Analysis of K-NN with the Integration of Bag of Words, TF-IDF, and N-Grams for Hate Speech Classification on Twitter

Kuncoro Hadi[1*], Ema Utami[2]

[1,2] *Universitas Amikom Yogyakarta, Indonesia*

*corr_author: kuncorohadi@students.amikom.ac.id

Abstract – Social media has emerged as one of the primary communication channels in the modern world, but it has simultaneously become a platform where hate speech can spread easily. This study attempts to evaluate the performance of a hate speech classification model using the K-Nearest Neighbors (K-NN) algorithm along with various feature extraction techniques, specifically Bag of Words (BoW), Term Frequency-Inverse Document Frequency (TF-IDF), and N-Grams. The dataset used in this study consists of 13169 entries, which represent a diverse range of hate speech examples commonly encountered on social media platforms. In this experimental investigation, we assess the efficacy of the model using each feature extraction technique. The findings reveal that the K-NN model exhibits optimal performance when the k parameter is set to 3 (k=3). Under this configuration, the model achieves an accuracy of 86.88%, with a precision of 88.27%, a recall of 86.88%, and an F1-Score of 86.50%. These results show that the integration of TF-IDF feature extraction technique with K-NN algorithm produces superior performance in hate speech classification.

Keywords: hate speech, K-Nearest Neighbors, Bag of Words, TF-IDF, N-Grams, F1 Score

I. INTRODUCTION

Twitter allows users to share opinions, thoughts, and information, enabling rapid communication and engagement with various issues [1], [2]. While it promotes freedom of expression, this can lead to the spread of hate speech, despite Twitter's measures to mitigate it [3]-[6]. Recent research has identified the presence of provocative language on Twitter, highlighting the need for a classification model trained on hate speech to address this issue [7], [8]. Detecting hate speech is essential for fostering a safer, inclusive environment and preventing societal discord, polarization, and threats to democratic processes [9].

The study by [10] identified hate speech and offensive language with various labels on Twitter, while research [11] focused on preprocessing and classification based on race, religion, and neutrality, showing that parameter selection and training data size impact outcomes. Feature extraction techniques like BoW, TF-IDF, and N-Grams are essential for converting raw text into numerical data for machine learning [12]. TF-IDF has been widely used with algorithms such as SVM, Naive Bayes, and K-NN, improving accuracy in hate speech detection [13]. Enhancements like class frequency further boost performance and reduce errors [14], proving effective in other areas like text similarity, SMS spam detection, and SQL injection prevention, especially when combined with N-Grams [15], [5]. Studies [16] and [17] demonstrated N-Grams' effectiveness in detecting fake news and sentiment analysis, with research [18]-[20] showing its success in classifying tweets, emphasizing N-Grams' broad applicability in enhancing automated detection accuracy across various domains.

Each feature extraction technique has strengths and weaknesses, but combining them can create a more robust model, previous research has shown [21]. Choosing the proper technique is crucial for improving accuracy and efficiency in hate speech classification [22]. The study by [23] focused on detecting hate speech in Indonesian using different machine learning models, achieving the highest accuracy with varied classifiers and feature extraction methods. Creating a safer online environment requires identifying and mitigating hate speech on social media [24]. Hate speech classification in NLP has been tackled using machine learning and deep learning approaches with varying success, with even simple models like K-NN proving effective [8], [25]-[27]. Challenges include handling the complexity of social media language, such as diverse grammar and slang, and large datasets, necessitating meticulous sentiment analysis [28]. The study by [29] demonstrated that machine learning and deep learning methods could be employed for hate speech classification on social media.

The literature review by [1] emphasizes that machine learning approaches, though simpler than deep learning,

can yield competitive results when combined with suitable feature extraction techniques. The study by [30] showed that K-NN with BoW achieved 66.21% accuracy in hate speech detection, while [23] improved this to 83.03%, highlighting BoW's relevance in text classification tasks [31]. TF-IDF further enhances model performance, as shown by [23], where replacing BoW with TF-IDF increased accuracy to 84.11%. This is supported by [9], demonstrating that TF-IDF reduces classification errors and improves precision, while [32] confirms its effectiveness in converting research paper metadata into numerical vectors. The study by [17] demonstrated significant accuracy improvements in sentiment analysis of Indonesian-language reviews by combining N-Grams (N=2) with Naive Bayes and K-NN, achieving 97.26% and 93.76% accuracy, respectively, highlighting N-Grams' importance in enhancing sentiment analysis performance.

Several approaches have been undertaken to classify hate speech on Twitter. This study builds upon existing approaches by focusing on hate speech analysis using the dataset from research by [10]. The feature extraction methods employed are BoW, TF-IDF, and N-Grams. Classification will be conducted using the K-NN algorithm, similar to the methodology in the study by [11]. This research aims to evaluate the performance of the K-NN model in the classification of Indonesian-language hate speech on Twitter using BoW, TF-IDF, and N-Grams are used for feature extraction. The study also examines how variations in the parameter k affect the model's performance. In addition to measuring accuracy, the F1-Score will be used as the primary evaluation metric for hate speech classification. As discussed in research [33], the F1-Score offers a balance between precision and recall, which highlights its utility in assessing model performance [34].

