p-ISSN: 2301-5373
e-ISSN: 2654-5101

Identification Of Hoax Based On Text Mining Using K-Nearest Neighbor Method

I Wayan Santiyasa[a1], Gede Putra Aditya Brahmantha[a2], I Wayan Supriana[a3], I GA Gede Arya Kadyanan[a4], I Ketut Gede Suhartana[a5], Ida Bagus Made Mahendra[a6]

[a]Informatics Department
[a]Faculty of Math and Natural Sciences, Udayana University
Bali, Indonesia
[1]santiyasa67@gmail.com
[2]adit.hermawan333@gmail.com
[3]wayan.supriana@unud.ac.id
[4]gungde@unud.ac.id
[5]suhartana@unud.ac.id
[6]ibm.mahendra@unud.ac.id

Abstract

At this time, information is very easy to obtain, information can spread quickly to all corners of society. However, the information that spreaded are not all true, there is false information or what is commonly called hoax which of course is also easily spread by the public. the public only thinks that all the information circulating on the internet is true. From every news published on the internet, it cannot be known directly that the news is a hoax or valid one. The test uses 740 random contents / issue data that has been verified by an institution where 370 contents are hoaxes and 370 contents are valid. The test uses the K-Nearest Neighbor algorithm, before the classification process is performed, the preprocessing stage is performed first and uses the TF-IDF equation to get the weight of each feature, then classified using K-Nearest Neighbor and the test results is evaluated using 10-Fold Cross Validation. The working principle of the K-Nearest Neighbor algorithm is that the data testing is classified based on the closest neighbors with the most data training. The test uses the k value with a value of 2 to 10. The value of k states how many neighbors or data are closest to an object that the system could identify. The optimal use of the k value in the implementation is obtained at a value of k = 4 with precision, recall, and F-Measure results of 0.764856, 0.757583, and 0.751944 respectively and an accuracy of 75.4%

Keywords: *k-nearest neighbor, text mining, classification*

1. Introduction

At this time information is very easy to obtain, information can spread quickly to all corners of society. With the rapid development of IT, now with just the tap of a finger, someone can get and share information. However, the information that is spread is not all true, there is false information or what is commonly called Hoax which of course is also easily spread by the public. People only think that all information circulating on the internet is true. Ironically, hoax can make the community itself uneasy and there can even be disputes due to circulating hoaxes.

There are several studies that discuss the identification of hoaxes previously, based on a study entitled The Naïve Bayes Experiment on Detecting Hoax News in Indonesian by Rahutomo, F., Pratiwi, IYR, & Ramadhani DM, 2019. In this study, static tests were performed on 600 random news with the percentage of training and test data of 60% : 40%, 70% : 30%, and 80% : 20% and each percentage is performed three times the test time so that the average value of accuracy is 82.3%, 82.7% and 83%. The percentage of training and test data is 80%:20%, resulting in the highest average accuracy compared to other data percentages. The nave Bayes method can classify online news in Indonesian with an average accuracy of 82.6% for 600 news data [1]. Another research is the K-Nearest Neighbor

Classification Method for Hoax Analysis by Amin, A.I., 2019. The method used is Multinomial Naive Bayes and K-Nearest Neighbor. The documents collected are 300 hoax documents and 300 non-hoax documents in Indonesian. The Documents were taken from websites, broadcast messages and hoax posts on social networks. The best accuracy result with the KNN method is 81.67% and the best accuracy result with the Multinomial Naive Bayes method is 93.33% [2]. The difference from previous researches are this research mainly focus on the classfication result that influenced by the changes of the k value and using different method in preprocessing and weighting.

Based on the existing problems and related research that forms the basis for conducting research, therefore the researcher wants to conduct research that identifies hoaxes using the K-Nearest Neighbor method. The accuracy of hoax identification using the K-Nearest Neighbor method will be influenced by the changes of the k value. The value of k states how many neighbors or data are closest to an object that the system could identify.

2. Research Methods

2.1. Research Stage

The research stages are data collection, preprocessing, TF-IDF weighting, KNN Classification, and evaluation.

2.2. Data Collection

The research uses 740 random content/issue data that has been verified by TurnBackHoax, of which 370 hoax content data and 370 valid content data. The data is obtained by doing a web crawling on the TurnBackHoax site, each content will be converted into a .txt document and labeled (Hoax or valid) according to the manually verified articles performed by TurnBackHoax institution. All data will be divided into training and testing data using the 10-fold cross validation method. In the research conducted, the data used is only content data that uses Indonesian language.

