JUSIKOM PRIMA (Jurnal Sistem Informasi dan Ilmu Komputer Prima)
Vol. 6 No. 1, Agustus 2022 E-ISSN : 2580 - 2879

Comparative Analysis of Indonesian Text Mining News Online Classification Using the K-Nearest Neighbor and Random Forest Algorithm

Sarah Tri Yosepha Sitorus[1], Oloan Sihombing*[2], Evta Indra[3], Stiven Hamonangan Sinurat[4], Palma Juanta[5]
[1,2] Program Studi Sistem Informasi, Fakultas Teknologi dan Ilmu Komputer, Universitas Prima Indonesia
Jalan Sampul
E-mail : *oloansihombing@unprimdn.ac.id

ABSTRAK- The rapid development of internet technology today makes many news media grow pretty rapidly. Newspaper companies have utilized internet technology to spread the latest news online through online mass media. Hundreds of thousands of stories are written and published daily on online-based Indonesian news portals, making it difficult for readers to find the news topics they want to read. In making it easier for readers to find the news they are looking for, news needs to be classified according to its respective categories, such as education, current news, finance, and sports. So to classify categories, a text classification method is needed or often called Text Mining. Text mining is a data mining classification technique for processing text using a computer to produce helpful text analysis. In this study, a comparison of 2 methods for developing texts was carried out to get accuracy above 80%.
Kata kunci : Classification, Online News, Random Forest, K-Nearest Neighbor.

1. INTODUCTION

News is information often accessed by the public through many media such as digital media, newspapers, television, and social media [1]. News text information is needed because many today want to keep abreast of information developments in education, the latest news, finance, and sports [2]. The rapid development of internet technology today makes many news media grow quite rapidly [3]. Newspaper companies have utilized internet technology to disseminate the latest news online through online mass media [4][5].

Hundreds of thousands of stories are written and published daily on online-based Indonesian news portals, making it difficult for readers to find the news topics they want to read [6]. In making it easier for readers to find the news they are looking for, news needs to be classified according to its respective categories, such as education, current news, finance, and sports [7].

Before classifying news categories, we need a way to process the news. One way to classify news categories is to use Text Mining [8]. Text mining is a data mining classification technique for processing text using a computer to produce helpful text analysis [9]. At this stage, Text mining has stages that are often referred to as preprocessing (preprocessing) data, where preprocessing (preprocessing) has five stages, namely tokenizing, case folding, stopwords, stemming, and TF-IDF Vectorizer [10].

Research on news classification has been carried out by [11] with the title "Online Classification of News Documents Using Suffix Tree Clustering" by testing the generated suffix tree time and response time required by the application to process news data search using the clustering algorithm, obtaining an accuracy value of 80%.

The second study by [12] with the title "Classification of Sports News Using the Naïve Bayes Method with Enhanced Confix Stripping Stemmer" tested the training and test data. Researchers took 151 training news from 6 categories, namely, football, basketball, racket, MotoGP, Formula 1, and other sports news, while the test data using sports news test documents were taken from the sport.detik.com site randomly with the number 30 news documents. From the results of the experiments carried out, the results obtained an accuracy value of 77% with an error rate of 23%.

Therefore, it is necessary to develop a classification method to get a better accuracy value, so the researchers are interested in conducting a study entitled "Comparative Analysis of Indonesian Text Mining News Online Classification Using the K-Nearest Neighbor and Random Forest Algorithm" it is hoped that the research conducted will get accuracy value above 80%.

2. METHOD

2.1 Method

Classification is the process of finding a set of models or functions that describe and identify data classes to use the model to predict the class of an unknown item [13]. Two procedures involved in classification are constructing a classification model from a predefined set of data classes (training data set), using the model to classify test data, and evaluating the model's accuracy [14]. The classification algorithm used to classify news in this study uses K-Nearest Neighbor and Random Forest.