II. METHOD

The research process involves several stages: data collection, text preprocessing, feature extraction, classification, and evaluation.

A. Data Collection

The data for this research was obtained from the study by [10], which provided a publicly available dataset (Fig. 1). The dataset's contents are shown in the image below. Figure 1 illustrates the dataset content, which consists of 1 tweet column and 12 class columns. This dataset includes various types of hate speech in the Indonesian language, totaling 13,169 tweets. The class used in this study is the Hate Speech class (HS column).

	Tweet	HS	Abusive	\
0	– disaat semua cowok berusaha melacak perhatia...	1	1	
1	RT USER: USER siapa yang telat ngasih tau elu?...	0	1	
2	41. Kadang aku berfikir, kenapa aku tetap perc...	0	0	
3	USER USER AKU ITU AKU\n\nKU TAU MATAMU SIPIT T...	0	0	
4	USER USER Kaum cebong kapir udah keliatan dong...	1	1	
...	...	..	...	
13164	USER jangan asal ngomong ndasmu. congor lu yg ...	1	1	
13165	USER Kasur mana enak kunyuk'	0	1	
13166	USER Hati hati bisu :(.g\n\nlagi bosan huft \...	0	0	
13167	USER USER USER USER Bom yang real mudah terdet...	0	0	
13168	USER Mana situ ngasih(": itu cuma foto ya kuti...	1	1	

	HS_Individual	HS_Group	HS_Religion	HS_Race	HS_Physical	HS_Gender	\
0	1	0	0	0	0	0	
1	0	0	0	0	0	0	
2	0	0	0	0	0	0	
3	0	0	0	0	0	0	
4	0	1	1	0	0	0	
...	...	...	...	...	...	...	
13164	1	0	0	0	1	0	
13165	0	0	0	0	0	0	
13166	0	0	0	0	0	0	
13167	0	0	0	0	0	0	
13168	1	0	0	0	0	0	

	HS_Other	HS_Weak	HS_Moderate	HS_Strong
0	1	1	0	0
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	1	0
...	...	...	...	...
13164	0	1	0	0
13165	0	0	0	0
13166	0	0	0	0
13167	0	0	0	0
13168	1	1	0	0

[13169 rows x 13 columns]

Fig. 1 Dataset content and number of class columns

The total data in the HS column is 13169 tweets. There are two label categories. Label 0 indicates that the tweet does not contain hate speech (non-hate speech), totaling 7608 tweets. On the other hand, label 1 indicates that the tweet contains hate speech, with a total of 5561 tweets. This data was then subjected to preprocessing.

B. Text Pre-Processing

The text preprocessing process is a crucial step in preparing data before it is used in a model, aiming to prepare the text documents for subsequent processing steps [35]. The following image is an example of a dataset that has undergone preprocessing. Fig. 2 shows that each data row is assigned a unique ID, and the text is converted to lowercase. Irrelevant characters such as URLs, retweet symbols, usernames, and emojis are removed. The text is then filtered to retain only alphanumeric characters, and slang words are normalized using a slang dictionary. The stemming process is performed using the Sastrawi dictionary to convert words to their root forms. Stopwords are applied to focus on significant words. The preprocessing results are stored in the 'Tweet_Clean' column and saved into a new file (Fig. 2).

C. Feature Extraction

The feature extraction methods used in this study are the BoW, TF-IDF, and N-Grams approaches.

1) BoW: The BoW approach is a feature extraction method used to classify documents by evaluating the frequency of unique words appearing within a large text corpus [36]. BoW offers simplicity and flexibility, enabling text representation based on the pattern of word occurrences in a document, making it suitable for various text analysis applications [23]. The formula for calculating the BoW value is provided in (1). Each word t in document d results in a feature matrix where each row represents a single document, and each column represents a unique word from the entire document corpus.

$$\text{BoW}(t,d) = \text{The number of occurrences of the word } t \text{ in document } d \quad (1)$$

ID	Tweet	Tweet_Clean
d00001	- disaat semua cowok berusaha melacak perhatia...	cowok usaha lacak perhati lantas remeh perhati...
d00002	rt user: user siapa yang telat ngasih tau elu?...	telat tau edan sarap gaul cigax jifla cal licew
d00003	41. kadang aku berfikir, kenapa aku tetap perc...	41 kadang pikir percaya tuhan jatuh kali kali ...
d00004	user user aku itu aku\n\nku tau matamu sipit t...	ku tau mata sipit lihat
d00005	user user kaum cebong kapir udah keliatan dong...	kaum cebong kafir lihat dongok dungu haha
...	...	...
d13165	user jangan asal ngomong ndasmu. congor lu yg ...	bicara ndasmu congor sekata anjing
d13166	user kasur mana enak kunyuk'	kasur enak kunyuk
d13167	user hati hati bisu :(.g\n\nlagi bosan huft \...	hati hati bisu bosan duh
d13168	user user user user bom yang real mudah terdet...	bom real mudah deteksi bom kubur dahsyat ledak...
d13169	user mana situ ngasih(": itu cuma foto ya kuti...	situ foto kutil onta