Table 1. Example of Dataset

Tweet	Label
“ Z sampaikan teman2 untuk sementara waktu jgn ke lippo dulu sdh zona merah, 14 orang terpapar d lippo ” dapat info dr teman semoga Ini cm HOAX 😭 😭 😭	hoax
Pemberitaan mengenai PT Dirgantara Indonesia (Persero) dijual ke pihak asing, kami nyatakan HOAX. Berita ini adalah berita bohong yang berulang dari tahun 2017 lalu.	valid

2.3. Preprocessing

The preprocessing stage is a process to prepare raw data before other processes are carried out. In general, preprocessing is done by eliminating inappropriate data or converting data into a form that is easier to process by the system. Preprocessing is very important in conducting sentiment analysis, especially for social media which mostly contains informal and unstructured words or sentences and has a lot of noise [3]. The preprocessing stage consists of the case folding, data cleansing, language normalization, stopwords removal, stemming, and tokenization, those methods are described below:

2.3.1. Case Folding

Case folding is the first stage in preprocessing that aims to replace word by word from uppercase to lowercase letters.

Table 2. Case Folding Process

Before Case Folding	After Case Folding
“ Z sampaikan teman2 untuk sementara waktu jgn ke lippo dulu sdh zona merah, 14 orang terpapar d lippo ” dapat info dr teman semoga Ini cm HOAX 😭 😭 😭	" z sampaikan teman2 untuk sementara waktu jgn ke lippo dulu sdh zona merah, 14 orang terpapar d lippo " dapat info dr teman semoga ini cm hoax 😭 😭 😭

2.3.2. Data Cleansing

Data Cleansing aims to cleans the text by removing irrelevant text such as username, emoticons, links, and so on.

Table 3. Data Cleansing Process

Before Data Cleansing	After Data Cleansing
" z sampaikan teman2 untuk sementara waktu jgn ke lippo dulu sdh zona merah, 14 orang terpapar d lippo " dapat info dr teman semoga ini cm hoax 😭 😭 😭	" z sampaikan teman2 untuk sementara waktu jgn ke lippo dulu sdh zona merah, 14 orang terpapar d lippo " dapat info dr teman semoga ini cm hoax

2.3.3. Language Normalization

Language normalization aims to replace common words abbreviations into the original word and replace non-standard words into standard words.

Table 4. Word List Example for Language Normalization Process

Word Before Language Normalization	Word After Language Normalization
bhs	bahasa
ngomong	berkata
kpd	kepada
nggak	tidak
temen	teman

Table 5. Language Normalization Process

Before Language Normalization	After Language Normalization
" z sampaikan teman2 untuk sementara waktu jgn ke lippo dulu sdh zona merah, 14 orang terpapar d lippo " dapat info dr teman semoga ini cm hoax	" z sampaikan teman untuk sementara waktu jangan ke lippo dulu sudah zona merah, 14 orang terpapar di lippo " dapat info dari teman semoga ini cuma hoax

2.3.4. Stopword Removal

A stopword is a list of not important and not very relevant common words. In this process, these common words are deleted in order to reduce the number of words processed by the system.

Table 6. Stopword Removal Process

Before Stopword Removal	After Stopword Removal
" z sampaikan teman untuk sementara waktu jangan ke lippo dulu sudah zona merah, 14 orang terpapar di lippo " dapat info dari teman semoga ini cuma hoax	" z teman lippo zona merah, 14 orang terpapar lippo " info teman semoga hoax

2.3.5. Stemming

Stemming is changing affixed words into basic words, this proccess is performed using Sastrawi library.

Table 7. Stemming Process

Before Stemming	After Stemming
" z teman lippo zona merah, 14 orang terpapar lippo " info teman semoga hoax	“ z teman lippo zona merah 14 orang papar lippo “ info teman moga hoax

2.3.6. Tokenization

Tokenization is the process of separating words from a text into multiple tokens. This process will remove any spaces too.

Table 8. Tokenization Process

Before Tokenization	After Tokenization
“ z teman lippo zona merah 14 orang papar lippo “ info teman moga hoax	['z', 'teman', 'lippo', 'zona', 'merah', '14', 'orang', 'papar', 'lippo', 'info', 'teman', 'moga', 'hoax']

2.4. Term Frequency Inverse Document Frequency (TF-IDF)

The TFIDF is a method to calculate the weight of each word used in the classfication, this method generally used in information retrieval. Accuracy and efficiency is the key value of this simple method. The method of Term Frequency Inverse Document Frequency (TFIDF) is a method of assigning weighting a relationship a word (term) to a document. The TFIDF is a statistical measure used to evaluate how important a word is in a sentence or a document. For a single document each sentence is considered a document. The frequency with which a word appears in a specific document show how important the word is in the document. Document frequency containing the word indicates how common the word is. The weight of the word gets smaller if it appears in many documents and gets bigger if it appears frequently in a document [1].