2.1.1 K-Nearest Neighbor

The KNN algorithm is an algorithm that is often used to classify objects based on training data that has the closest distance to the test data. KNN is also part of the Supervised Learning algorithm [15]. This algorithm will look for various k objects closest to the data to be categorized. After that, the data will be classified into a category by voting on the category with the most significant probability [16].

The following is the equation of Euclidean Distance K-Nearest Neighbor:

$$distance = \sqrt{\sum_{i=1}^{n}(x_{i2} - x_{i1})^2} \text{ [16]}$$

Distance is the distance, x_i2 is the test data, x_i1 is the training data, i is the attribute, and is the number of attributes.

2.1.2 Random Forest

The Random Forest technique is a Learning Ensemble algorithm that uses and builds a Tree Structure. The tree structure is being developed in stages. A decision tree is built by selecting or picking facts at random. In Random Forest, the system selects the results based on the Decision Tree to select the data class [17].

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(fi - yi)^2 \text{ [17]}$$

N is the number of data points, and Fi is The value returned by the Yi model is the actual value for data point i.

2.2 Research Flow

To make the research run on the existing problems, the researcher makes a research flow which can be seen in Figure 2.1, while the research flow to be carried out is as follows.:

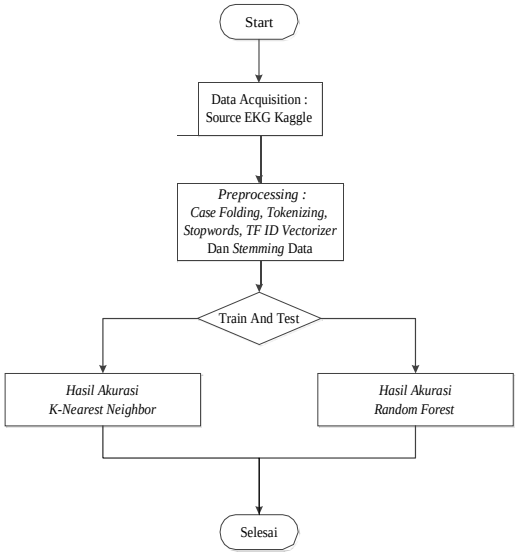

Figure 1 Flowchart

1.Data Acquisition

In this process, the researcher took the dataset of the Indonesian state news sourced from Kaggle published by Eko Prasetio Widhi.

2. Preprocessing

At this stage, the researcher performs the preprocessing stage of the data, which is divided into several processes such as tokenizing, case folding, stopwords, stemming, and TF-IDF Vectorizer.

3. Train and Test

At this stage, the researchers processed the training data distribution and test data to obtain classification accuracy values using the KNN and Random Forest algorithms.

3. RESULT AND DISCUSSION

3.1 Problem Analysis

The need for text mining analysis is significant in dealing with unstructured news texts. One thing that needs to be done in text mining is text classification or categorization. Text mining analysis is carried out to make it easier for us to retrieve or handle information from the internet or the digital world by classifying it with already available data. Text categorization includes many methodologies, such as the K-Nearest Neighbor and Random Forest methods.

3.2 Data Analysis

In classifying Indonesian online news text mining data, it is taken from [18], where the classified data has 2434 rows and five columns. The dataset will be classified into five news categories such as education (Education), finance (Finance), the latest (hot), the latest (news), and sports (sport), according to the article content. The dataset to be processed can be seen in Figure 2:

Figure 2. Dataset

3.3 Data processing

To perform data processing on the dataset, the researcher uses the Python programming language with the help of the existing library on Google Collaboratory. In the text mining classification carried out in this study, the researchers compared the K-Nearest Neighbor and Random Forest algorithms to see the best method for classifying

Text Mining news online.