Fig. 2 Preprocessing results

2) TF-IDF: The TF-IDF method assigns weights to words based on their relevance to a specific document [37], thereby helping to determine the importance of a word in the context of the entire document. The formula for calculating the TF-IDF value is provided in (2).

$$\text{TF-IDF}\ (t,d,D) = \text{TF}\ (t,d)\ \text{x IDF}\ (t,D) \quad (2)$$

3) N-Grams: N-Grams is an approach for determining sequences of N words that can be represented as features in feature extraction techniques [16]. This approach aids in identifying sequential patterns in text, which can be used for more in-depth text analysis and classification [38]. The formula used in N-Grams is provided in (3), all of these methods are used to extract features from the text, which are then input into the K-NN model for classification.

$$\text{N-gram}\ (t,d) = \text{The sequence of words } t \text{ in document } d. \quad (3)$$

D. K-NN

The K-NN algorithm is used to classify objects based on the training data closest to the object in question [39]. The proximity or distance of these neighbors is calculated using the Euclidean distance method. The Euclidean distance method helps measure the interpretive distance between two objects [40]. The representation of the Euclidean distance method is calculated using (4),where $d(p,q)$ is the Euclidean distance between data points p and q. In this formula p_i and q_i are the values of the i-th feature for the data points p and q.

$$d(p,q) = \sqrt{\sum_{i=1}^{p} (p_i - q_i)^2} \quad (4)$$

E. Evaluation

Measuring model performance is a crucial step in machine learning, as it helps in determining the most suitable model for use [41]. One of the techniques to assess model performance, particularly in classification tasks (supervised learning) in machine learning, is the confusion matrix [42]. This matrix consists of two rows and two columns: TP (True Positive), TN (True Negative), FP (False Positive), and FN (False Negative), as shown in Table I. The confusion matrix is structured as shown in the table, where the predicted classes are compared against the actual classes. The matrix contains TP, TN, FP, and FN. This matrix is used to evaluate the performance of a classification model by providing insights into the number of correct and incorrect predictions for each class.

Once the confusion matrix is determined, it is then used to calculate the values of accuracy, precision, recall, and F1-score [43]. These values are calculated using the following equations:

1) Accuracy: Measures the proportion of correct predictions out of the total predictions made by the model, calculated using (5).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \text{ x } 100\% \quad (5)$$

*2) Precision***:** Measures how accurate the model's predictions are by calculating the ratio of true positive predictions to all positive predictions, calculated using (6).

$$Precision = \frac{TP}{TP + FP} \text{ x } 100\% \quad (6)$$

3) Recall: Measures the model's ability to identify all positive instances present in the data, calculated using (7).

$$Recall = \frac{TP}{TP + FN} \text{ x } 100\% \quad (7)$$

4) F1-score: A calculation that represents the balance between precision and recall. If the values of FN and FP are not close to each other, it is preferable to use the F1-score over accuracy. It is calculated using (8).

$$\text{F1-score} = 2 \text{ x } \frac{Precision \text{ x } Recall}{Precision + Recall} \text{ x } 100\% \quad (8)$$

The results of these metrics are then used to compare the performance of the model across different feature extraction methods and K-values in K-NN.

III. RESULT AND DISCUSSION

After the data has undergone preprocessing, classification is performed using the K-NN algorithm. The feature extraction techniques used are BoW, TF-IDF, and N-Grams. The results of this classification process are presented in the classification results section. Subsequently, an analysis of the model's performance, including the interpretation and implications of the findings, is discussed in the discussion section.

A. Classification Results

In this study, the performance of the K-NN model is evaluated using BoW, TF-IDF, and N-Grams, from these

TABLE I
CONFUSION MATRIX

Predicted Class	**Actual Class**	
	Positive	**Negative**
Positive	TP	FP
Negative	FN	TN

techniques for hate speech classification in Indonesian tweets. Each model is tested with K-values of 3, 5, 7, 9, 11, 13, and 15, and the results are evaluated using performance metrics such as Accuracy, Precision, Recall, and F1-Score. Table II below presents the performance evaluation results based on various K values. The performance results of the K-NN model are observed based on the k parameter and the feature extraction technique used.

Based on the model performance results according to the k parameter and the feature extraction technique used. The findings of this evaluation are as follows:

1) Performance Metrics Analysis

- Accuracy: At k=3, K-NN + TF-IDF achieved the highest accuracy at 86.88%, followed K-NN + BoW and K-NN + N-Grams reached 85.31% and 82.43%, respectively. As k increased to 15, accuracy declined for all models, with K-NN + TF-IDF remaining the most accurate at 81.32%, while the K-NN + N-Grams experienced the largest drop to 71.42%.
- Precision: K-NN + TF-IDF had the highest precision at 88.27% at k=3, with K-NN + BoW and K-NN + N-Grams had lower precision at 85.32% and 82.96%, respectively. At k=15, K-NN + TF-IDF still led with 81.26%, while the K-NN + N-Grams dropped to 72.49%.

Recall: At k=3, K-NN + TF-IDF achieved the highest recall at 86.88%, with the K-NN + BoW had a recall of 85.31%, and K-NN + N-Grams had the lowest at 82.43%. At k=15, recall decreased across all models, but K-NN + TF-IDF remained the most effective at 81.32%.