The step to find weight with TF-IDF method are:

a. Find term frequency *(tf)*

b. Find weighting term frequency (W_{tf})

$$W_{tf} = \begin{cases} 1 + log10tft,d, jika\ tft,d > 0 \\ 0, \end{cases} \quad (1)$$

c. Find document frequency (df)

d. Find the weight of inverse document frequency (idf)

p-ISSN: 2301-5373
e-ISSN: 2654-5101

$$idf_t = Log\frac{N}{df_t} \quad (2)$$

e. Find the weight of TF-IDF

$$W_{t,d} = Wtf_{t,d}\, x\, idf_t \quad (3)$$

Notes :

$tf_{t,d}$ = term frequency

$Wtf_{t,d}$ = weight of term frequency

df = the number of times the document contains a term

N = the total number of documents.

$W_{t,d}$ = weight of TF-IDF.

2.5. K-Nearest Neighbor

K-Nearest Neighbor (KNN) is a method for classifying objects based on learning data (neighbors) that are closest to the object. Near or far neighbors are usually calculated based on the Euclidean distance. required a classification system as a system capable of finding information. The KNN method is divided into two phases, namely learning (training) and classification or testing (testing). In the learning phase, this algorithm only performs feature vector storage and classification of learning data. In the classification phase, the same features are calculated for the data to be tested (whose classification is unknown). The distance from this new vector to all learning data vectors is calculated, and the k closest neighbors are taken. [5].

The steps are performed as follows :

a. Define the k value.

b. Count the distance of the object of each data group. The distance calculation used the Euclidean distance equation.

$$D(x, y) = \sqrt{\sum_{1=1}^{n}(x_i - y_i)^2} \quad (4)$$

Notes :

D = Distance

x = Train data

y = Test data

c. Classification results obtained.

2.6. K-Fold Cross Validation

In K-fold cross validation, the initial data are randomly partitioned into k mutually different subsets or "folds," D1, D2, ..., Dk, each of the partitions is approximately the same size. Training and testing is performed k times. In iteration i, the partition Di is used as the test set, and the remaining partitions are collectively reserved to train the model. In the first iteration, the subsets D2, ..., Dk collectively used as the training data set to get the first model, which is tested on D1; the second iteration is trained on the subsets D1, D3, ..., Dk and tested on D2; etc. Unlike the holdout and random subsampling methods, here each sample is used the same amount for training and once for testing. In general, 10-Fold Cross validation is great for estimating accuracy due to its relatively low bias and variance. [6]

2.7. Evaluation

The evaluation is performed to find how the classification is performed, in this research, evaluation will be conducted using Confusion Matrix in each iteration of K-Fold Cross Validation.

Confusion matrix is a method used to perform accuracy calculations on the concept of data mining [7]. The following table shows the Confusion Matrix for the two-class classification model.

Table 9. Confusion Matrix

	Prediction result: Valid	Prediction Result: Hoax
Source: Valid	TN	FP
Source: Hoax	FN	TP

In this study, the meaning of entries in the confusion matrix are as follows:

following:

- TP for correct prediction that the document is a Hoax
- TN for correct prediction that the document is Valid
- FP for false prediction that the document is a Hoax
- FN for false prediction that the document is Valid

The formula for calculating system performance using entries from the confusion matrix is as follows:

True Positive Rate = $\frac{TP}{TP+FN}$ (5)

True Negative Rate = $\frac{TN}{TN+FP}$ (6)

Accuracy = $\frac{TP+TN}{TP+TN+FP+FN} \times 100\%$ (7)

Precision = $\frac{TP}{TN+FN} \times 100\%$ (8)

Recall = $\frac{TP}{TP+FN} \times 100\%$ (9)

F-Measure = $\frac{2}{\frac{1}{Precision}+\frac{1}{Accuracy}}$ (10)

Precision is the proportion of positive predicted cases that are also true positives in the actual data. Recall is the proportion of positive cases that are actually predicted to be true positives. [8]

3. Research Results and Discussion

The research uses 740 random content/issue data that has been verified by TurnBackHoax, of which 370 hoax content data and 370 valid content data. The data processed through the preprocessing stage, those are changing all letters to lowercase, and then purging irrelevant text, language normalization, removing stopwords, returning words to their basic form, and finally separating sentences into words. After going through the preprocessing stage, the TF-IDF weighting was performed with following formula (3). After the weights of each words are obtained, classification is performed using the K-Nearest Neighbor.

