Jurnal Ilmiah
Teknologi Sistem
Informasi

Jitsi

Vol. 5 (2024) No. 4

https://jurnal-itsi.org/index.php/jitsi; E-ISSN: 2722-4600; ISSN: 2722-4619
DOI: 10.62527/jitsi.5.4.287

Hoax News Detection in Indonesian Political Headlines Using Multinomial Naive Bayes

Bertrand Baldomero Ferguson [#], Wirawan Istiono [#]

[#] *Department of Informatics Universitas Multimedia Nusantara, Jalan Scientia Boulevard Gading, Curug Sangereng Serpong, Kabupaten Tangerang, Banten 15810, Indonesia*
E-mail: bertrand.ferguson[at]student.umn.ac.id, wirawan.istiono[at]umn.ac.id

A B S T R A C T S

Social media is a means of online social interaction on the Internet, where users can freely share information. Because of the freedom, it cannot be denied that some people will misuse social responsible for misusing social media as a place to spread false news. Based on a survey of 2,032 respondents conducted by DailySocial.id in 2018, it was concluded that the majority of Indonesians do not have the ability to detect hoax news. Therefore, the research aims to design and build a hoax news detection application using the Android-based Multinomial Naive Bayes algorithm. At the design stage, the application is designed to receive input in the form of textual political news headlines. It then uses the Multinomial Naive Bayes algorithm to detect hoaxes by comparing the resulting text with data sets. In the testing phase, the algorithm is tested on a confusion matrix and shows the degree of hoax detection. The accuracy of the hoax detection is 88.9%, the precision is 93.33%, the recall is 84%, the recall is 84%, and the F1 score is 88.4%. With a detection application, it is hoped that this hoax news will be able to contribute to the online environment of the Indonesian people by verifying the information before sharing it on social media.

Manuscript received Oct 4, 2024; revised Oct 14, 2024. accepted Nov 18, 2024 Date of publication Dec 30, 2024. International Journal, JITSI : Jurnal Ilmiah Teknologi Sistem Informasi licensed under a Creative Commons Attribution-Share Alike 4.0 International License

Keywords / Kata Kunci — ***Confusion matrix; Fake news; Hoax detection; Indonesian Political Headlines; Multinomial Naïve Bayes algorithm***

CORRESPONDING AUTHOR

Wirawan Istiono
Department of Informatic, Universitas Multimedia Nusantara, Jalan Scientia Boulevard Gading, Curug Sangereng Serpong, Kabupaten Tangerang, Banten 15810, Indonesia
Email: wirawan.istiono[at]umn.ac.id

1. INTRODUCTION

The rapid development of technological innovation has made it easier for society to obtain information quickly, anytime and anywhere. All the latest information from different parts of the world can be accessed through the Internet. Technological advancements have made it easier to access online media through smartphones, laptops, tablets, computers, etc. One of the most popular online media used by almost everyone is social media, which users use to communicate, freely share information and interact with others [1], [2]. Due to the freedom that individuals have in uploading/distributing both positive and negative content, it cannot be denied that some irresponsible individuals misuse social media as a platform to spread unverified information or fake news [3], [4].

Based on survey conducted in 2019 by Mastel.id online with 941 respondents, the percentage of fake news spread on social media in Indonesia is 92,4%. A survey on how often people in Indonesia receive fake news showed that 14,7% of respondents received fake news more than once a day, 34,6% of respondents receive fake news daily, 32,5% of respondents receive fake news once a week, and 18,2% of respondents receive fake news once a month [5]. Separately, there is information from the official website of the Research and Development Agency of the Ministry of Home Affairs litbang.kemendagri.go.id regarding a survey conducted by DailySocial.id

on 2.032 respondents in 2018, it was recorded that as 44,19% of respondents admitted that they were not sure they had the ability to detect fake news. Meanwhile, another 51,03% chose to remain silent when they found fake news on social media [6]. From the result of this research, it can be concluded that the majority of Indonesian people do not have the ability to detect fake news [7]. According to Danu Nur Irwanto, currently the technology to detect fake news in Indonesia is still rarely found, some applications to prevent fake news are only sites for reporting fake news [8]. Due to the increasing amount of online content, including news and information that is widely disseminated on social media, news verification can more effective if supported by AI technology, which can help process large amounts of data quickly and efficiently compared to manual fact checking [9].

The use of the Multinomial Naive Bayes algorithm in the creation of models for the detection of hoaxes is due to the fact that this algorithm is suitable for text processing, especially for the classification of texts whose feature is the frequency of each word that appears. This algorithm can handle quite large datasets, besides providing accurate results, especially for datasets whose complexity is not too complicated [10]. The Multinomial Naive Bayes variant was used because it has been shown that MNB can reduce the error rate by 27% compared to other Naive Bayes variants [11], [12]. Similar research was previouly conducted by Candra Surya Sriyano and Erwin Budi Setiawan in 2021, titled "Detection of Hoax News Using Multinomial Naive Bayes on Twitter with TF-IDF Weighting Feature", which produced model accuracy results of 72,06% [13], [14]. Previous researchers used data crawled from Twitter, and previous researchers said that the accuracy results were not optimal because previous researchers did not preprocess the data properly (there were still symbols/links in the data used) [15], [16]. To create a machine learning model, the data that will be used is in the form of news headlines, which will be used as samples of hoax news labeled political obtained from the Turnbackhoaks site and for original news labeled political using data from the media sites Kompas, CNN, Detik. The creation of a hoax news detection model using news headlines was created because of the clickbait culture that occurs in the digital era, which is exploited by individual journalists by using news headlines that really arouse readers curiousity, but the content of the news does not match the written title. This makes readers feel that they are reading fake and misleading news because it does not meet their expectations and they feel disadvantaged [17], [18]. Title dataset obtained by independently conducting website scraping of 3.975 entities. After the model is created, the accuracy of the model is checked using the confusion matrix by determining the values of accuracy, precision, recall and f1-score.