- F1-Score: The K-NN + TF-IDF model had the highest F1-Score at 86.50% at k=3, followed K-NN + BoW and K-NN + N-Grams scored lower at 85.21% and 82.05%, respectively. By k=15, K-NN + TF-IDF maintained the top F1-Score at 81.22%, with K-NN + N-Grams the lowest at 69.79%.

2) Inter-Model Comparison

- K-NN + Bag of Words: Performed well at k=3 with an accuracy of 85.31%, precision of 85.32%, recall of 85.31%, and F1-Score of 85.21%. However, performance declined significantly as k increased, with all metrics dropping to around 77.7% at k=15. This indicates that BoW struggles with complex hate speech variations and lacks context consideration.
- K-NN + TF-IDF: Consistently the best performer across all metrics. At k=3, it achieved the highest accuracy (86.88%), precision (88.27%), recall (86.88%), and F1-Score (86.50%). While performance declined with increasing k, it remained the strongest model, with metrics still above 81% at k=15, demonstrating TF-IDF's robustness in capturing hate speech.

TABLE II
K-NN MODEL PERFORMANCE RESULTS

k=	Method	Accuracy	Precision	Recall	F1 Score
3	K-NN + BoW	85.31%	85.32%	85.31%	85.21%
3	**K-NN + TF-IDF**	**86.88%**	**88.27%**	**86.88%**	**86.50%**
3	K-NN + N-Grams	82.43%	82.96%	82.43%	82.05%
5	K-NN + BoW	81.98%	81.93%	81.98%	81.86%
5	K-NN + TF-IDF	84.25%	84.69%	84.25%	83.97%
5	K-NN + N-Grams	78.59%	79.07%	78.59%	78.05%
7	K-NN + BoW	80.38%	80.31%	80.38%	80.29%
7	K-NN + TF-IDF	83.11%	83.28%	83.11%	82.84%
7	K-NN + N-Grams	75.40%	75.83%	75.40%	74.66%
9	K-NN + BoW	78.98%	78.89%	78.98%	78.67%
9	K-NN + TF-IDF	82.50%	82.54%	82.50%	82.33%
9	K-NN + N-Grams	73.64%	74.00%	73.64%	72.78%
11	K-NN + BoW	78.54%	78.44%	78.54%	78.41%
11	K-NN + TF-IDF	82.05%	82.03%	82.05%	81.92%
11	K-NN + N-Grams	73.31%	73.98%	73.31%	72.21%
13	K-NN + BoW	78.33%	78.23%	78.33%	78.18%
13	K-NN + TF-IDF	81.73%	81.69%	81.73%	81.61%
13	K-NN + N-Grams	72.28%	73.13%	72.28%	70.93%
15	K-NN + BoW	77.76%	77.65%	77.76%	77.60%
15	K-NN + TF-IDF	81.32%	81.26%	81.32%	81.22%
15	**K-NN + N-Grams**	**71.42%**	**72.49%**	**71.42%**	**69.79%**

- K-NN + N-Grams: Showed the lowest performance, with k=3 metrics at around 82% and declining drastically to an F1-Score of 69.79% at k=15. This suggests N-Grams are less effective in hate speech detection when used alone.

3) Best Performance Results

- Best Results for k: A k value of 3 yielded the best accuracy, precision, recall, and F1-Score performance for nearly all feature extraction techniques tested. As the k value increased, there was a tendency for performance to decline, particularly for N-Grams, indicating that selecting a lower k value is more optimal for this classification task.
- Performance of Feature Extraction Techniques: The TF-IDF technique performed significantly better performance compared to BoW and N-Grams at k=3. The N-Grams technique exhibited the lowest performance compared to TF-IDF and BoW, although its performance slightly declined as the k value increased.

Model Performance Comparison illustrated in the graph in Fig. 3. This results in the display of four evaluation metrics: accuracy, precision, recall, and F1 score, which also shows that TF-IDF feature extraction proves to provide the best performance for the K-NN model in the hate speech classification task. The optimal performance of the K-NN model in hate speech classification is achieved at k=3, indicating that choosing a small number of nearest neighbors is more appropriate for this task. Increasing the value of k above 3 tends to reduce the results and effectiveness of the model, especially for N-Gram. Adding more neighbors does not always produce benefits in this context. These results underline the importance of choosing the proper feature extraction technique and k parameter in the K-NN model for hate speech classification. Combining feature extraction techniques also serves as an effective strategy to improve the classification performance in this study.

B. Discussion

Several findings warrant further discussion based on the results obtained. The analysis of these results not only provides insights into the effectiveness of each approach used but also aids in understanding how the K-NN model can be optimized. This discussion will focus on several key aspects that influence model performance, such as the selection of the k value, the advantages of feature extraction techniques, and comparisons with previous research findings.

1) Optimal Performance at k=3: Based on the experimental results, k=3 is the optimal value for the K-NN model in hate speech classification. This finding aligns with the results from [23], which also used k=3. At lower k values, the model demonstrates stable and superior performance across almost all metrics. Higher k values reduce the model's effectiveness, particularly in N-Grams models, indicating that adding more neighbors does not necessarily yield benefits.