3.1. The Test of Parameter k

Tests were performed on the KNN Algorithm using K-Fold Cross Validation which resulted in 10 combinations of training data and test data, each partition contains 10% of the data. The data is obtained from the TurnBackHoax Institution which is labeled according to the type of identification manually according to the label on the TurnBackHoax institution. Each combination of training data and test data is included in the KNN process. The tests performed include testing the changes in the value of k starting from k=2, k=3, to k=10, the optimal k value is obtained from the k value which produces the highest accuracy from other k values.

Experiments were performed using the KNN algorithm with the aim of comparing the value of the k parameter and finding the value of k which resulted in higher accuracy in the application of the KNN algorithm and the number of k used was 2,3,4, to 10. Evaluating the test of each change in the value of k in the Classification KNN is performed using the 10-Fold Cross validation method.

Table 10. Tests Result

Neighbors	*Precision*	*Recall*	*F-Measure*	*Accuracy*
2	0.697585	0.736544	0.733128	0.735135
3	0.766831	0.758818	0.752873	0.755405
4	0.764856	0.757583	0.751944	0.754054
5	0.716095	0.698657	0.68765	0.695946
6	0.748055	0.729735	0.720707	0.727027
7	0.703489	0.671075	0.653432	0.668919
8	0.73615	0.707349	0.693543	0.704054
9	0.683088	0.635909	0.60719	0.635135
10	0.684051	0.64859	0.62693	0.647297

