TikTok di Indonesia Terbanyak Kedua di Dunia per April 2023, Nyaris Salip AS?" <https://databoks.katadata.co.id/datapublish/2023/05/24/pengguna-tiktok-di-indonesia-terbanyak-kedua-di-dunia-per-april-2023-nyaris-salip-as> (accessed Jun. 26, 2023).
- [8] I. K. Dharmendra, N. N. U. Januhari, I. P. Ramayasa, and I. M. A. W. Putra, "Uji Komparasi Sentiment Analysis Pada Opini Alumni Terhadap Perguruan Tinggi," *Jurnal Teknik Informatika UNIKA Santo Thomas*, pp. 1–6, May 2022, doi: 10.54367/JTIUST.V7I1.1748.
- [9] A. Deolika and E. Taufiq Luthfi, "ANALISIS PEMBOBOTAN KATA PADA KLASIFIKASI TEXT MINING," *Jurnal Teknologi Informasi*, vol. 3, no. 2, Dec. 2019.
- [10] N. R. Dewi, E. Y. Puspanignrum, and H. Maulana, "ANALISIS SENTIMEN TWEET VAKSINASI COVID-19 MENGGUNAKAN RNN DENGAN METODE TF-IDF DAN WORD2VEC," *Jurnal Informatika dan Sistem Informasi*, vol. 3, no. 1, Apr. 2022.
- [11] D. Wahyudi and Y. Sibaroni, "Deep Learning for Multi-Aspect Sentiment Analysis of TikTok App using the RNN-LSTM Method," *Building Informatics, Technology and Science*, vol. 4, no. 1, pp. 169–177, Jun. 2022, doi: 10.47065/bits.v4i1.1665.
- [12] M. Kiko, A. Reiki, Y. Sibaroni, and E. B. Setiawan, "Comparison of Term Weighting Methods in Sentiment Analysis of the New State Capital of Indonesia with the SVM Method," *International Journal on Information and Communication Technology (IJOICT)*, vol. 8, no. 2, pp. 53–65, 2022, doi: 10.21108/IJOICT.V8I2.681.
- [13] A. Alshehri and A. Algarni, "TF-TDA: A Novel Supervised Term Weighting Scheme for Sentiment Analysis," *Electronics (Switzerland)*, vol. 12, no. 7, Apr. 2023, doi: 10.3390/electronics12071632.
- [14] D. T. Putra and E. B. Setiawan, "Sentiment Analysis on Social Media with Glove Using Combination CNN and RoBERTa," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 3, pp. 457–563, Jun. 2023, doi: 10.29207/RESTI.V7I3.4892.
- [15] Y. S. Triyantono, S. Al Faraby, and M. Dwifabri, "Analisis Sentimen Terhadap Ulasan Film Menggunakan Word2vec Dan Svm," *eProceedings of Engineering*, vol. 8, no. 4, Aug. 2021.
- [16] M. F. Putra, A. Herdiani, and D. Puspari, "Analisis Pengaruh Normalisasi, TF-IDF, Pemilihan Feature-set Terhadap Klasifikasi Sentimen Menggunakan Maximum Entropy (Studi Kasus : Grab dan Gojek)," *e-Proceeding of Engineering*, vol. 6, no. 2, p. 8520, Aug. 2019.



- [17] D. T. Hermanto, A. Setyanto, and E. T. Luthfi, “Algoritma LSTM-CNN untuk Binary Klasifikasi dengan Word2vec pada Media Online,” *Creative Information Technology Journal*, vol. 8, no. 1, pp. 64–77, Mar. 2021, doi: 10.24076/citec.2021v8i1.264.
- [18] G. A. Buntoro and U. M. Ponorogo, “Sentiments Analysis for Governor of East Java 2018 in Twitter,” *Journal Publications & Informatics Engineering Research*, vol. 3, no. 2, 2019, doi: 10.33395/sinkron.v3i2.10025.
- [19] N. Aliyah Salsabila, Y. Ardhito Winatmoko, A. Akbar Septiandri, and A. Jamal, “Colloquial Indonesian Lexicon,” *Proceedings of the 2018 International Conference on Asian Language Processing, IALP 2018*, pp. 226–229, Jan. 2019, doi: 10.1109/IALP.2018.8629151.
- [20] J. A. Septian, T. M. Fachrudin, and A. Nugroho, “Analisis Sentimen Pengguna Twitter Terhadap Polemik Persepakbolaan Indonesia Menggunakan Pembobotan TF-IDF dan K-Nearest Neighbor,” *Journal of Intelligent System and Computation*, vol. 1, no. 1, pp. 43–49, Aug. 2019, doi: 10.52985/insyst.v1i1.36.
- [21] M. Lestandy, A. Abdurrahim, and L. Syafa’ah, “Analisis Sentimen Tweet Vaksin COVID-19 Menggunakan Recurrent Neural Network dan Naïve Bayes,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 4, pp. 802–808, Aug. 2021, doi: 10.29207/RESTI.V5I4.3308.
- [22] A. N. Assidyk, E. B. Setiawan, S. Si, I. Kurmiawan, S. Pd, and M. Si, “Analisis Perbandingan Pembobotan Tf-idf Dan Tf-rf Pada Trending Topic Di Twitter Dengan Menggunakan Klasifikasi K-nearest Neighbor,” *eProceedings of Engineering*, vol. 7, no. 2, Aug. 2020.
- [23] M. Rahman, M. D. Rahman, A. Djunaidy, and F. Mahananto, “Penerapan Weighted Word Embedding pada Pengklasifikasian Teks Berbasis Recurrent Neural Network untuk Layanan Pengaduan Perusahaan Transportasi,” *Jurnal Sains dan Seni ITS*, vol. 10, no. 1, pp. A1–A6, Aug. 2021, doi: 10.12962/j23373520.v10i1.56145.
- [24] Y. Widyarningsih, G. P. Arum, and K. Prawira, “APLIKASI K-FOLD CROSS VALIDATION DALAM PENENTUAN MODEL REGRESI BINOMIAL NEGATIF TERBAIK,” *BAREKENG: Jurnal Ilmu Matematika dan Terapan*, vol. 15, no. 2, pp. 315–322, Jun. 2021, doi: 10.30598/barekengvol15iss2pp315-322.