2) The Advantages of TF-IDF: The TF-IDF feature extraction technique provided the best performance among all the methods tested, as discussed in other studies [9]. In other contexts, such as text similarity detection or spam detection [13], TF-IDF has also demonstrated high efficiency [21]. In this case, the K-NN model using TF-IDF showed the best results at k=3, making it a highly effective technique for hate speech classification in this case.

3) N-Grams Performance: The model that solely used N-Grams exhibited significant weaknesses, particularly at higher k values. Therefore, the use of N-Grams as a standalone technique is less recommended [16].

4) Comparison with Previous Research Results: The experimental results obtained in this study surpass those of several prior studies on hate speech classification. For instance, the study by [44] using the K-NN method with TF-IDF at k=10 achieved a maximum accuracy of 67.86%. Additionally, the study by [45] using the K-NN method with TF-IDF resulted in an accuracy of 59.68%, while the study by [34] using DistilBERT with SVM produced an F1-Score of 78.58%.

In this study, the most optimal results for hate speech classification were achieved with k=3, yielding the highest accuracy of 86.88% and an F1-Score of 86.50%. These results represent a significant improvement compared to previous studies, whether utilizing the same model or different approaches. This improvement underscores the effectiveness of K-NN with the integration of BoW, TF-IDF, and N-Grams in classifying hate speech on social media. It highlights the superiority of this method in delivering more balanced and optimal results across all metrics. Hate speech detection heavily relies on the linguistic context. In Indonesian, for example, slang, abbreviations, and code-switching between formal and informal language can present challenges that may not exist in other languages. This makes it harder for a model trained on Indonesian data to effectively generalize to the from other languages without further adaptation.

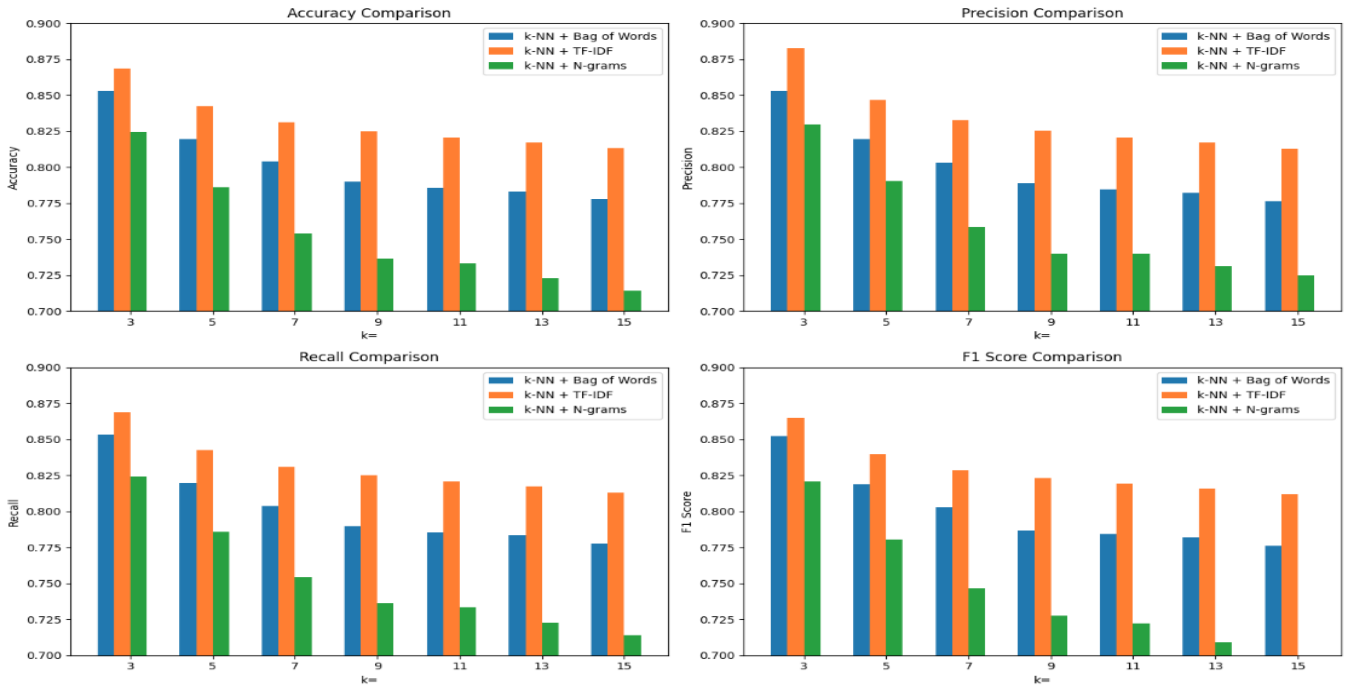

Fig. 3 Model Performance Comparison

IV. CONCLUSION

This study demonstrates that in the context of hate speech classification, selecting appropriate feature extraction techniques and k values is crucial for building an effective model. TF-IDF has proven to be a highly effective technique, with k=3 yielding the best results, achieving the highest accuracy of 86.88% and an F1-Score of 86.50%. Moreover, the recommendations for future research include exploring more sophisticated feature extraction methods that allow words to be represented in lower-dimensional vectors, capturing semantic relationships between words. FastText, for example, also considers as sub-words. This is very helpful in dealing with languages with much morphology, like Indonesian. BERT produces different representations for the same word in different contexts, which can capture nuances in hate speech that may not be detected by static methods such as TF-IDF. Further optimization of the k parameter and other hyperparameters, pruning (reducing the number of irrelevant parameters) and quantization (reducing data precision) techniques can be used to reduce model size and speed up inference time, and developing multi-label approaches to address the complexity of hate speech that spans more than one category.

