Journal of
Information Systems Engineering
and Business Intelligence

Vol.8, No.1, April 2022
Available online at: http://e-journal.unair.ac.id/index.php/JISEBI

Detecting Emotion in Indonesian Tweets: A Term-Weighting Scheme Study

Kuncahyo Setyo Nugroho[1)*], Fitra A. Bachtiar[2)], Wayan Firdaus Mahmudy[3)]

[1)2)3)]*Department of Informatics Engineering, Faculty of Computer Science, Brawijaya University, Indonesia*
Jalan Veteran No.8, Lowokwaru, Malang
[1)]ksnugroho26@gmail.com, [2)]fitra.bachtiar@ub.ac.id, [3)]wayanfm@ub.ac.id

Abstract

Background: Term-weighting plays a key role in detecting emotion in texts. Studies in term-weighting schemes aim to improve short text classification by distinguishing terms accurately.
Objective: This study aims to formulate the best term-weighting schemes and discover the relationship between n-gram combinations and different classification algorithms in detecting emotion in Twitter texts.
Methods: The data used was the Indonesian Twitter Emotion Dataset, with features generated through different n-gram combinations. Two approaches assign weights to the features. Tests were carried out using ten-fold cross-validation on three classification algorithms. The performance of the model was measured using accuracy and F1 score.
Results: The term-weighting schemes with the highest performance are Term Frequency-Inverse Category Frequency (TF-ICF) and Term Frequency-Relevance Frequency (TF-RF). The scheme with a supervised approach performed better than the unsupervised one. However, we did not find a consistent advantage as some of the experiments found that Term Frequency-Inverse Document Frequency (TF-IDF) also performed exceptionally well. The traditional TF-IDF method remains worth considering as a term-weighting scheme.
Conclusion: This study provides recommendations for emotion detection in texts. Future studies can benefit from dealing with imbalances in the dataset to provide better performance.

Keywords: Emotion Detection, Feature Engineering, Term-Weighting, Text Mining

Article history: Received 19 January 2022, decision after peer review 19 February 2022, accepted 19 March 2022, available online 28 April 2022

I. INTRODUCTION

Social media is now the norm for communication, interaction, and collaboration. Therefore, the number of social media users is increasing exponentially in the past years [1], including Twitter. The posts, also known as tweets, limit the content to 280 characters, which are usually concise and straightforward. Tweets may contain opinions and emotions about a current issue [2], so Twitter is a good source of data for sentiment analysis studies, including the classification of tweets based on their polarity for specific purposes [3]–[6]. However, sentiment analysis often fails to identify emotions, and it only recognizes a tweet based on its positive or negative associations [7]. More specific emotion detection is needed to analyze the public's responses toward a certain issue more in-depth.

Emotion is a strong feeling directed at something in response to internal or external stimuli with personal significance. Emotion affects many aspects of life, such as social interaction, behavior, attitude, and decision-making [8]. Branching off from sentiment analysis, emotion detection attempts to extract more subtle expressions such as joy, sadness, or anger in texts. The analysis results can benefit product testing and public policy [9]. Emotion detection can also improve interactions between humans and computers and create intelligent systems that consider users' emotional states [10]. As information technology develops, the need for emotion detection in texts increases to improve artificial intelligence.

There are two main approaches to emotion detection in texts: the lexicon-based and the learning-based. The lexicon-based approach predicts emotions in texts using a lexicon generator [11]–[13]—a dictionary consisting of words that have been pre-rated based on their emotional content. Each word in a sentence is analyzed to determine the context and the emotion. Combination functions such as sum or average are used to predict the end of a sentence. Meanwhile, the learning-based approach applies specific machine-learning methods to classify and predict emotions in texts [14],

* Corresponding author

ISSN 2443-2555 (online) 2598-6333 (print) © 2022 The Authors. Published by Universitas Airlangga.
This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/)
doi: http://dx.doi.org/10.20473/jisebi.8.1.61-70

[15]. The detection is similar to other text classification tasks, such as sentiment analysis. The classification is carried out in four steps: text preprocessing, feature engineering, model training, prediction, and evaluation [16], [17].

In-text classification turns textual representation into vectors to optimize the classification algorithm. This process consists of indexing and term-weighting. Term-weighting calculates the weight of each term in a text to find out the availability and similarity of that term in the text, which in this case, is a tweet [18]. Selecting a subset of terms from the text aims to create a representation and make computation faster and classification more effective. Since the scheme plays an essential role in classification [19], we can use it to produce more information-rich terms and determine the appropriate values for text classification requirements.

Most studies on term-weighting for text classification in Indonesia use a common method: the term frequency-inverse document [20], [21]. An in-depth study using a term-weighting scheme for emotion detection using Indonesian tweets is under-researched. This study has only been done in another language [22], [23]. Since every language has unique characteristics, the current study results are expected to yield different results. The primary objective of this research is to find out the effect of different term-weighting schemes on emotion detection in Indonesian tweets using various machine learning algorithms. This study can also be a starting point for further studies to determine the most effective term-weighting schemes in emotion detection.

II. METHODS

To analyze and detect emotions expressed in a tweet, we developed a supervised learning approach that classifies texts automatically. We compare the term-weighting schemes to obtain classification algorithms with the best performance. As seen in the proposed research method in Fig. 1, the process begins with data collection, text preprocessing, and feature extraction using n-grams and term-weighting schemes. The next stage is dividing data for training and testing, training the models, and evaluating the models. The next subsection describes each of these stages in greater detail.

