ISSN 2614-0047

Comparison of K-Nearest Neighbor and Naïve Bayes algorithms for hoax classification in Indonesian health news

Awang Hendrianto Pratomo [a,1], Faiz Rachmad [a,2,*], Frans Richard Kodong [a,3]

[a] Universitas Pembangunan Nasional Veteran Yogyakarta Caturtunggal, Depok, Sleman Regency, Special Region of Yogyakarta, Indonesia
[1] awang@upnyk.ac.id; [2] faizrachmad@gmail.com; [3] kodongfr@gmail.com
* corresponding author

ARTICLE INFO

Article history
Received August 5, 2024
Revised September 23, 2024
Accepted December 9, 2024

Keywords
Hoax classification
Health news
K-Nearest Neighbor
Naïve bayes
TF-IDF

ABSTRACT

The categorization of health-related hoaxes is paramount in determining if they report facts. This paper analyzes the accuracy of the K-Nearest Neighbor (KNN) and the Naïve Bayes Classifier as two algorithms for health news hoaxes classification. Text mining was employed by feature extraction employing the TF-IDF method from the news headlines to classify the clusters. A prototype model was used to develop the system. Models assessment included confusion matrices and k-fold cross-validation. K=3 KNN model attained an average accuracy of 82.91%, precision of 85.3% and recall of 79.38% with no predictors included. The best performance was recorded for using the Naive Bayes model at fixation of K=3 KNN model at an average accuracy of 86.42%, precision level of 88.10% and recall high of 84.05%. These findings suggest that the KNN surfaces in the last model level rather than in the absence of the Naive Bayes model concerning classifying the hoax position of health news visible through the confusion evaluative matrix. Although related studies have been conducted in the past, this study is dissimilar in terms of its preprocessing methods, size of the data, and outcomes. The dataset consists of 1219 hoaxes labelled and 1227 facts labelled news headlines.

1. Introduction

The dissemination of false information revolves around readers either purposefully spreading the given message or news to other potential victims who would also otherwise do the same, which leads to the circulation of the false information [1], [2]. The effects caused by false news cannot be evaluated instantaneously because these affect the readers' cognitive processes enormously [3]. Readers assimilate false news, and if they are not careful, it will affect their process of cognition [4]. It is clear that with the present-day levels of news dissemination, it would be pretty challenging to determine whether a particular piece of news falls under accurate or hoax news. This is inevitable because when a person reads some news on health, it is almost evident that they have to read the content and, at that rate, it all [5]. The health-related news does, in one way or another, change how one views one's body's health, even though this type of news has not been verified for its credibility.

Classification can utilize various techniques; some methods are hybrid classification approaches that hole and use artificial intelligence, including classification in their research [6]–[8]. The comparative study of k-nearest neighbor (KNN) and Naive Bayes (NB) classifiers reveals distinct strengths and weaknesses in various application domains. Both classifiers are widely used in machine learning for classification tasks, but they differ significantly in their approach and performance depending on the context. This analysis synthesizes insights from several studies to highlight these differences and their implications.

In the context of breast mass classification using mammograms, KNN demonstrated superior performance compared to Naive Bayes. The study found that KNN achieved the highest classification accuracy of 90.4% when using a specific set of discriminative features, outperforming other classifiers including Naive Bayes [9]. This suggests that KNN may be more effective in scenarios where spatial and geometric features are critical.

Naive Bayes has shown competitive performance in landslide susceptibility assessments. In a study comparing various machine learning models, Naive Bayes achieved an area under the curve (AUC) of 0.910, which was slightly lower than the top-performing SVM model but still effective [10]. Another study highlighted that Naive Bayes performed better than logistic regression in terms of accuracy and predictive power, especially when dealing with limited data [11], [12]. This indicates that Naive Bayes can be advantageous in scenarios with smaller datasets or when model simplicity is preferred. Naive Bayes is often used as a baseline in text classification due to its simplicity and efficiency. It performs well with large datasets and is particularly effective when the independence assumption holds [13]. In computationally intensive tasks like searchlight classification analysis in fMRI studies, a fast implementation of Gaussian Naive Bayes was found to be as effective as more complex classifiers like SVM, with the added benefit of reduced computational cost [14].

While KNN can provide high accuracy in certain domains, it is computationally expensive, especially with large datasets, as it requires storing the entire training set and computing distances for each prediction. Naive Bayes, on the other hand, is computationally efficient and works well with high-dimensional data, but its performance can be limited by the independence assumption of features [15], [16].