In data processing, the researchers made a flow chart for data processing, while the data processing flow chart can be seen in Figure 3:

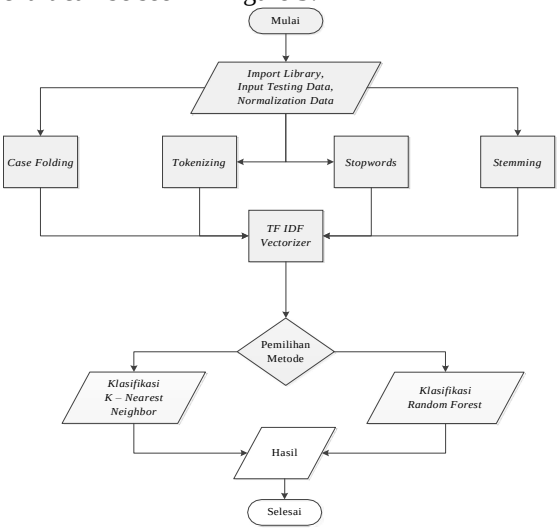

Figure 4. Data Processing Flowchart

3.3.1 Import Librart, Input Testing Data, Data Normalization

1. Import Library

At this stage, the library is called that will be used in data processing, while the library used in this study can be seen in Figure 4:

```
import pandas as pd
import matplotlib.pyplot as plt
import tensorflow as tf
import nltk

from tensorflow import keras
from keras.preprocessing.text import text_to_word_sequence
from nltk.corpus import stopwords
from nltk.stem import PorterStemmer
from nltk.stem import WordNetLemmatizer
from nltk.corpus import wordnet
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.model_selection import train_test_split
from sklearn.metrics import roc_auc_score
from sklearn.pipeline import Pipeline
from sklearn.metrics import classification_report
from sklearn.metrics import accuracy_score
from sklearn.ensemble import RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.preprocessing import LabelEncoder
```

Figure 5. *Import Library*

2. Import Testing Data

At this stage, the dataset is imported into the Google Collaboratory using the Pandas library*:*

Figure 6. *Import Dataset*

3. Normalization Data

At this stage, data normalization or data cleaning is done for processing, and data normalization is done by deleting the Unnamed, Article Title, and Article Link columns.

```
dataset.drop(columns=['Article Title', 'Article Link','Unnamed: 0'],inplace=True)
```

```
dataset
```

	Article Content	Kategori
0	Meski banyak orang menganggap hobi hanya seba...	EDU
1	Institut Teknologi Nasional (ITN) Yogyakarta m...	EDU
2	Pemerintah Provinsi DKI Jakarta melalui Kepala...	EDU
3	Tanaman bakau adalah salah satu ekosistem yang...	EDU
4	Membuat Curriculum Vitae (CV) dan surat lamara...	EDU
...	...	...
2429	Jakarta - Ikatan Motor Indonesia (IMI) melepas...	SPORT
2430	Indonesia Basketball League (IBL) 2022 akhirny...	SPORT
2431	Pemain sepak bola Indonesia, Pratama Arhan men...	SPORT
2432	Indonesia Patriots diharapkan tampil baik di s...	SPORT
2433	Rencana PP PBSI untuk membentuk desentralisasi...	SPORT

Figure 7. Data Normalization

3.3.2 Casefolding

This case folding procedure serves to: change the letters in the comment text into standard form, that is, into lowercase entirely. The results from before and after the case folding process can be seen in Figure 8 and Figure 9:

Figure 8. Before *Casefolding*

```
import re
def casefolding(Casefold):
  Casefold = Casefold.lower()
  Casefold = Casefold.strip(" ")
  Casefold = re.sub(r'[?|$|.|!2_:@\/#")(-+]',"",Casefold)
  return Casefold
dataset['Article Content'] = dataset['Article Content'].apply(casefolding)
dataset.head(1)
```

Figure 9. After Case Folding

3.3.3 Tokenizing

Stemming in Indonesian is relatively tricky. Various kinds of affixes include prefixes, suffixes, infixes, and confixes. Words in Indonesian also come from the repetition of words, combinations of affixes, and combinations of affixes with repeated words. In addition, Indonesian features compound words written together when tied at the beginning and end.