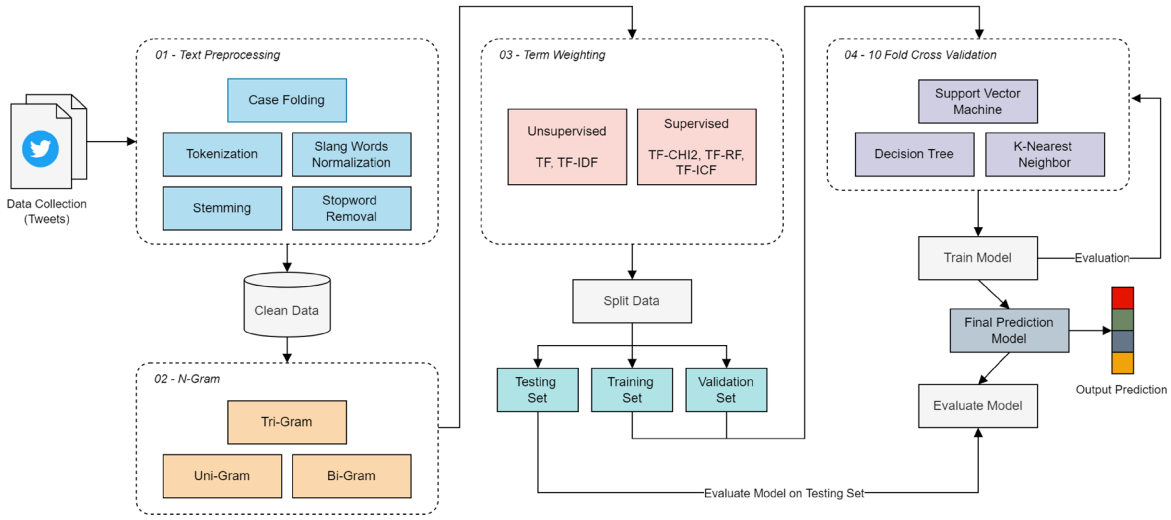

Fig. 1. Proposed research method

A. Dataset

We use the Indonesian Twitter Emotion Dataset [24] containing 4.403 tweets in Bahasa Indonesia that had been labeled with one of the five emotions: love, anger, sadness, joy, and fear. Each row consists of a tweet and its emotion label separated by a comma (,). The first row in the dataset is a header. For a tweet with a comma (,) inside the content, there is a quote (" ") to avoid column separation. The tweets in this dataset were preprocessed using several criteria. The username (@) is replaced with the term [USERNAME]. URL has been replaced with the term [URL]. Private numbers, such as phone numbers and invoices, had been replaced with the term [SENSITIVE-NO]. In terms of distribution, the dataset had unbalanced proportions, as shown in Fig. 2. Most of the tweets fell under the anger category, while the love and fear categories consisted of significantly fewer tweets. However, this study did not perform resampling to handle the imbalanced dataset.

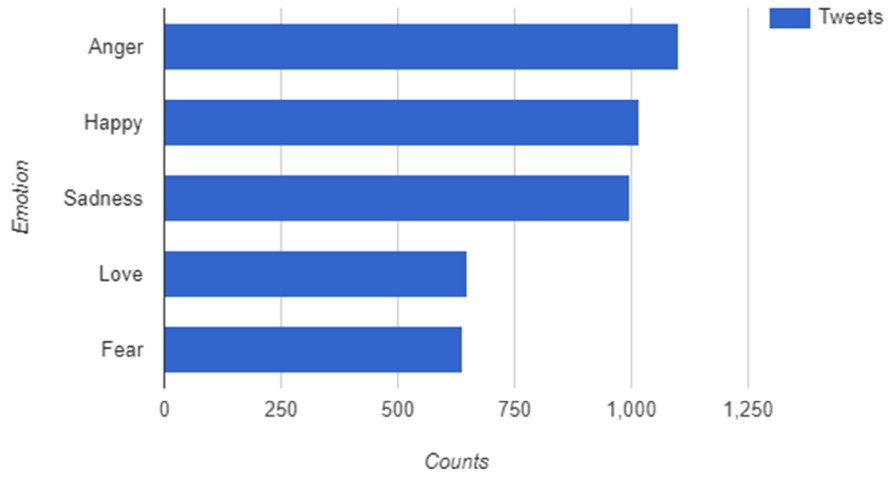

Fig. 2. The tweets distribution by the emotion in the dataset shows an imbalance

B. Text Preprocessing

The available datasets were not well structured, so they needed to be preprocessed. Text preprocessing is necessary for feature extraction to improve classification performance [25]. The text preprocessing in this study was in five stages: case-folding, tokenization, slang words normalization, stemming, and stop-word removal. Case folding converted all letters into lowercase. Characters other than letters such as numbers and punctuation marks were removed. Then, each word in a sentence in the dataset was turned into a collection of tokens. Slang words or abbreviations were changed into formal words. Every word was also turned into its stem or basic form. Stop words that do not affect any polarity, such as conjunctions, were removed.

C. N-Gram

N-gram generates features based on the preprocessed tokens. In general, an n-gram is an n-word token chunk of a sentence. One word equals 1-gram, also known as a uni-gram. Using n-gram can retain more of the original sequence structure of the text. Therefore, the n-gram representation can be more informative. Theoretically, with t unique words, there may be t^2 unique 2-gram, also known as a bi-gram. In practice, the number of features generated is small because not every word can follow every other word. However, there is usually a more distinct n-gram ($n > 1$) than words. This means n-gram is a significantly larger and less frequent feature space. The larger the n, the richer the information, and the greater the computational cost. This study combines three types of n-grams: uni-gram, bi-gram, and tri-gram.

D. Term-weighting Scheme

Term-weighting focuses on assigning weight (score) to each term in a text representation process. An appropriate text representation makes an effective text classification system [26]. This process increases a classifier's efficiency by highlighting the most discriminatory terms in each category. In other words, it is critical to assign a value to each term to describe the document properly. In general, term-weighting schemes can be unsupervised or supervised [27]. The unsupervised approach does not rely on prior training documents for class categories. The shortcoming is that it does not consider the distribution of documents. In contrast, the supervised approach replaces the IDF factor from the TF-IDF by using prior knowledge about the document category and the statistical information from the texts included in the category. Research has confirmed that the supervised approach outperforms the unsupervised one [28].