ACKNOWLEDGEMENT

We extend our sincere gratitude to AMIKOM Yogyakarta as the place of study and to Universitas Muhammadiyah Kalimantan Timur for their support.

REFERENCES

[1] M. Subramanian, V. Easwaramoorthy Sathiskumar, G. Deepalakshmi, J. Cho, and G. Manikandan, "A survey on hate speech detection and sentiment analysis using machine learning and deep learning models," Oct. 2023, *Elsevier B.V.* pp. 110-121, doi: 10.1016/j.aej.2023.08.038.

[2] D. R. Beddiar, M. S. Jahan, and M. Oussalah, "Data expansion using back translation and paraphrasing for hate speech detection," *Online Soc Netw Media*, vol. 24, Jul. 2021, p. 100153, doi: 10.1016/j.osnem.2021.100153.

[3] A. P. J. Dwitama, "Deteksi Ujaran Kebencian Pada Twitter Bahasa Indonesia Menggunakan Machine Learning: Reviu Literatur," *Jurnal Sains, Nalar, dan Aplikasi Teknologi Informasi*, vol. 1, no. 1, Aug. 2021, pp. 33-41 doi: 10.20885/snati.v1i1.5.

[4] K. Mutisari Hana, S. Al Faraby, and A. Bramantoro, "Multi-label Classification of Indonesian Hate Speech on Twitter Using Support Vector Machines," 2020, pp. 1-7, doi: 10.1109/ICoDSA50139.2020.9212992.

[5] H. C. Husada and A. S. Paramita, "Analisis Sentimen Pada Maskapai Penerbangan di Platform Twitter Menggunakan Algoritma Support Vector Machine (SVM)," *Teknika*, vol. 10, no. 1, pp. 18–26, Feb. 2021, doi: 10.34148/teknika.v10i1.311.

[6] M. S. Jahan and M. Oussalah, "A systematic review of hate speech automatic detection using natural language processing," Aug. 14, 2023, *Elsevier B.V.* p. 126323, doi: 10.1016/j.neucom.2023.126232.

[7] Rini, E. Utami, and A. D. Hartanto, "Systematic Literature Review of Hate Speech Detection with Text Mining," in *2020 2nd International Conference on Cybernetics and Intelligent System, ICORIS 2020*, Institute of Electrical and Electronics Engineers Inc., Oct. 2020, pp. 1-6, doi: 10.1109/ICORIS50180.2020.9320755.

[8] M. S. Asramanggala, S. S. Prasetyowati, and Y. Sibaroni, "Optimal Number Data Trains in Hoax News Detection of Indonesian using SVM and Word2Vec," *Building of Informatics, Technology and Science (BITS)*, vol. 5, no. 1, Jun. 2023, pp. 21-28, doi: 10.47065/bits.v5i1.3516.

[9] M. O. Ibrohim and I. Budi, "Hate speech and abusive language detection in Indonesian social media: Progress and challenges," Aug. 01, 2023, *Elsevier Ltd*. p. e18647, doi: 10.1016/j.heliyon.2023.e18647.

[10] M. O. Ibrohim and I. Budi, "Multi-label Hate Speech and Abusive Language Detection in Indonesian Twitter," 2019, pp.46-57, doi: 10.18653/v1/W19-3506.

[11] M. A. Gumilang, T. Dwi Puspitasari, F. Wulandari, E. Antika, H. A. Putranto, and A. Samsudin, "Implementation of K-Nearest Neighbor For Classify Hate Speech on Twitter," in *Proceedings - IEIT 2023: 2023 International Conference on Electrical and Information Technology*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 113–119. doi: 10.1109/IEIT59852.2023.10335596.

[12] D. Mengliev, M. Eshkulov, V. Barakhnin, R. Abdullayev, N. Boltayev, and B. Ibragimov, "Linguistic Nuances in Text Analysis: TF-IDF Metric's Algorithm Implementation for the Karakalpak Language Recognition," in *Proceedings - 2024 IEEE Ural-Siberian Conference on Biomedical Engineering, Radioelectronics and Information Technology, USBEREIT 2024*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 19–22. doi: 10.1109/USBEREIT61901.2024.10584051.

[13] G. Ubale and S. Gaikwad, "SMS Spam Detection Using TFIDF and Voting Classifier," in *2022 International Mobile and Embedded Technology Conference, MECON 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 363–366. doi: 10.1109/MECON53876.2022.9752078.

[14] S. Jain, Sapan. , K. Jain, and S. Vasal, "An Effective TF-IDF Model to Improve the Text - Classification Performance," *International Conference on Communication Systems and Network Technologies*, pp. 1066–1069, 2024, doi: 10.1109/CSNT60213.2024.10545818.