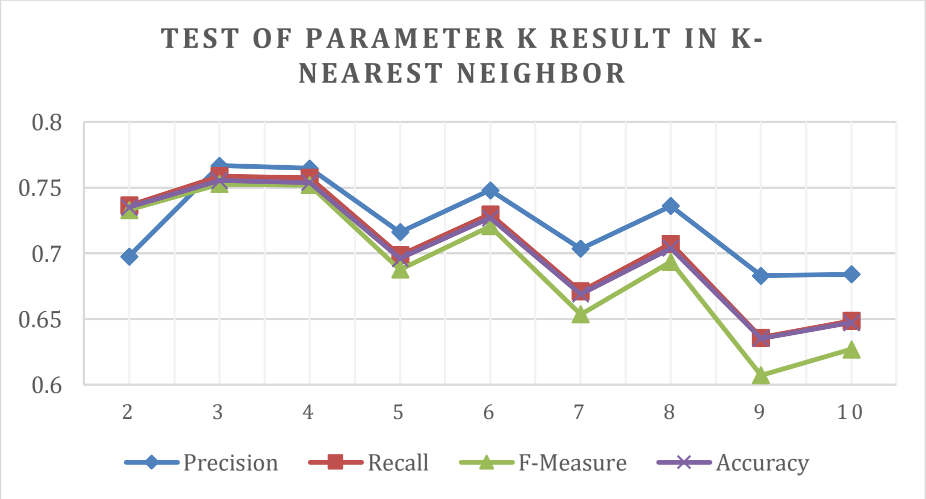

Figure 1. Graph of Test Results

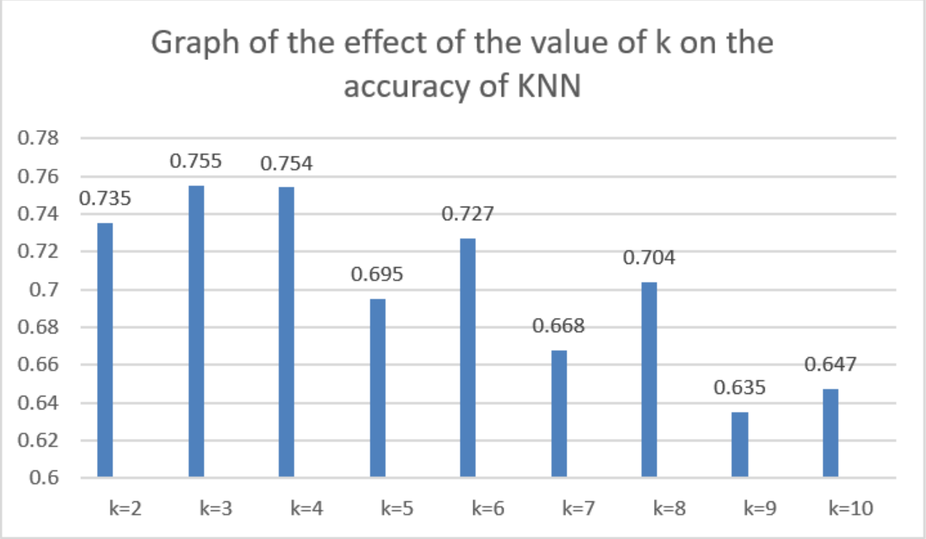

Figure 2. Graph of Test Results

Table 8 shows the results of the research on the effect of the k parameter on the results of the KNN classification. The test starts from k value is 2 which continues until k value is 10. The best results are found when k value is 3 with an average accuracy of 0.755 or 75.5% for each fold and the average F-Measure for each fold is 0.752 or 75 ,2%. From Figure 3, it can be seen that the classification results increase at its peak at k = 3 and then tend to decrease in proportion to the increase in the value of k. Since the value of k is the number of closest neighbors of the classification, tests involving a value of k that are too high cause the training data to be more biased which results in lower accuracy and gives poor results.

The difference in precision between k=3 and k=4 is 0.02 and the difference in accuracy is 0.001 so that the performance of KNN with k=4 is the most optimal k and with a lower F value indicates that the value of k=4 is more significant than k=3 .

From these results, it can be seen the effect of the value of k on accuracy, it can be seen that the accuracy increases from k=2 until the peak point of accuracy is at k=3 and then the accuracy tends to decrease in proportion to the increase in the value of k. In this experiment, the highest accuracy was obtained at 75.5% using the value of k=3 and 75.4% using the value of k=4.

4. Conclusions and Suggestions

From results of the research above that has been completed, it should be noted that the K-Nearest Neighbor method can be used to identify hoaxes in news documents, Testing the value of k is performed with a value of 2 to 10. The optimal use of the k value in the implementation is obtained at a value of k=4 with precision, recall, and F-Measure results of 0.764856, 0.757583, and 0.751944 and an accuracy of 75,4%. In this research the author only discusses K-Nearest Neighbor. Therefore, further researchers can develop other classification methods considering the wide range of classification methods and can be developed by applying them to different fields of science and case studies.

References

[1] F. Rahutomo, I. Y. R. Pratiwi, and D. M. Ramadhani, “Eksperimen Naïve Bayes Pada Deteksi Berita Hoax Berbahasa Indonesia,” J. Penelit. Komun. DAN OPINI PUBLIK, 2019, doi: 10.33299/jpkop.23.1.1805.

[2] A.I. Amin, Metode Klasifikasi K-Nearest Neighbor Untuk Analisis Hoax. Thesis, Bogor Agricultural University (IPB), Bogor. 2019.

[3] S. Mujilahwati, “Pre-Processing Text Mining Pada Data Twitter,” Semin. Nas. Teknol. Inf. dan Komun., vol. 2016, no. Sentika, pp. 2089–9815, 2016.

[4] V. Amrizal, “PENERAPAN METODE TERM FREQUENCY INVERSE DOCUMENT FREQUENCY (TF-IDF) DAN COSINE SIMILARITY PADA SISTEM TEMU KEMBALI INFORMASI UNTUK MENGETAHUI SYARAH HADITS BERBASIS WEB (STUDI KASUS: HADITS SHAHIH BUKHARI-MUSLIM),” J. Tek. Inform., 2018, doi: 10.15408/jti.v11i2.8623.

[5] M. M. Baharuddin, H. Azis, and T. Hasanuddin, “ANALISIS PERFORMA METODE K-NEAREST NEIGHBOR UNTUK IDENTIFIKASI JENIS KACA,” Ilk. J. Ilm., 2019, doi: 10.33096/ilkom.v11i3.489.269-274.

[6] J. Han, M. Kamber, and J. Pei, Data Mining: Concepts and Techniques. 2012.

[7] T. Rosandy, “PERBANDINGAN METODE NAIVE BAYES CLASSIFIER DENGAN METODE DECISION TREE (C4.5) UNTUK MENGANALISA KELANCARAN PEMBIAYAAN (Study Kasus : KSPPS / BMT AL-FADHILA,” J. Teknol. Inf. Magister Darmajaya, vol. 2, no. 01, pp. 52–62, 2016.

[8] D. M. W. Powers, “Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation,” pp. 37–63, 2020, [Online]. Available: http://arxiv.org/abs/2010.16061.

This page is intentionally left blank.

Halaman ini sengaja dikosongkan.