1) Term Frequency (TF)

TF is the most popular scheme because it is simple. TF assumes that the terms appearing more frequently in the document have higher importance. TF is a local weighting scheme because it depends on the number of times a particular term appears in a document. $TF(t_i, d_j)$ for the i^{th} term (t_i) in the j^{th} document can be calculated as follows (1):

$$TF(t_i, d_j) = \frac{f_{ij}}{\sum_{t \in T_j} f_{tj}} \qquad (1)$$

where the term t_i, f_{ij} indicates the number of times t_i appears in the given j^{th} document.

2) Term Frequency - Inverse Document Frequency (TF-IDF)

TF-IDF is a general scheme used to represent documents in a vector space model [29]. TF-IDF combines TF with inversed document frequency (IDF). Currently, TF-IDF is the most widely used scheme a text classification [30] and document categorization [31]. TF-IDF is based on the assumption that terms that appear less frequently in the dataset have the highest importance. TF-IDF can be calculated as follows (2):

$$TF \cdot IDF(t_i, d_j) = TF(t_i, d_j) \times log \frac{D}{d(t_i)} \quad (2)$$

where $TF(t_i, d_j)$ is TF of term t_i in document d_j, while $\frac{D}{d(t_i)}$ is IDF of term t_i.

3) Term Frequency - Chi-Square (TF-CHI2)

Chi-Square ($\mathcal{X}^2$) is a statistical test method that evaluates the correlation between two variables and determines whether the variables are related or not. TF-CHI2 was proposed by [32] to calculate how independent t_k and c_i were. TF-CHI2 is intended to select the term with the highest $\mathcal{X}^2$ value. TF-CHI2 can be calculated as follows (3):

$$TF \cdot IDF(t_i, d_j) = TF(t_i, d_j) \times \frac{N(AD - CB)^2}{(A + C)(B + D)(A + B)(C + D)} \quad (3)$$

where A is the number of documents in the positive class with t_i appearance, B is the number of documents in the positive class with no t_i appearance, C is the number of negative class documents where the term t_i appears, D is the number of documents of negative class with no term t_i appearance, and N is the entire documents.

4) Term Frequency - Relevance Frequency (TF-RF)

TF-RF assumes that a term with the highest frequency and concentration is in a positive class and not a negative class, as it detects positive samples more than negative samples [33]. The relevance frequency factor is expected to make the term discrimination more powerful, compared to the TF-IDF scheme that fails to discriminate positive samples from negative ones. TF-RF can be calculated as follows (4):

$$TF \cdot RF(t_i, d_j) = TF(t_i, d_j) \times \log\left(2 + \frac{b}{max(1, c)}\right) \quad (4)$$

where b is the number of documents containing term i (t_i), and c is the number of documents that do not contain term i (t_i).

5) Term Frequency - Inverse Category Frequency (TF-ICF)

While IDF pays attention to the occurrence of a term in a document set, TF-ICF considers the presence of a term in a class [34]. Class frequency (CF) states the number of classes in which term i (t_i) occurs, while $|C|$ shows the number of categories in a document. The term importance is assumed to have an inverse proportion to the number of classes containing the term. TF-ICF can be calculated as follows (5).

$$TF \cdot ICF(t_i, d_j) = TF(t_i, d_j) \times \log\left(\frac{|C|}{cf(t_i)}\right) \quad (5)$$

E. Classification Algorithm

To confirm the performance and effect of each term-weighting scheme in detecting emotions, we use three learning algorithms, namely k-nearest neighbor (k-NN), support vector machine (SVM), and decision tree (DT). The three algorithms perform well, but it is important to note that this study examines the effectiveness of term-weighting schemes rather than the algorithms' performance. Therefore, we did not attempt any hyperparameter tuning to determine the best parameters to improve the performance of the classification model.

F. Performance Evaluation

The model evaluation uses a ten-fold cross-validation procedure to avoid overfitting. The experiment done by randomizing the training set into ten parts. Meanwhile, the classification model training uses a nine-fold procedure.

The procedure was carried out up to ten times. In addition to accuracy, the classification model's performance also uses an F1 score as a metric of performance. Accuracy is a ratio of correct predictions to the overall classification results [35] and can be calculated using (6). An F1 score is the average harmonic of the precision and recall scores [36], as defined at (7). Precision is a ratio of true positive predictions to the total number of positive predictions, and recall measures the ratio of true positive predictions to the total number of true positive documents.

$$accuracy = \frac{|\ total\ correctly\ classified\ documents\ |}{|\ total\ number\ of\ documents\ |} \quad (6)$$

$$F1 = 2 \times \frac{precision \times recall}{precisio + recall} \quad (7)$$

III. Results

The experiment aims to determine the best term-weighting scheme and the relationship between the n-gram combination and the term-weighting scheme in different classification algorithms. After preprocessing, the dataset was divided into a training set and a testing set. Seventy-five percent of the dataset was used for training and the remainder for testing. The n-gram combination was used to generate features from the dataset. The number of features generated by the n-gram processes is shown in Fig. 3. The uni-gram generated 11.376 unique words. The combination of uni-gram and bi-gram generated 52.360 unique words. The combination of uni-gram, bi-gram, and tri-gram generated 95.421 unique words. The features that were formed would then be calculated using different term-weighting schemes.