Figure 10. Before *Tokenizing*

```
#tokenizing

def Token(ArticleContent):
    nstr =ArticleContent.split(',')
    dat=[]
    a = -1
    for hu in nstr:
        a = a + 1
    if hu == '':
        dat.append(a)
    p = 0
    b = 0
    for q in dat:
        b = q - p
        del nstr[b]
        p = p + 1
    return nstr
dataset['Article Content'] = dataset['Article Content'].apply(token)
dataset.head(1)
```

Figure 11. After *Tokenizing*

3.3.4 Stopword

In this procedure, words that do not have this effect will be removed, such as words that occur too often compared to other words. The following stopword process can be seen in Figure 12:

Figure 12. Before Filtering

```
#proses filtering

import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords

def stopword_removal(stopp):
    filtering = stopwords.words('indonesian','english')
    x = []
    data = []
    def myFunc(x):
      if x in filtering:
        return False
      else:
        return True
    fit = filter(myFunc, stopp)
    for x in fit:
        data.append(x)
    return data
dataset['Article Content'] = dataset['Article Content'].apply(stopword_removal)
dataset.head(1)
```

Figure 13. After Filtering

3.3.5 Stemming

Stemming adalah mengubah kata menjadi kata dasar, dimana imbuhan seperti in, ke dan sebagainya akan diubah menjadi kata dasar. Berikut ini proses Stemming dapat dilihat pada Figure 14:

Figure 14. Before Stemming

```
# proses stemming

from sklearn.pipeline import Pipeline
!pip install Sastrawi
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

def stemming(Stemm):
  factory = StemmerFactory()
  stemmer = factory.create_stemmer()
  do = []
  for w in Stemm:
      dt = stemmer.stem(w)
      do.append(dt)
  d_clean=[]
  d_clean= "".join(do)
  print(d_clean)
  return d_clean
dataset['Article Content'] = dataset['Article Content'].apply(stemming)
```

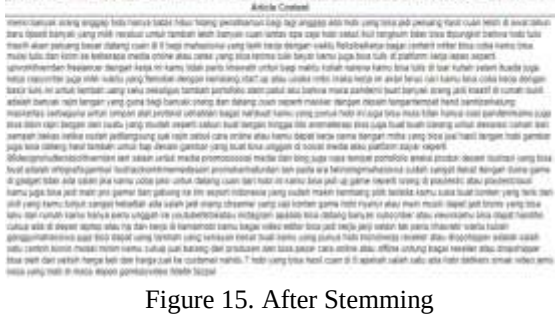

Figure 15. After Stemming

3.3.6 TF- IDF VECTORIZER

At this stage, the data will be weighted so that interaction occurs between each data. The system will test the existing training data so that the results will be more diverse.

```
#proses TF IDF

from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer

tf = TfidfVectorizer()
text_tf = tf.fit_transform(datastem['Article Content'].astype('U'))
text_tf
```

Figure 16. Source Code TF-IDF

3.3.7 Classification Results

After processing the text mining data processing stages on the news classification as many as 2434 article texts, the results obtained are that the education category is 821, Finance is 235, Hot is 756, News is 508, and Sport is 114.

```
dataset.groupby('Kategori').KategoriId.count().plot.bar(ylim=0)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fc04d441050>
```

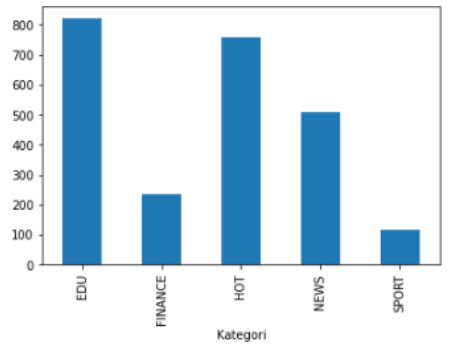

Figure 17Category Classification Results

3.4 Train and Test

In this process, the distribution of training data and test data uses machine learning with a ratio of 70: 30 using the standard 60 shuffle actual random state from google collaborative:

```
X_train, X_test, Y_train, Y_test = train_test_split(text,category, test_size = 0.3,
                                                    random_state = 60,shuffle=True, stratify=category)

print(len(X_train))
print(len(X_test))
```

```
1703
731
```