In conclusion, the choice between KNN and Naive Bayes should be guided by the specific requirements of the application, such as the nature of the data, computational resources, and the importance of model interpretability. While KNN may excel in scenarios requiring detailed spatial analysis, Naive Bayes offers simplicity and efficiency, making it suitable for text classification and situations with limited data. This study assesses KNN as a distance-based method that defines the proximity between documents, while Naïve Bayes uses distance to establish the occurrence of words. This study is capable of comparing these two approaches and succeeds in determining the better accuracy of each approach in the classification of hoax news compared to facts.

2. Method

This section explains the processes followed during this study in line with the research questions. The method employed is sequential and begins with the identification of various sources of data. The acquired text data is then preprocessed for analysis as such data is not presentable. The preprocessed data is then converted into numerical features using the Term Frequency-Inverse Document Frequency (TFIDF) technique. Two machine learning algorithms are employed in the classification of the hoax articles and factual news: K-Nearest Neighbor (KNN) and Naïve Bayes. The last stage involves evaluating the classification models using performance metrics, including accuracy, precision, and recall. The computations are done with the help of a confusion matrix. Details of the activities involved in these stages are provided in the subsections.

2.1. Data Collection

The data in this study employs secondary data collected from the covid19.go.id source, turnback hoax and data from previous studies. So, a dataset of 1134 news categorized as hoaxes and 1721 news categorized as facts is obtained. So, if accumulated, the total dataset used in this study amounted to 2855 data. Tables are examples of Datasets, which can be found in Table 1.

Table.1 News Dataset Example

No	Reviews	Sentiment Labels
1	Minum Air Putih Bisa Atasi Kekentalan Darah Pasien COVID-19	Hoax
2	Varian Baru Covid-19 Terdeteksi Lagi di Inggris, Berpotensi Kebal Antibodi.	Fakta
3	Campuran Garam dan Air Hangat Mampu Hilangkan Virus Covid-19	Hoax
4	Vaksin harus disimpan dalam tempat khusus yang bersuhu rendah agar tidak mudah rusak.	Fakta

2.2. Text Preprocessing

Text processing as show in Table 2 refers to the phase when the acquired text documents are refined to make text data that can be easily analyzed [17]. The text preprocessing conducted in this study has several phases, including but not limited to cleansing (erasing characters and digits) [18]–[20], case folding (transforming uppercase into lowercase) [21], tokenization (breaking up sentences into phrases or words) [22], stopword deletion (deletion of words that add no value based on a stopword list devised) [23], and stemming (reduction of derived words to their root form by eliminating suffixes) [24].

Table.2 Text Preprocessing Example

Preprocessing	Text
Input	Minum Air Putih Bisa Atasi Kekentalan Darah Pasien COVID-19.
Cleansing	Minum Air Putih Bisa Atasi Kekentalan Darah Pasien COVID
Case Folding	minum air putih bisa atasi kekentalan darah pasien covid
Tokenizing	['minum', 'air', 'putih', 'bisa', 'atasi', 'kekentalan', 'darah', 'pasien', 'covid']
Stopword Removal	['minum', 'air', 'putih', 'atasi', 'kekentalan', 'darah', 'pasien', 'covid']
Stemming	['minum', 'air', 'putih', 'atas', 'kental', 'darah', 'pasien', 'covid']

2.3. TFIDF Feature Weighting

The calculation of the TFIDF as show in Table 3 coefficient can be referred to as the stage of determining the weight of each word. This approach relies upon the two concepts related to weight partitioning: term frequency (TF) and inverse document frequency (IDF) [25]. The term fed into the TFIDF formula comes from the last step in the whole preprocessing operation. However, this last sentence is not redundant, and this method combines two concepts for weight calculation, namely term frequency (TF) is the frequency of occurrence of the term (t) in a sentence (d), the document frequency (DF) is the number of sentences where a term (t) appears. The more occurrences of a term within a single document, the greater the weight and less weight when the term is found in several documents.