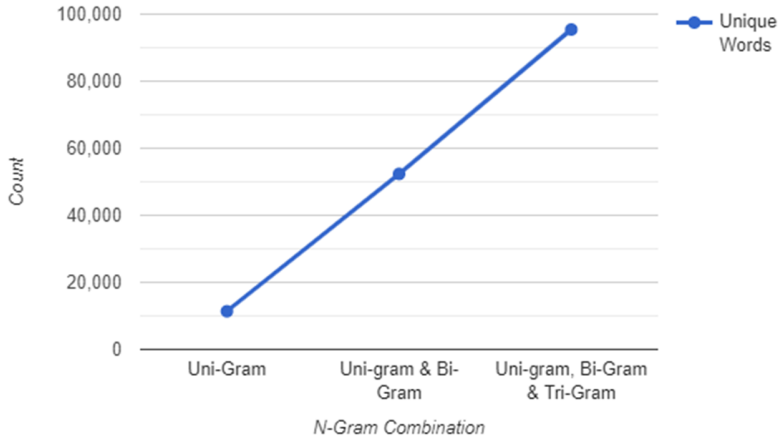

Fig. 3. The comparison of the number of features produced by the n-gram combinations

Each combination of n-gram, term-weighting scheme, and classification algorithms produces different results. The first test results in Table 1 show the performance of the classifier model using term-weighting schemes and uni-grams. The use of k-NN resulted in the best performance if paired with TF-RF, with an accuracy of 0.5145 and an F1 score of 0.5049. The decision tree using TF-ICF generated a higher accuracy score of 0.5455 and an F1 score of 0.5458. The highest accuracy was 0.6618 using SVM with TF-IDF, and the F1 score reached 0.6597. Fig. 4 shows that the term-weighting scheme with the supervised approach reached an average F1 score higher than the unsupervised approach for each classifier model.

The further test results show that a combination of uni-gram and bi-gram performed better, as shown in Table 2. TF-RF with k-NN shows the best performance, with an accuracy of 0.51 and an F1 score of 0.5012. Like k-NN, the decision tree performed best when combined with TF-ICF, with an accuracy of 0.5518 and an F1 score of 0.5548. Unlike the previous test, the best term-weighting scheme in SVM was TF-ICF with an accuracy of 0.6727 and an F1 score of 0.674. Fig. 5 shows that the term-weighting scheme with the supervised approach generates a better average F1 score for each classifier model.

TABLE 1
TERM-WEIGHTING SCHEME RESULT IN UNI-GRAM

ML Model	Term-weighting Scheme	Cross-Validation Accuracy (Std Dev.)	Testing Accuracy	F1 Score
k-NN	TF	0.5012 (+/- 0.0381)	0.4991	0.4885
	TF-IDF	0.4964 (+/- 0.0399)	0.48	0.4684
	TF-RF	**0.5048 (+/- 0.0376)**	**0.5145**	**0.5049**
	TF-ICF	0.2524 (+/- 0.0664)	0.2436	0.2046
	TF-CHI2	0.4345 (+/- 0.0416)	0.4382	0.433
SVM	TF	0.6515 (+/- 0.0431)	0.6527	0.651
	TF-IDF	**0.6491 (+/- 0.0465)**	**0.6618**	**0.6597**
	TF-RF	0.6512 (+/- 0.0411)	0.6509	0.6494
	TF-ICF	0.6400 (+/- 0.0408)	0.62	0.6216
	TF-CHI2	0.6248 (+/- 0.0573)	0.6445	0.6433
Decision Tree	TF	0.5082 (+/- 0.0466)	0.5391	0.5383
	TF-IDF	0.5052 (+/- 0.0578)	0.5327	0.5333
	TF-RF	0.5039 (+/- 0.0497)	0.5264	0.5294
	TF-ICF	**0.5264 (+/- 0.0519)**	**0.5455**	**0.5458**
	TF-CHI2	0.5209 (+/- 0.0484)	0.5236	0.5297