[15] L. Du and C. Hu, "Text similarity detection method of power customer service work order based on TFIDF algorithm," in *2022 IEEE 5th International Conference on Information Systems and Computer Aided Education, ICISCAE 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 978–982. doi: 10.1109/ICISCAE55891.2022.9927512.

[16] A. E. Qasem and M. Sajid, "Exploring the Effect of N-grams with BOW and TF-IDF Representations on Detecting Fake News," in *2022 International Conference on Data Analytics for Business and Industry, ICDABI 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 741–746. doi: 10.1109/ICDABI56818.2022.10041537.

[17] J. Asian, O. T. N. K. Putra, M. A. Ayu, and T. Mantoro, "Sentiment Analysis with N-Gram Preprocessing for Online-Shopping Reviews in Indonesian Language," in *2023 IEEE 9th International Conference on Computing, Engineering and Design, ICCED 2023*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 1-6, doi: 10.1109/ICCED60214.2023.10425567.

[18] G. N. A. Atillo, B. D. Gerardo, and R. P. Medina, "Twitter Sentiment Analysis with Maximum Entropy and Naive Bayes Using N -gram Approach," in *Proceedings of 2023 International Conference on Information Management and Technology, ICIMTech 2023*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 368–372. doi: 10.1109/ICIMTech59029.2023.10277786.

[19] P. Tijare, "Event Labeling Approach for Twitter Datasets Leveraging N-grams, Topics, and Machine Learning Algorithms for Enhanced Event Detection," in *4th International Conference on Communication, Computing and Industry 6.0, C216 2023*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 1-6, doi: 10.1109/C2I659362.2023.10430550.

[20] E. Payares, E. Puertas, and J. C. Martinez-Santos, "Quantum N-Gram Language Models for Tweet Classification," in *Proceedings - 2023 IEEE 5th International Conference on Cognitive Machine Intelligence, CogMI 2023*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 69–74. doi: 10.1109/CogMI58952.2023.00019.

[21] Z. Guo, Q. Li, X. Li, M. Xiao, R. Hu, and Y. Jiang, "SQL Injection Detection Method Based on N-Gram and TFIDF," in *Proceedings - 2023 International Seminar on Computer Science and Engineering Technology, SCSET 2023*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 204–207. doi: 10.1109/SCSET58950.2023.00053.

[22] R. Sathishkumar, T. Karthikeyan, K. P. Praveen, and S. M. Shamsundar, "Ensemble Text Classification with

TF-IDF Vectorization for Hate Speech Detection in Social Media," in *2023 International Conference on System, Computation, Automation and Networking, ICSCAN 2023*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 1-7, doi: 10.1109/ICSCAN58655.2023.10395354.

[23] E. Utami, Rini, A. F. Iskandar, and S. Raharjo, "Multi-Label Classification of Indonesian Hate Speech Detection Using One-vs-All Method," in *Proceedings - 2021 IEEE 5th International Conference on Information Technology, Information Systems and Electrical Engineering: Applying Data Science and Artificial Intelligence Technologies for Global Challenges During Pandemic Era, ICITISEE 2021*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 78–82. doi: 10.1109/ICITISEE53823.2021.9655883.

[24] E. Puraivan, R. Venegas, and F. Riquelme, "An empiric validation of linguistic features in machine learning models for fake news detection," *Data Knowl Eng*, vol. 147, Sep. 2023, p. 102207, doi: 10.1016/j.datak.2023.102207.

[25] H. Kibriya, A. Siddiqa, W. Z. Khan, and M. K. Khan, "Towards safer online communities: Deep learning and explainable AI for hate speech detection and classification," *Computers and Electrical Engineering*, vol. 116, May 2024, p. 109153, doi: 10.1016/j.compeleceng.2024.109153.

[26] S. T. Rabani, A. M. Ud Din Khanday, Q. R. Khan, U. A. Hajam, A. S. Imran, and Z. Kastrati, "Detecting suicidality on social media: Machine learning at rescue," *Egyptian Informatics Journal*, vol. 24, no. 2, pp. 291–302, Jul. 2023, doi: 10.1016/j.eij.2023.04.003.

[27] N. Sevani, I. A. Soenandi, Adianto, and J. Wijaya, "Detection of Hate Speech by Employing Support Vector Machine with Word2Vec Model," in *7th International Conference on Electrical, Electronics and Information Engineering: Technological Breakthrough for Greater New Life, ICEEIE 2021*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 1-5, doi: 10.1109/ICEEIE52663.2021.9616721.

[28] M. Wankhade, A. C. S. Rao, and C. Kulkarni, "A survey on sentiment analysis methods, applications, and challenges," *Artif Intell Rev*, vol. 55, no. 7, pp. 5731–5780, Oct. 2022, doi: 10.1007/s10462-022-10144-1.

[29] A. Toktarova, D. Syrlybay, B. Myrzakhmetova, G. Anuarbekova, G. Rakhimbayeva, and B. Zhylanbaeva, "Hate Speech Detection in Social Networks using Machine Learning and Deep Learning Methods," 2023, pp, 396-406, doi: 10.14569/IJACSA.2023.0140542.