Table.3 TFIDF Calculation Example

No	Term	df(t)	Nd/df(t)	IDF(t) = Log(Nd/df(t)) +1	W=tf*(IDF+1)			
					D1	*D2*	*D3*	*D4*
1	varian	1	4	1.602	1.602	0	0	0
2	covid	3	1.33	1.123	1.123	0	1.123	1.123
3	inggris	1	4	1.602	1.602	0	0	0
4	potensi	1	4	1.602	1.602	0	0	0
5	kebal	1	4	1.602	1.602	0	0	0
6	antibody	1	4	1.602	1.602	0	0	0
7	vaksin	1	4	1.602	0	1.602	0	0
8	Simpan	1	4	1.602	0	1.602	0	0
9	Khusus	1	4	1.602	0	1.602	0	0
10	Suhu	1	4	1.602	0	1.602	0	0
11	Rendah	1	4	1.602	0	1.602	0	0
12	Mudah	1	4	1.602	0	1.602	0	0
13	Rusak	1	4	1.602	0	1.602	0	0
14	Minum	1	4	1.602	0	0	1.602	0
15	Air	2	2	1.301	0	0	1.301	1.301
16	Putih	1	4	1.602	0	0	1.602	0
17	Atas	1	4	1.602	0	0	1.602	0
18	Kental	1	4	1.602	0	0	1.602	0
19	Darah	1	4	1.602	0	0	1.602	0
20	Pasien	1	4	1.602	0	0	1.602	0
21	Campur	1	4	1.602	0	0	0	1.602

No	Term	df(t)	Nd/df(t)	IDF(t) = Log(Nd/df(t)) +1	W=tf*(IDF+1)			
					D1	*D2*	*D3*	*D4*
22	Garam	1	4	1.602	0	0	0	1.602
23	Hangat	1	4	1.602	0	0	0	1.602
24	Hilang	1	4	1.602	0	0	0	1.602
25	Virus	1	4	1.602	0	0	0	1.602

2.4. K-Nearest Neighbor (KNN)

The K-nearest neighbour (KNN) method is one of the classification techniques that is often employed. Utilization of KNN seeks to categorize the objects that have not been previously known based on the data that is closest to them [26]. In this two-stage approach, the amount of data more accurately called the nearest neighbour is defined by the user and represented by k. The method employed in this case to measure the distance to the nearest neighbor is the Cosine Similarity since it considers the level of precision. The formula for Cosine Similarity calculation may be taken from the equation in 1.

$$\cos(A, B, W) = \frac{\sum_{t=1}^{n}(At*Wt) \times (Bt*Wt)}{\sqrt{\sum_{t=1}^{n}(At*Wt)^2} \times \sqrt{\sum_{t=1}^{n}(Bt*Wt)^2}} \quad (1)$$

At and Bt denote the normalized term frequencies (TF) of word t in documents A and B, respectively, while Wt represents the word's weight. The method of Wandabwa's research may be briefly stated as the following: First, a training matrix R is prepared, containing the number of scalars defining the training document for each row and the number of words for each of the columns, the value of the matrix R(i,j) equals to the number of word j on the document i. After that, the TF-IDF approach normalises the documents and opts for the relevant word weights. Then, the weights of the vector W are estimated. After that, the value of k nearest neighbours for each training document is determined by the computed cosine similarity distance to the nearest document. The smaller the distance between several two documents, a and b, the more significant the cosine similarity score.

2.5. Naïve Bayes Classifier

The Naïve Bayes technique is a method that is used for the prediction of probabilities. This method uses a probability theory of the British scientist Thomas Bayes. This method is used the way it is by predicting future possibilities from the existing information [27]. Naive Bayes classifier is a probability-based method for pushing the rectangles over the surface of the target class of text documents and very rapidly deals with a vast quantity of data.

The accuracy of the classification model developed using the naive Bayes classifier improves with the amount of fresh data collected by the organization and the data selected for training the data. In other words, if the data selected for training can capture all or most of the curly data owned by the system, then the classification system to be created will be of a better level. Once the classification system achieves a better level of performance, then the system can be employed to classify more data. Eq. 2 depicts the Naïve Bayes formula where $P(ci \mid x)\ P(ci|x)\ P(ci \mid x)$ defines the probability of a particular class ci will occur given something has occurred $x, P(ci)\ P(ci)\ P(ci)$ refers to the prevalence of the cif that is independent in the population/ dataset and $P(x \mid ci)\ P(x|ci)\ P(x \mid ci)$ describes the probability of occurrence of x under the given condition that ci is accrued in the training data.

$$P(ci|x) = \frac{P(x|ci)P(ci)}{Px} \quad (2)$$

2.6. Confusion Matrix

Testing is done to find out the extent of the performance of the system built. Commonly used testing methods are Accuracy, Precision and Recall [28]. When testing is carried out, whether the algorithm's performance is effective when applied to the corpus data will be known. When the model can predict correctly all data that becomes training data, and when the model is faced with test data, the model performance of a classification algorithm is determined. Testing is done by creating a Confusion Matrix. Confusion Matrix is a table that states the classification process's correctness level. The confusion matrix table can be seen in Table 4.