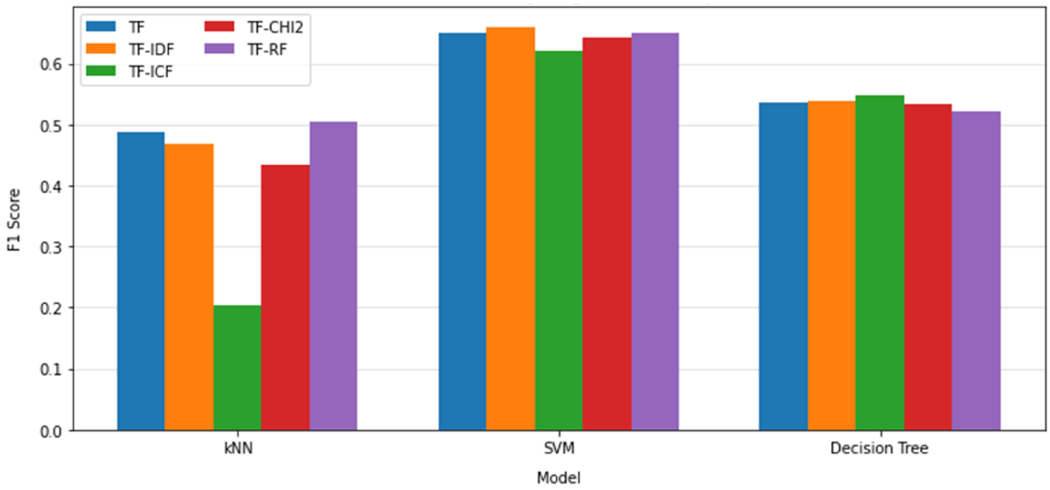

Fig. 4 The result of the F1 score for each term-weighting scheme using uni-gram

TABLE 2
TERM-WEIGHTING SCHEME RESULT IN UNI-GRAM AND BI-GRAM

ML Model	Term-weighting Scheme	Cross-Validation Accuracy (Std Dev.)	Testing Accuracy	F1 Score
k-NN	TF	0.5048 (+/- 0.0481)	0.5009	0.4897
	TF-IDF	0.4994 (+/- 0.0483)	0.5064	0.4963
	TF-RF	**0.5036 (+/- 0.0471)**	**0.51**	**0.5012**
	TF-ICF	0.1727 (+/- 0.0690)	0.1645	0.0734
	TF-CHI2	0.3830 (+/- 0.0317)	0.3991	0.3792
SVM	TF	0.6539 (+/- 0.0468)	0.6591	0.657
	TF-IDF	0.6609 (+/- 0.0526)	0.6636	0.6604
	TF-RF	0.6555 (+/- 0.0500)	0.6591	0.6571
	TF-ICF	**0.6742 (+/- 0.0562)**	**0.6727**	**0.674**
	TF-CHI2	0.6361 (+/- 0.0578)	0.6355	0.6323
Decision Tree	TF	0.5070 (+/- 0.0481)	0.5373	0.5401
	TF-IDF	0.5058 (+/- 0.0437)	0.5264	0.5292
	TF-RF	0.4997 (+/- 0.0508)	0.5236	0.5261
	TF-ICF	**0.5270 (+/- 0.0631)**	**0.5518**	**0.5548**
	TF-CHI2	0.5415 (+/- 0.0496)	0.4909	0.4879

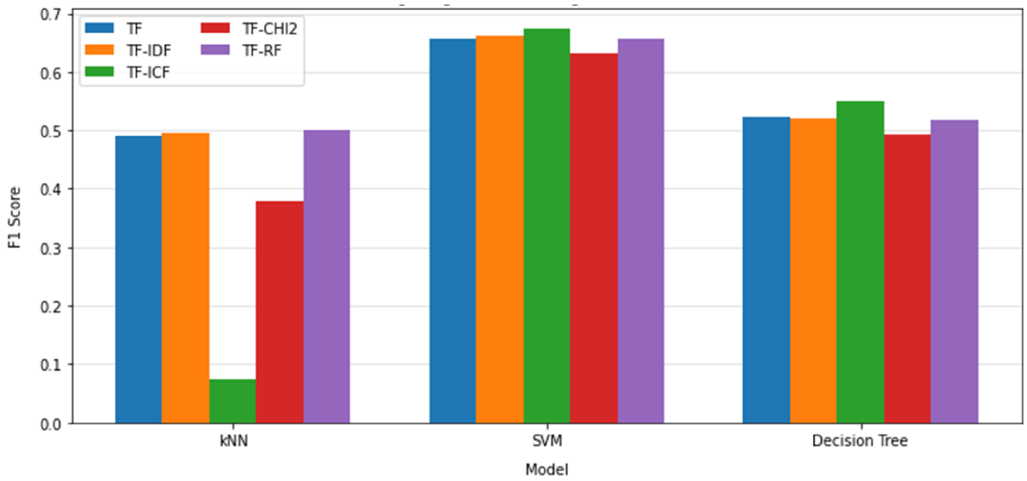

Fig. 5 The result of the F1 score for each term-weighting scheme using a combination of uni-gram and bi-gram

The last test results on the combinations of uni-gram, bi-gram, and tri-gram are shown in Table 3. Unlike the previous two tests, k-NN showed the best performance when using TF-IDF, with an accuracy of 0.5091 and an F1 score of 0.4984. In comparison, the decision tree's accuracy was 0.5564, and F1 score was 0.5582 in the TF-ICF scheme. SVM using TF-IDF showed the best performance, with an accuracy of 0.6673 and an F1 score of 0.6635. Fig. 6 shows the F1 test results of the term-weighting schemes with the three classification algorithms. The term-weighting scheme with the unsupervised approach shows a better average F1 score for each classifier model.

Based on the results of the three tests, the overall performance of the term-weighting scheme and the classifier model increases when using a uni-gram combination. The performance trend declines when using a triadic combination of uni-gram, bi-gram, and tri-gram. Meanwhile, SVM seems the most stable classifier model because it shows the best performance in the three tests. TF-ICF and TF-RF are term-weighting schemes that show the best performance among other schemes. The term-weighting scheme with the supervised method shows consistent results with different classifiers in various tests.

TABLE 3
TERM-WEIGHTING SCHEME RESULT IN UNI-GRAM, BI-GRAM, AND TRI-GRAM

ML Model	Term-weighting Scheme	Cross-Validation Accuracy (Std Dev.)	Testing Accuracy	F1 Score
k-NN	TF	0.5036 (+/- 0.0516)	0.4982	0.4869
	TF-IDF	**0.5061 (+/- 0.0449)**	**0.5091**	**0.4984**
	TF-RF	0.5018 (+/- 0.0473)	0.5036	0.4952
	TF-ICF	0.1782 (+/- 0.0528)	0.1782	0.109
	TF-CHI2	0.3512 (+/- 0.0410)	0.3655	0.3374
SVM	TF	0.6497 (+/- 0.0621)	0.6545	0.6516
	TF-IDF	**0.6564 (+/- 0.0401)**	**0.6673**	**0.6635**
	TF-RF	0.6503 (+/- 0.0600)	0.6545	0.6514
	TF-ICF	0.6609 (+/- 0.0602)	0.6564	0.6588
	TF-CHI2	0.6376 (+/- 0.0690)	0.6218	0.6184
Decision Tree	TF	0.5094 (+/- 0.0466)	0.5173	0.5221
	TF-IDF	0.5133 (+/- 0.0487)	0.5082	0.5131
	TF-RF	0.5115 (+/- 0.0483)	0.5145	0.5182
	TF-ICF	**0.5379 (+/- 0.0562)**	**0.5564**	**0.5582**
	TF-CHI2	0.5448 (+/- 0.0405)	0.4527	0.443