[30] S. Akuma, T. Lubem, and I. T. Adom, "Comparing Bag of Words and TF-IDF with different models for hate speech detection from live tweets," *International Journal of Information Technology (Singapore)*, vol. 14, no. 7, pp. 3629–3635, Dec. 2022, doi: 10.1007/s41870-022-01096-4.

[31] R. Raut and F. Spezzano, "Enhancing hate speech detection with user characteristics," *Int J Data Sci Anal*, pp. 1–11, Aug. 2023, doi: 10.1007/s41060-023-00437-1.

[32] G. Mustafa, M. Usman, M. T. Afzal, A. Shahid, and A. Koubaa, "A comprehensive evaluation of metadata-based features to classify research paper's topics," *IEEE Access*, vol. 9, pp. 133500–133509, 2021, doi: 10.1109/ACCESS.2021.3115148.

[33] N. S. Mullah and W. M. N. W. Zainon, "Advances in Machine Learning Algorithms for Hate Speech Detection in Social Media: A Review," 2021, *Institute of Electrical and Electronics Engineers Inc.* pp. 88364-88376, doi: 10.1109/ACCESS.2021.3089515.

[34] N. Azmi Verdikha, R. Habid, and A. Johar Latipah, "Analisis DistilBERT dengan Support Vector Machine (SVM) untuk Klasifikasi Ujaran Kebencian pada Sosial Media Twitter," *METIK JURNAL*, vol. 7, no. 2, pp. 101–110, Dec. 2023, doi: 10.47002/metik.v7i2.583.

[35] T. Winarti, H. Indriyawati, V. Vydia, and F. W. Christanto, "Performance comparison between naive bayes and k-nearest neighbor algorithm for the classification of indonesian language articles," *IAES International Journal of Artificial Intelligence*, vol. 10, no. 2, pp. 452–457, 2021, doi: 10.11591/IJAI.V10.I2.PP452-457.

[36] G. Muppala and T. Devi, "Accurate Recasting of Giant Text into Charts Using Rapid Automatic Keyword Extraction Algorithm in Comparison with Bag of Words Algorithm," in *Proceedings of International Conference on Contemporary Computing and Informatics, IC3I 2023*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 2548–2552. doi: 10.1109/IC3I59117.2023.10397804.

[37] S. Chawla, R. Kaur, and P. Aggarwal, "Text classification framework for short text based on TFIDF-FastText," *Multimed Tools Appl*, vol. 82, no. 26, pp. 40167–40180, Nov. 2023, doi: 10.1007/s11042-023-15211-5.

[38] G. N. A. Atillo, B. D. Gerardo, and R. P. Medina, "Sentiment Analysis in Product Reviews with Maximum Entropy and Naïve Bayes Using N-gram Method," in *2023 6th International Conference on Information and Communications Technology, ICOIACT 2023*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 522–526. doi: 10.1109/ICOIACT59844.2023.10455843.

[39] S. Shekhar, N. Hoque, and D. K. Bhattacharyya, "PKNN-MIFS: A Parallel KNN Classifier over an Optimal Subset of Features," *Intelligent Systems with Applications*, vol. 14, p. 73, 2022, doi: 10.1016/j.iswa.2022.20.

[40] S. R. Cholil, T. Handayani, R. Prathivi, and T. Ardianita, "IJCIT (Indonesian Journal on Computer and Information Technology) Implementasi Algoritma

Klasifikasi K-Nearest Neighbor (KNN) Untuk Klasifikasi Seleksi Penerima Beasiswa," 2021, pp. 118-127, doi: 10.31294/ijcit.v6i2.10438.

[41] Z. Qavidel Fard, Z. S. Zomorodian, and S. S. Korsavi, "Application of machine learning in thermal comfort studies: A review of methods, performance and challenges," *Energy Build*, vol. 256, p. 111771, 2022, doi: 10.1016/j.enbuild.2021.111771.

[42] E. Helmud, E. Helmud, F. Fitriyani, and P. Romadiana, "Classification Comparison Performance of Supervised Machine Learning Random Forest and Decision Tree Algorithms Using Confusion Matrix," *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 13, no. 1, pp. 92–97, Feb. 2024, doi: 10.32736/sisfokom.v13i1.1985.

[43] R. Yacouby Amazon Alexa and D. Axman Amazon Alexa, "Probabilistic Extension of Precision, Recall, and F1 Score for More Thorough Evaluation of Classification Models," Nov. 2020, pp. 79-91, doi: 10.18653/v1/2020.eval4nlp-1.9.

[44] R. Prasetyo Vincentius and H. Samudra Anton, "Hate Speech Content Detection System on Twitter using K-Nearest Neighbor Method," *AIP Conf Proc*, Apr. 2022, pp. 050001-1-050001-10, doi: 10.1063/5.0080185.

[45] N. H. Cahyana, S. Saifullah, Y. Fauziah, A. S. Aribowo, and R. Drezewski, "Semi-supervised Text Annotation for Hate Speech Detection using K-Nearest Neighbors and Term Frequency-Inverse Document Frequency," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 10, pp. 147–151, 2022, doi: 10.14569/IJACSA.2022.0131020.