Table.4 Confusion matrix

		True Class	
		Positive	*Negative*
Predicted Class	Positive	True Positives (TP)	False Negative (FN)
	Negative	False Positives (FP)	True Negatives (TN)

From Table 4, the Confusion Matrix contains the actual versus predicted classifications and can be divided into four categories: True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN). TP refers to the number of correct predictions where positive values were correctly classified, while FP refers to incorrect predictions where negative values were classified as positive. TN refers to the correct classification of negative values, and FN represents incorrect predictions where positive values were classified as negative. Equations of accuracy, precision, and recall are in 3 to 5.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+FP+FN+TN}} \tag{3}$$

$$\text{Presisi} = \frac{\text{TP}}{\text{TP+FP}} \tag{4}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \tag{5}$$

3. Results and Discussion

This Import Volume considers the metrics and assessment of models used in this study, predominantly the KNN (K-Nearest Neighbor) classifier with several patterns for support (K=1, 3, 5, 7) and Naïve Bayes. The tests were performed using k-fold cross-validation and confusion matrix's evaluations for the model's performance. For classification purposes, 2446 randomly selected and labelled points were used. The outcomes are presented first in a table that displays accuracy, precision, and recall, then in a graphical depiction of the performance of both models.

For the K-Nearest Neighbor model, several values of K were tested. The model was evaluated using k-fold cross-validation with k = 5, meaning the data was split into five-folds, and each fold was used for validation in different iterations. The first set of tests used K = 1, and the test can be seen in Table 5.

Table.5 K-Nearest Neighbor Testing with K=1 value

Fold	Confusion Matrix			
	True Positive	*False Positive*	*True Negative*	*False Negative*
1	190	40	210	50
2	180	33	217	59
3	193	38	208	50
4	197	41	202	49
5	200	41	197	51

The confusion matrix generated for all the model folds, as provided in Table 5, indicates that most positive and negative cases were correctly categorized. This means that the model performed reasonably well in classifying the news titles. A portion of it regarding the accuracy is presented in Table 6, in which the value ranges from 78.76% to 99.90% (average=81.52%), precision from 78.53 to 99.64 (average=83.28), and recall from 76.39 to 94.23 (average 78.73). The results indicate that the model can differentiate hoaxes from factual news articles with an acceptable level of predictiveness when K = 1 is utilized.

Table.6 K-Nearest Neighbor Cross Validation Testing with K=1 value

Fold	Accuracy	Precision	Recall
1	81.63%	82.60%	79.16%
2	81.18%	84.50%	75.31%
3	82.00%	83.54%	79.42%
4	81.59%	82.77%	80.08%
5	81.18%	82.98%	79.68%
Average	81.52%	83.28%	78.73%

Based on the calculation of accuracy, precision and recall in Table 6. The results of testing the K-Nearest Neighbor model with a value of K = 1 used for hoax prediction on health news provide an average accuracy of 81.52%, precision of 83.28% and recall of 78.73% in classifying news in the form of hoaxes or facts.

Next, the KNN model with K = 3 was tested, and the results are summarized in Table 7. Based on the resulting confusion matrix table, it can be seen that for each fold, getting a greater value on the true positive and true negative values proves that the model is good enough to classify news titles. The confusion matrix calculates the accuracy, precision and recall values of the K-Nearest Neighbor algorithm model with a value of K = 3.

Table.7 K-Nearest Neighbor Confusion Matrix Testing with K=3 value

Fold	Confusion Matrix			
	True Positive	*False Positive*	*True Negative*	*False Negative*
1	190	35	215	50
2	184	31	219	55
3	190	29	217	53
4	198	38	205	48
5	206	34	204	45

The following are the results of confusion matrix testing on each iteration, and the average value can be seen in Table 8.

Table.8 K-Nearest Neighbor Cross Validation Testing with K=3 value

Fold	Accuracy	Precision	Recall
1	82.65%	84.44%	79.16%
2	82.41%	85.58%	76.98%
3	83.23%	86.75%	78.18%
4	82.41%	83.89%	80.48%
5	83.84%	85.83%	82.07%
Average	82.91%	85.3%	79.38%

Based on the calculation results of accuracy, precision and recall in Table 8. The test results of the K-Nearest Neighbor model with a value of K = 3 used for hoax prediction on health news provide average accuracy results of 82.91%, precision of 85.3% and recall of 79.38% in classifying news in the form of hoaxes or facts.

Next, the performance of the K-Nearest Neighbor model with K = 5 is detailed in Table 9, which illustrates the results across various folds, highlighting the model's ability to effectively classify news titles based on the metrics of true positives, false positives, true negatives, and false negatives.