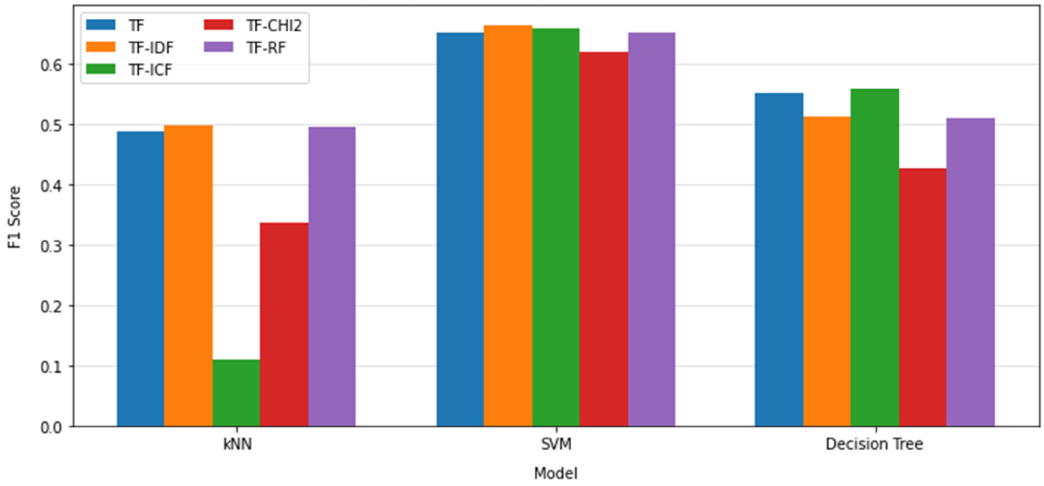

Fig. 6 The result of the F1 score for each term-weighting scheme using a combination of uni-gram, bi-gram, and tri-gram

IV. Discussion

We tested the effectiveness of the classification models by varying the term-weighting schemes on the dataset. The results show that different term-weighting schemes can increase term discrimination effectiveness in detecting emotion in texts. N-grams also play an essential role in generating features from the dataset, even though further research is needed to corroborate the results. In theory, $n > 2$ will generate fewer features. We attempted to modify the n-gram combinations to generate more features even though it may not improve the classifier model's performance. The experimental results show that the features generated by a combination of uni-gram and bi-gram perform better.

Although TF-CHI2 showed the best performance in [37], we did not find a significant increase in performance from TF-IDF in the whole trial. Overall, we found TF-ICF to be the best term-weighting scheme. A study by [38] showed that TF-ICF outperformed other approaches such as TF-RF and TF-IDF. Similarly, a study [23] classifying Arabic texts using k-NN showed that TF-ICF was the best approach. However, it did not give satisfactory results in our tests. Instead, SVM with TF-ICF performed the most satisfactorily. We also found that the advantages of the term-weighting scheme with the supervised approach over the unsupervised one need further investigation. Although in our tests, TF-ICF and TF-RF generated better results than other methods, we did not find a consistent advantage. Even in several other experiments, the traditional TF-IDF method still performed well. Therefore, TF-IDF remains a good choice for term-weighting schemes.

This study has proven the effect of different term-weighting schemes on emotion detections, but the findings can be extended to other tasks. For example, sentiment analysis can classify emotions into a more general form based on their polarity—angry emotions are classified as negative, and happy emotions as positive. We also observed that the performance of a classifier model could be improved. A class imbalance in the dataset can lead to a bias in the class selection. One way to deal with this is to use resampling techniques [39].

The practical implications of this study can benefit other fields of study. For example, they can be used to detect stress or mental disorders [40], [41], so the health policy or interventions will be more targeted. It can also be applied to conversational media, where a system can show the emotions of its users [42].

V. Conclusions

Considering the critical role of a term-weighting scheme in text classification, especially in emotion detection in short texts, we proposed a term-weighting scheme study to improve the term-distinguishing power in text classification. The current study's experiments were carried out using the Indonesian Twitter Emotion Dataset corpus. While an algorithm is an essential part of a text classifier system, text representation is the primary model for building a text classifier. The bag-of-words approach remains a mainstay in many text representations today. However, many other techniques from term types or term weights are also studied for more information. This study focuses on finding the best text classifier performance in emotion detection based on different term-weighting schemes.

In this study, the n-gram combinations affect the performance. The higher the combination, the higher the number of features (unique words) produced. A ten-fold cross-validation test was performed using three machine learning algorithms against five term-weighting schemes. We found that TF-ICF and TF-RF are the highest performing term-

weighting schemes based on the testing and analysis. In general, the term-weighting scheme with a supervised approach is better than the unsupervised one. However, we did not find consistent gains—other experiments found that TF-IDF also performed well.

In emotion detection, further research improves the current research's results to achieve better model performance. Future studies can, for example, consider the effect of class imbalance on the dataset for each text weighting scheme. Class imbalance in datasets can create a bias in choosing the majority class. Other learning algorithms can also be explored by considering hyperparameter tuning. Tuning aims to find a combination of parameters to achieve high performance. Finally, this term-weighting scheme can be used in other similar datasets or similar text classification tasks, such as sentiment analysis.

Author Contributions: *Kuncahyo Setyo Nugroho*: Conceptualization, Methodology, Formal Analysis, Software, Writing - Original Draft, Writing - Review & Editing. *Fitra Abdurrachman Bachtiar*: Supervision, Investigation, Validation, Formal Analysis. *Wayan Firdaus Mahmudy*: Supervision, Investigation, Formal Analysis.

Funding: This research received no specific grant from any funding agency.

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

REFERENCES

[1] M. Anderson and A. Smith, "Social Media Use in 2021," 2018. Accessed: Aug. 30, 2021. [Online]. Available: https://www.pewresearch.org/internet/2021/04/07/social-media-use-in-2021/.

[2] M. De Choudhury, S. Counts, and M. Gamon, "Not all moods are created equal! Exploring human emotional states in social media," in *ICWSM 2012 - Proceedings of the 6th International AAAI Conference on Weblogs and Social Media*, 2012, vol. 6, no. 1, pp. 66–73.