Figure 18. Split Data Test Dan Data Train

3.4.1 Training K-Nearest Neighbor

In this section, KNN validation will be carried out using k-fold cross-validation. At this stage, the data is separated into 2, namely data validation and training data as a learning process.

```
knn = Pipeline([('tfidf', TfidfVectorizer()),
                ('knn', KNeighborsClassifier()),
               ])

knn.fit(X_train, Y_train)

test_predict = knn.predict(X_test)

train_accuracy = round(knn.score(X_train,Y_train)*100)
test_accuracy =round(accuracy_score(test_predict, Y_test)*100)

print("K-Nearest Neighbour Train Accuracy Score : {}% ".format(train_accuracy ))
print("K-Nearest Neighbour Test Accuracy Score  : {}% ".format(test_accuracy ))
print()
print(classification_report(test_predict, Y_test, target_names=target_category))
```

Figure 19. Source Code Train And Test KNN

```
K-Nearest Neighbour Train Accuracy Score : 93%
K-Nearest Neighbour Test Accuracy Score  : 89%

              precision    recall  f1-score   support

         EDU       0.94      0.89      0.91       261
         HOT       0.72      0.77      0.74        66
     FINANCE       0.94      0.92      0.93       233
        NEWS       0.83      0.88      0.85       144
       SPORT       0.74      0.93      0.82        27

    accuracy                           0.89       731
   macro avg       0.83      0.88      0.85       731
weighted avg       0.89      0.89      0.89       731
```

Figure 20. KNN Accuracy Results

Figure 20 shows that for the classification results using the K-Nearest Neighbor algorithm, the test accuracy is 89%, and the Train accuracy is 93% with precision, recall, f1-score, and support values in each category classified.

3.4.2 Training Random Forest

One of the best methods in machine learning uses decision trees or decision trees to carry out the selection process, where the tree or decision tree will be split recursively depending on the data in the same class. In this scenario, using more trees will increase the accuracy achieved to be more ideal. Determining categorization using Random Forest is based on the voting results and the resulting tree.

```
rfc = Pipeline([('tfidf', TfidfVectorizer()),
                ('rfc', RandomForestClassifier(n_estimators=100)),
               ])

rfc.fit(X_train, Y_train)

test_predict = rfc.predict(X_test)

train_accuracy = round(rfc.score(X_train,Y_train)*100)
test_accuracy =round(accuracy_score(test_predict, Y_test)*100)

print("Random Forest Train Accuracy Score : {}% ".format(train_accuracy ))
print("Random Forest Test Accuracy Score  : {}% ".format(test_accuracy ))
print()
print(classification_report(test_predict, Y_test, target_names=target_category))
```

Figure 21 Source Code Train And Test Random Forest

```
Random Forest Train Accuracy Score : 100%
Random Forest Test Accuracy Score  : 85%

              precision    recall  f1-score   support

         EDU       0.98      0.91      0.94       265
         HOT       0.44      0.97      0.60        32
     FINANCE       0.99      0.78      0.87       286
        NEWS       0.76      0.82      0.79       142
       SPORT       0.18      1.00      0.30         6

    accuracy                           0.85       731
   macro avg       0.67      0.90      0.70       731
weighted avg       0.91      0.85      0.87       731
```

Figure 22 Hasil Akurasi Random Forest

Figure 22 shows that for the classification results using the Random Forest algorithm, the test accuracy is 100%, and the Train accuracy is 85% with the values of precision, recall, f1-score, and support for each category that is classified.