Table.9 K-Nearest Neighbor Confusion Matrix Testing with K=5 value

Fold	Confusion Matrix			
	True Positive	*False Positive*	*True Negative*	*False Negative*
1	178	38	212	62
2	182	41	209	57
3	187	33	213	56
4	195	37	206	51
5	210	30	208	41

Based on Table 9, it can be seen that each fold gets a more excellent value on the true positive and actual negative values, proving that the model is good enough to classify news titles. The confusion matrix calculates the accuracy, precision and recall values of the K-Nearest Neighbor algorithm model with a value of K = 5. The following are the results of confusion matrix testing on each iteration, and the average value can be seen in Table 10.

Table.10 K-Nearest Neighbor Confusion Matrix Testing with K=5 value

Fold	Accuracy	Precision	Recall
1	79.59%	82.4%	74.16%
2	79.95%	81.61%	76.15%
3	81.79%	85%	76.95%
4	82.04%	84.05%	79.26%
5	85.48%	87.5%	83.66%
Average	81.76%	84.11%	78.04%

Based on the results of the calculation of accuracy, precision and recall in Table 10. The results of testing the K-Nearest Neighbor model with a value of K = 5 used for hoax prediction on health news provide average accuracy results of 81.76%, precision of 84.11% and recall of 78.04% in classifying news in the form of hoaxes or facts.

The performance of the KNN model with K set to 7 is detailed in Table 11, showcasing how this particular configuration influences the classification accuracy, precision, and recall in distinguishing between hoaxes and factual news articles.

Table.11 K-Nearest Neighbor Confusion Matrix Testing with K=7 value

Fold	Confusion Matrix			
	True Positive	*False Positive*	*True Negative*	*False Negative*
1	185	36	214	55
2	184	40	210	55
3	177	32	214	66
4	190	40	203	56
5	204	29	209	47

Table 11 shows that each fold gets a more excellent value on the true positive and actual negative values, proving that the model is good enough to classify news titles. The confusion matrix calculates the accuracy, precision and recall values of the K-Nearest Neighbor algorithm model with a value of K = 7. The following are the results of confusion matrix testing on each iteration, and the average value can be seen in Table 12.

Table.12 K-Nearest Neighbor Cross Validation Testing with K=7 value

Fold	Accuracy	Precision	Recall
1	81.42%	83.71%	77.08%
2	80.57%	82.14%	76.98%
3	79.95%	84.68%	72.83%
4	80.36%	82.60%	77.23%
5	84.45%	87.55%	81.27%
Average	81.35%	84.14%	77.08%

Based on the calculation results of accuracy, precision and recall in Table 12. The results of testing the K-Nearest Neighbor model with a value of K = 7 used for hoax prediction on health news provide an average accuracy of 81.35%, precision of 84.14% and recall of 77.08% in classifying news in the form of hoaxes or facts.

This evaluation focuses on the Naïve Bayes model, implemented to analyze the classification of health news articles. Employing k-fold cross-validation with k set to 5, the model undergoes five distinct iterations, generating a separate confusion matrix that provides insights into its performance. The results of this rigorous testing process are presented in Table 13.

Table.13 Naïve Bayes Confusion Matrix Testing

Fold	Confusion Matrix			
	True Positive	*False Positive*	*True Negative*	*False Negative*
1	196	30	220	44
2	203	30	220	36
3	203	28	218	40
4	200	26	217	46
5	42	2	45	3

Table 13 shows that each fold gets a more excellent value on the true positive and actual negative values, proving that the model is good enough to classify news titles. The confusion matrix calculates the accuracy, precision and recall values of the Naïve Bayes algorithm model. The following are the results of confusion matrix testing on each iteration, and the average value can be seen in Table 14.

Table.14 Naïve Bayes Cross Validation Testing

Fold	Accuracy	Precision	Recall
1	84.89%	86.72%	81.66%
2	86.50%	87.12%	84.93%
3	86.09%	87.87%	83.53%
4	85.27%	88.49%	81.30%
5	89.36%	90.28%	88.84%
Average	86.42%	88.10%	84.05%

Based on the accuracy, precision, and recall calculations in Table 12, The test results of the Naïve Bayes model used for hoax prediction on health news provide an average accuracy of 86.42%, precision of 88.10% and recall of 84.05% in classifying news in the form of hoaxes or facts.

The following graphs present the relative performance of the two models. In particular, Fig. 1 depicts the performance of the Naïve Bayes model measured in terms of accuracy, precision, and recall across all folds.