[3] A. R. Prananda and I. Thalib, "Sentiment Analysis for Customer Review: Case Study of GO-JEK Expansion," *J. Inf. Syst. Eng. Bus. Intell.*, vol. 6, no. 1, p. 1, Apr. 2020, doi: 10.20473/jisebi.6.1.1-8.

[4] D. Alita, S. Priyanta, and N. Rokhman, "Analysis of Emoticon and Sarcasm Effect on Sentiment Analysis of Indonesian Language on Twitter," *J. Inf. Syst. Eng. Bus. Intell.*, vol. 5, no. 2, p. 100, Oct. 2019, doi: 10.20473/jisebi.5.2.100-109.

[5] L. Muflikhah and D. J. Haryanto, "High Performance of Polynomial Kernel at SVM Algorithm for Sentiment Analysis," *J. Inf. Technol. Comput. Sci.*, vol. 3, no. 2, pp. 194–201, 2018, doi: 10.25126/jitecs.20183260.

[6] R. A. Cahya, F. A. Bachtiar, and W. F. Mahmudy, "Comparison of Bagging Ensemble Combination Rules for Imbalanced Text Sentiment Analysis," *J. Inf. Technol. Comput. Sci.*, vol. 6, no. 1, pp. 33–49, 2021, doi: 10.25126/jitecs.202161206.

[7] L. Nahar, Z. Sultana, N. Iqbal, and A. Chowdhury, "Sentiment Analysis and Emotion Extraction: A Review of Research Paradigm," May 2019, doi: 10.1109/ICASERT.2019.8934654.

[8] W. Wang, L. Chen, K. Thirunarayan, and A. P. Sheth, "Harnessing twitter 'big data' for automatic emotion identification," in *Proceedings - 2012 ASE/IEEE International Conference on Privacy, Security, Risk and Trust and 2012 ASE/IEEE International Conference on Social Computing, SocialCom/PASSAT 2012*, 2012, pp. 587–592, doi: 10.1109/SocialCom-PASSAT.2012.119.

[9] A. Seyeditabari, N. Tabari, and W. Zadrozny, "Emotion Detection in Text: a Review," Jun. 2018, [Online]. Available: https://arxiv.org/abs/1806.00674.

[10] F. Ren and Y. Bao, "A review on human-computer interaction and intelligent robots," *Int. J. Inf. Technol. Decis. Mak.*, vol. 19, no. 1, pp. 5–47, Feb. 2020, doi: 10.1142/S0219622019300052.

[11] A. Bandhakavi, N. Wiratunga, S. Massie, and D. Padmanabhan, "Lexicon Generation for Emotion Detection from Text," *IEEE Intell. Syst.*, vol. 32, no. 1, pp. 102–108, Jan. 2017, doi: 10.1109/MIS.2017.22.

[12] S. Poria, A. Gelbukh, E. Cambria, A. Hussain, and G. Bin Huang, "EmoSenticSpace: A novel framework for affective common-sense reasoning," *Knowledge-Based Syst.*, vol. 69, no. 1, pp. 108–123, Oct. 2014, doi: 10.1016/j.knosys.2014.06.011.

[13] S. M. Mohammad and P. D. Turney, "Crowdsourcing a word-emotion association lexicon," in *Computational Intelligence*, Aug. 2013, vol. 29, no. 3, pp. 436–465, doi: 10.1111/j.1467-8640.2012.00460.x.

[14] V. V Ramalingam, A. Pandian, A. Jaiswal, and N. Bhatia, "Emotion detection from text," in *Journal of Physics: Conference Series*, Apr. 2018, vol. 1000, no. 1, p. 012027, doi: 10.1088/1742-6596/1000/1/012027.

[15] E. Batbaatar, M. Li, and K. H. Ryu, "Semantic-Emotion Neural Network for Emotion Recognition from Text," *IEEE Access*, vol. 7, pp. 111866–111878, 2019, doi: 10.1109/ACCESS.2019.2934529.

[16] K. Vasa, "Text Classification through Statistical and Machine Learning Methods: A Survey," *Int. J. Eng. Dev. Res.*, vol. 4, no. 2, pp. 655–658, 2016.

[17] T. Y. Christyawan and W. F. Mahmudy, "Text Classification and Visualization on News Title Using Self Organizing Map," in *3rd International Conference on Sustainable Information Engineering and Technology, SIET 2018 - Proceedings*, Jul. 2018, pp. 332–336, doi: 10.1109/SIET.2018.8693189.

[18] T. Sabbah *et al.*, "Modified frequency-based term-weighting schemes for text classification," *Appl. Soft Comput.*, vol. 58, pp. 193–206, Sep. 2017, doi: 10.1016/j.asoc.2017.04.069.

[19] A. T. Ni'mah and A. Z. Arifin, "Perbandingan Metode Term-weighting terhadap Hasil Klasifikasi Teks pada Dataset Terjemahan Kitab Hadis," *Rekayasa*, vol. 13, no. 2, pp. 172–180, Aug. 2020, doi: 10.21107/rekayasa.v13i2.6412.

[20] K. S. Nugroho, I. Istiadi, and F. Marisa, "Naive Bayes classifier optimization for text classification on e-government using particle swarm optimization," *J. Teknol. dan Sist. Komput.*, vol. 8, no. 1, pp. 21–26, 2020, doi: 10.14710/jtsiskom.8.1.2020.21-26.