3.5 Algorithm Comparison

After the train and test process is carried out to get the accuracy results, the next step is to compare or compare the accuracy values, which aims to see

the highest accuracy value in the two algorithms used in classifying online news text mining.

For train data accuracy values in the two algorithms, the difference in values is not too far, namely 7%, where the Random Forest algorithm obtains the highest train accuracy value with a train accuracy value of 100% while K-Nearest Neighbor obtains a train accuracy value of 93%. In the test accuracy values, the two algorithms have a difference that is not too far from the train data, which is 4%, where the K-Nearest Neighbor algorithm reaches a test accuracy value of 89%, and the Random Forest algorithm reaches an accuracy value of 85%. A comparison of the accuracy values of the two algorithms can be seen in table 3.1:

```
K-Nearest Neighbour Train Accuracy Score : 93%
K-Nearest Neighbour Test Accuracy Score  : 89%

Random Forest Train Accuracy Score : 100%
Random Forest Test Accuracy Score  : 85%
```

Figure 23 Accuracy Comparison

4. CONCLUSSION

4.1 Conclusion

In classifying the online news text category, the dataset has a total of 2434 rows and 5 columns which are classified into 5 news categories such as education (Education), finance (Finance), latest (hot), latest (news), and sports (sport) according to content. Article content. Using the KNN and Random Forest algorithms get the highest accuracy of 89% on the Random Forest algorithm and 85% on the K-Nearest Neighbor algorithm.

4.2 Suggestions

It is hoped that further research will develop algorithms or methods for classifying online news to get a better accuracy value, and researchers expect to detect authenticity or fake news on classified only.

BIBLIOGRAPHY

[1] L. G. Irham, A. Adiwijaya, and U. N. Wisesty, "Klasifikasi Berita Bahasa Indonesia Menggunakan Mutual Information dan Support Vector Machine," J. Media Inform. Budidarma, vol. 3, no. 4, p. 284, 2019, doi: 10.30865/mib.v3i4.1410.

[2] S. Al Faraby, F. Informatika, U. Telkom, and D. Frekuensi, "Analisis Dan Implementasi Support Vector Machine Dengan String Kernel Dalam Melakukan Klasifikasi Berita Berbahasa Indonesia Analysis and Implementation Support Vector Machine With String Kernel for Classification indonesian news," e-Proceeding Eng., vol. 5, no. 1, pp. 1701–1710, 2018.

[3] W. F. Mahmudy and A. W. Widodo, "Klasifikasi Artikel Berita Menggunakan Naive Bayes Classifier yang Dimodifikasi," Tekno, vol. 21, pp. 1–10, 2014.

[4] H. Mustofa and A. A. Mahfudh, "Klasifikasi Berita Hoax Dengan Menggunakan Metode Naive Bayes," Walisongo J. Inf. Technol., vol. 1, no. 1, p. 1, 2019, doi: 10.21580/wjit.2019.1.1.3915.

[5] H. Muhabatin, C. Prabowo, I. Ali, C. L. Rohmat, and D. R. Amalia, "Klasifikasi Berita Hoax Menggunakan Algoritma Naïve Bayes Berbasis PSO," INFORMATICS Educ. Prof. J. Informatics, vol. 5, no. 2, p. 156, 2021, doi: 10.51211/itbi.v5i2.1531.

[6] "Klasifikasi Berita Indonesia Menggunakan Metode Naive Bayesian Classification dan Support Vector Machine dengan Confix Stripping Stemmer.pdf." .

[7] B. S. Prakoso, D. Rosiyadi, H. S. Utama, and D. Aridarma, "Klasifikasi Berita Menggunakan Algoritma Naive Bayes Classifer Dengan Seleksi Fitur Dan Boosting," J. RESTI (Rekayasa Sist. dan Teknol. Informasi), vol. 3, no. 2, pp. 227–232, 2019, doi: 10.29207/resti.v3i2.1042.