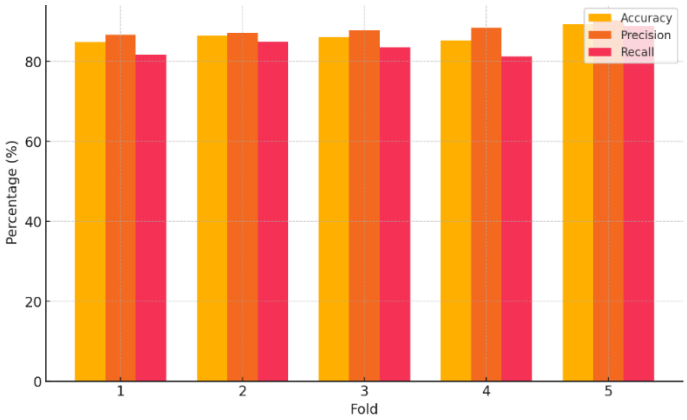

Fig. 1. Graph of Naïve Bayes Method

The graph demonstrates the model's high performance in most areas of the objectives with relative consistency, more so in accuracy and precision. Fig. 2, on the other hand, presents the performance with K = 3 of the KNN model, where the accuracy and precision are also relatively good but not as good as the performance achieved using the Naive Bayes model.

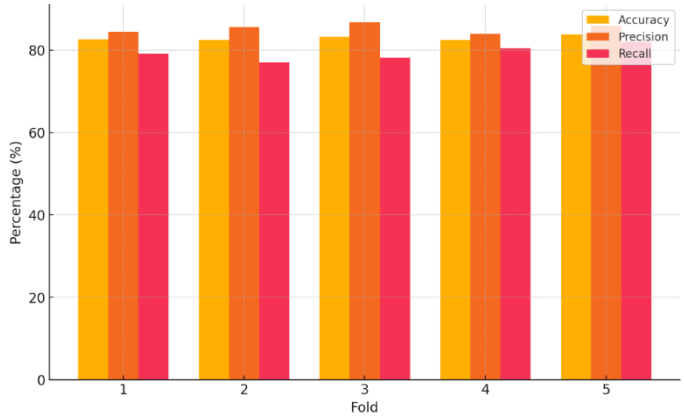

Fig. 2. Graph of the K-Nearest Neighbor Method (K=3)

The Naïve Bayes model provides slightly better results than the KNN model, especially in well-defined accuracy and precision metrics, as indicated above. This implies that Naïve Bayes could handle the current classification better than the KNN even though it mainly concerns textual information where the probability distinguishes a hoax from the fact [29], [30]. Still, KNN does not perform poorly by any means, as it offers respectable performance in all cases and differs only in the values of K used.

However, the benefits of this study are not limited to academic interest. Misinformation is becoming more widespread, especially in health, and therefore, it is essential to build robust classifiers such as the ones developed here. Such detection systems can also be of excellent public health since they can curb the dissemination false information that may lead to adverse behavioural outcomes.

This research reinforces the debates concerning the use of data in addressing issues of media literacy and public information campaigns. With the deployment of sophisticated ML models, news agencies and fact-checking organizations' capacity to address such fabrications will be enhanced. In addition, the results indicate that even though KNN might be helpful, specific models such as Naïve Bayes are more efficient, helping professionals select competent models for comparable objectives.

Such a skill to correctly associate health information into the respective category is also essential from society's view as it helps build an educated society. This study is relevant for improving public knowledge of health issues and consequently promoting health and responsible news consumption by combating misleading information. To conclude, this study highlights the promise of machine learning models in improving the detection of hoaxes in health news, with significant consequences for public health and journalism.

4. Conclusion

From the analysis of the implementation and the test results that have been done, it can be seen that in the K-Nearest Neighbor model with the value of K, which is closest to the neighbour, namely K=1, K=3, K=5, K=7 themselves average result is at K=3, of the results obtained K=3 produce the highest accuracy of 82.91%, precision of 85.3% and recall of 79.38%. This means that the K-Nearest Neighbor model of K = 3 provides faster results when compared with the other models of KNN. While the Naïve Bayes model gets an average accuracy of 86.42%, 88.10% precision and 84.05% recall. This also proves that the average Naïve Bayes constructed was better than the K-Nearest Neighbor constructed at K = 3. The results of the comparison exhibited that when subsequently applied Naïve Bayes model, the accuracy, precision and recall results observed were better as opposed to when the K-Nearest Neighbor method was applied with K equal to three. The recommendation that can be used to improve future works is that in the K-Nearest Neighbor algorithm, an additional algorithm in decision-making should be incorporated for more accuracy. In this analysis using relatively narrow time data, newer news data can be appended for further study because news is available daily.