[21] B. A. Ardhani, N. Chamidah, and T. Saifudin, "Sentiment Analysis Towards Kartu Prakerja Using Text Mining with Support Vector Machine and Radial Basis Function Kernel," *J. Inf. Syst. Eng. Bus. Intell.*, vol. 7, no. 2, p. 119, Oct. 2021, doi: 10.20473/jisebi.7.2.119-128.
[22] A. Mazyad, F. Teytaud, and C. Fonlupt, "A comparative study on term-weighting schemes for text classification," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2018, vol. 10710 LNCS, pp. 100–108, doi: 10.1007/978-3-319-72926-8_9.
[23] D. S. Guru, M. Ali, M. Suhil, and M. Hazman, "A study of applying different term-weighting schemes on Arabic text classification," in *Lecture Notes in Networks and Systems*, vol. 43, Springer, Singapore, 2019, pp. 293–305.
[24] M. S. Saputri, R. Mahendra, and M. Adriani, "Emotion Classification on Indonesian Twitter Dataset," in *Proceedings of the 2018 International Conference on Asian Language Processing, IALP 2018*, Jan. 2019, pp. 90–95, doi: 10.1109/IALP.2018.8629262.
[25] A. I. Kadhim, "An Evaluation of Preprocessing Techniques for Text Classification," *Int. J. Comput. Sci. Inf. Secur.*, vol. 16, no. 6, pp. 22–32, 2018.
[26] B. Naderalvojoud, A. S. Bozkir, and E. A. Sezer, "Investigation of term-weighting schemes in classification of imbalanced texts," *Proc. Eur. Conf. Data Min. 2014 Int. Conf. Intell. Syst. Agents 2014 Theory Pract. Mod. Comput. 2014 - Part Multi Conf. Comput. Sci. Inf. Syst. MC*, pp. 39–46, 2014.
[27] A. Alsaeedi, "A survey of term-weighting schemes for text Classification," *Int. J. Data Mining, Model. Manag.*, vol. 12, no. 2, pp. 237–254, 2020, doi: 10.1504/IJDMMM.2020.106741.
[28] Y. Gu and X. Gu, "A supervised term-weighting scheme for multi-class text categorization," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2017, vol. 10363 LNAI, pp. 436–447, doi: 10.1007/978-3-319-63315-2_38.
[29] Z. Erenel, H. Altinçay, and E. Varoğlu, "Explicit use of term occurrence probabilities for term-weighting in text categorization," *J. Inf. Sci. Eng.*, vol. 27, no. 3, pp. 819–834, 2011, doi: 10.6688/JISE.2011.27.3.2.
[30] F. A. Bachtiar, W. Paulina, and A. N. Rusydi, "Text Mining for Aspect Based Sentiment Analysis on Customer Review : a Case Study in the Hotel Industry," in *5th International Workshop on Innovations in Information and Communication Science and Technology*, 2020, no. March.
[31] S. W. Kim and J. M. Gil, "Research paper classification systems based on TF-IDF and LDA schemes," *Human-centric Comput. Inf. Sci.*, vol. 9, no. 1, pp. 1–21, Aug. 2019, doi: 10.1186/s13673-019-0192-7.
[32] F. Debole and F. Sebastiani, "Supervised Term-weighting for Automated Text Categorization," in *Text Mining and its Applications. Studies in Fuzziness and Soft Computing*, Springer, Berlin, Heidelberg, 2004, pp. 81–97.
[33] M. Lan, C. L. Tan, and H. B. Low, "Proposing a new term-weighting scheme for text categorization," in *Proceedings of the National Conference on Artificial Intelligence*, 2006, vol. 1, pp. 763–768.
[34] D. Wang and H. Zhang, "Inverse-category-frequency based supervised term-weighting schemes for text categorization," *J. Inf. Sci. Eng.*, vol. 29, no. 2, pp. 209–225, Mar. 2013, doi: 10.6688/JISE.2013.29.2.2.
[35] K. S. Nugroho and F. A. Bachtiar, "Text-Based Emotion Recognition in Indonesian Tweet using BERT," in *2021 4th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, Dec. 2022, pp. 570–574, doi: 10.1109/isriti54043.2021.9702838.
[36] F. Rustam, A. Mehmood, M. Ahmad, S. Ullah, D. M. Khan, and G. S. Choi, "Classification of Shopify App User Reviews Using Novel Multi Text Features," *IEEE Access*, vol. 8, pp. 30234–30244, 2020, doi: 10.1109/ACCESS.2020.2972632.
[37] Z. H. Deng, S. W. Tang, D. Q. Yang, M. Zhang, L. Y. Li, and K. Q. Xie, "A comparative study on feature weight in text categorization," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 3007, pp. 588–597, 2004, doi: 10.1007/978-3-540-24655-8_64.
[38] G. Domeniconi, G. Moro, R. Pasolini, and C. Sartori, "A comparison of term-weighting schemes for text classification and sentiment analysis with a supervised variant of tf.idf," in *Communications in Computer and Information Science*, 2016, vol. 584, pp. 39–58, doi: 10.1007/978-3-319-30162-4_4.
[39] C. Padurariu and M. E. Breaban, "Dealing with data imbalance in text classification," in *Procedia Computer Science*, Jan. 2019, vol. 159, pp. 736–745, doi: 10.1016/j.procs.2019.09.229.
[40] S. Ghosh, A. Ekbal, and P. Bhattacharyya, "What Does Your Bio Say? Inferring Twitter Users' Depression Status From Multimodal Profile Information Using Deep Learning," *IEEE Trans. Comput. Soc. Syst.*, 2021, doi: 10.1109/TCSS.2021.3116242.
[41] S. Ghosh, A. Ekbal, and P. Bhattacharyya, "A Multitask Framework to Detect Depression, Sentiment and Multi-label Emotion from Suicide Notes," *Cognit. Comput.*, vol. 14, no. 1, pp. 110–129, Feb. 2022, doi: 10.1007/s12559-021-09828-7.
[42] S. Ghosh, D. Varshney, A. Ekbal, and P. Bhattacharyya, "Context and Knowledge Enriched Transformer Framework for Emotion Recognition in Conversations," in *Proceedings of the International Joint Conference on Neural Networks*, Jul. 2021, vol. 2021-July, doi: 10.1109/IJCNN52387.2021.9533452.

Publisher's Note: Publisher stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.