[8] B. Kurniawan, S. Effendi, and O. S. Sitompul, "Klasifikasi Konten Berita Dengan Metode Text Mining," J. Dunia Teknol. Inf., vol. 1, no. 1, pp. 14–19, 2012, [Online]. Available: http://download.portalgaruda.org/article.php?
article=58993&val=4123.

[9] B. K. Palma, D. T. Murdiansyah, and W. Astuti, "Klasifikasi Teks Artikel Berita Hoaks Covid-19 dengan Menggunakan Algotrima K- Nearest Neighbor," eProceedings …, vol. 8, no. 5, pp. 10637–10649, 2021, [Online]. Available: https://openlibrarypublications.telkomuniversity.ac.id/index.php
/engineering/article/view/15672.

[10] J. Kelvin, Prima et al., "Analisis Perbandingan Sentimen Corona Virus Disease-2019 (Covid19) Pada Twitter Menggunakan Metode Logistic Regression Dan Support Vector Machine (Svm)," vol. 5, no. 2, 2022.

[11] R. Darwanto, A. Z. Arifin, and H. T. Ciptaningtyas, "Klasifikasi Online Dokumen Berita Dengan Menggunakan Suffix Tree Clustering,"Klasifikasi Online Dokumen Berita Dengan Menggunakan Suffix Tree Clustering. pp.1–8.

[12] Y. D. Pramudita, S. S. Putro, and N. Makhmud, "Klasifikasi Berita Olahraga Menggunakan Metode Naïve Bayes dengan Enhanced Confix Stripping Stemmer," J. Teknol. Inf. dan Ilmu Komput., vol. 5, no. 3, p. 269, 2018, doi: 10.25126/jtiik.201853810.

[13] S. Kurniawan, W. Gata, D. A. Puspitawati, N. -, M. Tabrani, and K. Novel, "Perbandingan Metode Klasifikasi Analisis Sentimen Tokoh Politik Pada Komentar Media Berita Online," J. RESTI (Rekayasa Sist. dan Teknol. Informasi),

vol. 3, no. 2, pp. 176–183, 2019, doi: 10.29207/resti.v3i2.935.

[14] H. Rhomadhona and J. Permadi, "Klasifikasi Berita Kriminal Menggunakan NaÃ¯ve Bayes Classifier (NBC) dengan Pengujian K-Fold Cross Validation," J. Sains dan Inform., vol. 5, no. 2, pp. 108–117, 2019, doi: 10.34128/jsi.v5i2.177.

[15] D. A. Fauziah, A. Maududie, and I. Nuritha, "Klasifikasi Berita Politik Menggunakan Algoritma K-nearst Neighbor," Berk. Sainstek, vol. 6, no. 2, p. 106, 2018, doi: 10.19184/bst.v6i2.9256.

[16] A. N. Kasanah, M. Muladi, and U. Pujianto, "Penerapan Teknik SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Objektivitas Berita Online Menggunakan Algoritma KNN," J. RESTI (Rekayasa Sist. dan Teknol. Informasi), vol. 3, no. 2, pp. 196–201, 2019, doi: 10.29207/resti.v3i2.945.

[17] B. S. Prakoso, D. Rosiyadi, D. Aridarma, H. S. Utama, F. Fauzi, and M. A. N. Qhomar, "Optimalisasi Klasifikasi Berita Menggunakan Feature Information Gain Untuk Algoritma Naive Bayes Terhubung Random Forest," J. Pilar Nusa Mandiri, vol. 15, no. 2, pp. 211–218, 2019, doi: 10.33480/pilar. v15i2.684.

[18] P. E. Widhi, "News Indonesian Classification | Kaggle." https://www.kaggle.com/datasets/ekoprasetiowidhi5/news-indonesian classification(accessed Jun. 19, 2022).