References

[1] Y. Tsfati, H. G. Boomgaarden, J. Strömbäck, R. Vliegenthart, A. Damstra, and E. Lindgren, "Causes and consequences of mainstream media dissemination of fake news: literature review and synthesis," *Ann. Int. Commun. Assoc.*, vol. 44, no. 2, pp. 157–173, Apr. 2020, doi: 10.1080/23808985.2020.1759443.

[2] M. D. Molina, S. S. Sundar, T. Le, and D. Lee, "'Fake News' Is Not Simply False Information: A Concept Explication and Taxonomy of Online Content," *Am. Behav. Sci.*, vol. 65, no. 2, pp. 180–212, Feb. 2021, doi: 10.1177/0002764219878224.

[3] X. Zhang and A. A. Ghorbani, "An overview of online fake news: Characterization, detection, and discussion," *Inf. Process. Manag.*, vol. 57, no. 2, p. 102025, Mar. 2020, doi: 10.1016/j.ipm.2019.03.004.

[4] N. Rabb, L. Cowen, J. P. de Ruiter, and M. Scheutz, "Cognitive cascades: How to model (and potentially counter) the spread of fake news," *PLoS One*, vol. 17, no. 1, p. e0261811, Jan. 2022, doi: 10.1371/journal.pone.0261811.

[5] J. P. Dillard, R. Li, and C. Yang, "Fear of Zika: Information Seeking as Cause and Consequence," *Health Commun.*, vol. 36, no. 13, pp. 1785–1795, Nov. 2021, doi: 10.1080/10410236.2020.1794554.

[6] M. R. Delavar, "Hybrid machine learning approaches for classification and detection of fractures in carbonate reservoir," *J. Pet. Sci. Eng.*, vol. 208, p. 109327, Jan. 2022, doi: 10.1016/j.petrol.2021.109327.

[7] M. Zgurovsky, V. Sineglazov, and E. Chumachenko, *Artificial Intelligence Systems Based on Hybrid Neural Networks*, vol. 904. Cham: Springer International Publishing, p. 512, 2021, doi: 10.1007/978-3-030-48453-8.

[8] N. Almugren and H. Alshamlan, "A Survey on Hybrid Feature Selection Methods in Microarray Gene Expression Data for Cancer Classification," *IEEE Access*, vol. 7, pp. 78533–78548, 2019, doi: 10.1109/ACCESS.2019.2922987.

[9] H. Singh, V. Sharma, and D. Singh, "Comparative analysis of proficiencies of various textures and geometric features in breast mass classification using k-nearest neighbor," *Vis. Comput. Ind. Biomed. Art*, vol. 5, no. 1, p. 3, Dec. 2022, doi: 10.1186/s42492-021-00100-1.

[10] B. T. Pham, B. Pradhan, D. Tien Bui, I. Prakash, and M. B. Dholakia, "A comparative study of different machine learning methods for landslide susceptibility assessment: A case study of Uttarakhand area (India)," *Environ. Model. Softw.*, vol. 84, pp. 240–250, Oct. 2016, doi: 10.1016/j.envsoft.2016.07.005.

[11] P. Tsangaratos and I. Ilia, "Comparison of a logistic regression and Naïve Bayes classifier in landslide susceptibility assessments: The influence of models complexity and training dataset size," *CATENA*, vol. 145, pp. 164–179, Oct. 2016, doi: 10.1016/j.catena.2016.06.004.

[12] W. Chen, X. Yan, Z. Zhao, H. Hong, D. T. Bui, and B. Pradhan, "Spatial prediction of landslide susceptibility using data mining-based kernel logistic regression, naive Bayes and RBFNetwork models for the Long County area (China)," *Bull. Eng. Geol. Environ.*, vol. 78, no. 1, pp. 247–266, Feb. 2019, doi: 10.1007/s10064-018-1256-z.

[13] S. Xu, "Bayesian Naïve Bayes classifiers to text classification," *J. Inf. Sci.*, vol. 44, no. 1, pp. 48–59, Feb. 2018, doi: 10.1177/0165551516677946.

[14] M. Ontivero-Ortega, A. Lage-Castellanos, G. Valente, R. Goebel, and M. Valdes-Sosa, "Fast Gaussian Naïve Bayes for searchlight classification analysis," *Neuroimage*, vol. 163, pp. 471–479, Dec. 2017, doi: 10.1016/j.neuroimage.2017.09.001.

[15] A. Jamain and D. J. Hand, "The Naive Bayes Mystery: A classification detective story," *Pattern Recognit. Lett.*, vol. 26, no. 11, pp. 1752–1760, Aug. 2005, doi: 10.1016/j.patrec.2005.02.001.

[16] S. Chen, G. I. Webb, L. Liu, and X. Ma, "A novel selective naïve Bayes algorithm," *Knowledge-Based Syst.*, vol. 192, p. 105361, Mar. 2020, doi: 10.1016/j.knosys.2019.105361.

[17] H. Hassani, C. Beneki, S. Unger, M. T. Mazinani, and M. R. Yeganegi, "Text Mining in Big Data Analytics," *Big Data Cogn. Comput.*, vol. 4, no. 1, p. 1, Jan. 2020, doi: 10.3390/bdcc4010001.

[18] K. Stöger, D. Schneeberger, P. Kieseberg, and A. Holzinger, "Legal aspects of data cleansing in medical AI," *Comput. Law Secur. Rev.*, vol. 42, p. 105587, Sep. 2021, doi: 10.1016/j.clsr.2021.105587.

[19] C. P. Chai, "Comparison of text preprocessing methods," *Nat. Lang. Eng.*, vol. 29, no. 3, pp. 509–553, May 2023, doi: 10.1017/S1351324922000213.

[20] M. Siino, I. Tinnirello, and M. La Cascia, "Is text preprocessing still worth the time? A comparative survey on the influence of popular preprocessing methods on Transformers and traditional classifiers," *Inf. Syst.*, vol. 121, p. 102342, Mar. 2024, doi: 10.1016/j.is.2023.102342.

[21] M. A. Rosid, A. S. Fitrani, I. R. I. Astutik, N. I. Mulloh, and H. A. Gozali, "Improving Text Preprocessing For Student Complaint Document Classification Using Sastrawi," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 874, no. 1, p. 012017, Jun. 2020, doi: 10.1088/1757-899X/874/1/012017.

[22] S. Choo and W. Kim, "A study on the evaluation of tokenizer performance in natural language processing," *Appl. Artif. Intell.*, vol. 37, no. 1, Dec. 2023, doi: 10.1080/08839514.2023.2175112.

[23] S. Chanda and S. Pal, "The Effect of Stopword Removal on Information Retrieval for Code-Mixed Data Obtained Via Social Media," *SN Comput. Sci.*, vol. 4, no. 5, p. 494, Jun. 2023, doi: 10.1007/s42979-023-01942-7.

[24] K. Divya, B. Siddhartha, N. Niveditha, and B. Divya, "An Interpretation of Lemmatization and Stemming in Natural Language Processing," *J. Univ. Shanghai Sci. Technol.*, vol. 22, no. 10, pp. 350–357, 2020. [Online]. Available at: https://jusst.org/an-interpretation-of-lemmatization-and-stemming-in-natural-language-processing/.

[25] U. Buatoom, W. Kongprawechnon, and T. Theeramunkong, "Document Clustering Using K-Means with Term Weighting as Similarity-Based Constraints," *Symmetry (Basel).*, vol. 12, no. 6, p. 967, Jun. 2020, doi: 10.3390/sym12060967.

[26] S. Zhang, "Challenges in KNN Classification," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 10, pp. 4663–4675, Oct. 2022, doi: 10.1109/TKDE.2021.3049250.

[27] N. Deepa, J. Sathya Priya, and T. Devi, "Towards applying internet of things and machine learning for the risk prediction of COVID-19 in pandemic situation using Naive Bayes classifier for improving accuracy," *Mater. Today Proc.*, vol. 62, pp. 4795–4799, 2022, doi: 10.1016/j.matpr.2022.03.345.

[28] R. Yacouby and D. Axman, "Probabilistic Extension of Precision, Recall, and F1 Score for More Thorough Evaluation of Classification Models," in *Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems*, Nov. 2020, pp. 79–91, doi: 10.18653/v1/2020.eval4nlp-1.9.

[29] W. H. Bangyal *et al.*, "Detection of Fake News Text Classification on COVID-19 Using Deep Learning Approaches," *Comput. Math. Methods Med.*, vol. 2021, pp. 1–14, Nov. 2021, doi: 10.1155/2021/5514220.

[30] H. D. Cahyono, A. Mahadewa, A. Wijayanto, D. W. Wardani, and H. Setiadi, "Fast Naïve Bayes classifiers for COVID-19 news in social networks," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 34, no. 2, pp. 1033–1041, 2024, doi: 10.11591/ijeecs.v34.i2.pp1033-1041.