This study aims to know how to design and build a hoax news detection application using the Android-based Naive Bayes Algorithm. The next research objective is to calculate the level of accuracy, precision, recall and f1-score of the Naive Bayes algorithm in hoax news detection applications. The limitation of the problem in this research is that it can only detect hoax news in Android-based applications using news titles, hoax news detection can only detect news in text form using Indonesian language, the machine learning model is only able to detect political news in Indonesia, and the dataset is obtained by scraping websites from Kompas, CNN, Detik news for original news and Turnbackhoax for hoax news. The benefits of creating a hoax news detection application are to identify false or misleading information, thereby helping to prevent the rapid spread of hoax news, increasing contribution to the online environment of Indonesian society by first verifying information before sharing it on social media and preventing individuals and the group become victims of fraud or other criminal activities from the hoax news received.

2. RESEARCH METHODOLOGY

2.1 CountVectorizer

CountVectorizer is one of the machine learning libraries that helps in the process of building natural language processing models by extracting text features [19]. CountVectorizer extracts text features by transforming text into a matrix that contains the frequency of occurrence of each word in the text. CountVectorizer can transform text features into a vector representation [20].

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

TF-IDF (Term Frequency-Inverse Document Frequency) is a method used to determine the frequency value of a word in a document/article as well as its frequency across various documents. The TF-IDF algorithm is typically used in processing large datasets [21], [22]. This algorithm assigns weights to each keyword in each category to find similarities between keywords and existing categories. Before weighting, text processing is performed, followed by TF-IDF weight calculation, query relevance weight, and similarity weight. The calculation of TF-IDF weight begins with calculating the Term Frequency (TF), which simply counts the occurrences of each word. Next, it calculates the Inverse Document Frequency (IDF), which counts the number of documents containing that word [23], [24]. If the word is found more frequently in other documents, it will be considered less important. The mathematical calculation for TF-IDF is as follows [22]:

$$W_{dt} = tf_d \times idf_t \quad (1)$$

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

Description:
W_{dt} = The weight value of the t-word in document d
tf_d = The number of occurrences of t-word in document d
N = Total number of documents
df_t = Number of documents that use t-word
d = Document d
t = t-word

2.3. Multinomial Naive Bayes Algorithm

Multinomial Naive Bayes is a variation of the Naive Bayes algorithm used for text/document classification with good accuracy results. The formula used to compute the class for a document is [16]:

$$C_{\text{map}} = \arg\max_{c \in C} P(c) \prod_{n=1}^{k} P(t_n \mid c) \quad (3)$$

Description :
C_{map} = Probability of a document belonging to class c
$P(c)$ = Prior probability of class c
$P(t_n|c)$ = Probability of the n-word given class c

$$P(t_n \mid c) = \frac{W_{ct}+1}{(\sum W' \epsilon V W')+B'} \quad (4)$$

Description :
W_{ct} = TF-IDF weighting value or W of word t in class c
$\sum W' \in V\, W'$ = Total sum of W from all words in class c
B' = Total W from unique words

2.4. Confusion Matrix

A confusion matrix is a matrix used to evaluate the performance of a built classification model. This matrix compares the actual values with the predicted results of the machine learning model. The size of the matrix depends on the number of target classes of the labels to be predicted. Below is an illustration of the confusion matrix shown in Table 1 [25], [26].

TABLE 1. Confusion matrix table

Reinforcement Condition	Classifier says YES	Classifier says NO
In reality YES	True Positive	False Negative
In reality NO	False Positive	True Negative

Generally, a confusion matrix is used to generate four values: accuracy, precision, recall, and F1 score [14]. Accuracy represents how accurately the model classifies, expressed as a percentage [15].

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

Precision describes the agreement between the requested data and the predictions made by the model [15].

$$Precision = \frac{TP}{TP+FP} \quad (6)$$

Recall describes the success of the model in retrieving specific information [15].

$$Recall = \frac{TP}{TP+FN} \quad (7)$$

The F1-score represents the weighted average comparison of precision and recall [15].

$$F1 = \frac{2\times(Precision\times Recall)}{Precision+Recall} \quad (8)$$

The research flow used in the design and development of a hoax news detection application using the Naive Bayes algorithm on an Android platform from start to finish can be seen in Figure 1.

1. Literature Review
   The first stage begins with finding and studying topics related to the research. This process involves learning from articles and journals on the Internet that are relevant to the sources used in the research. The purpose of this stage is to provide a theoretical basis for solving the research problem.
2. Data Collection
   In the second stage, data is collected to create a dataset of news articles with political topics in Indonesia through website scraping from online media such as Kompas, CNN, and Detik for valid news and Turnbackhoax for hoax news.

3. Application Design
   This stage involves designing the initial concept of the application, including creating flowcharts, designing the user interface, and structuring the database.
4. Programming and algorithm implementation on the Android application
   This stage begins with the development of the application based on the design and predetermined features. Implementation of the algorithm into the application is also included in this phase.
5. Application testing
   The next stage, after the development of the application and algorithm model is completed, is to test the application to ensure that the features and algorithm implementation work correctly. Black box testing is used for system testing, while model accuracy testing is performed using confusion matrix calculations.

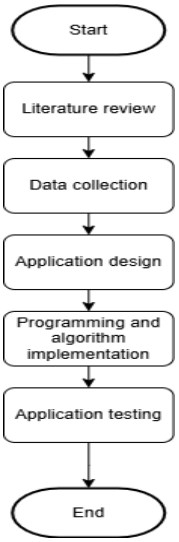


FIG 1. Research Flow Diagram

2.4. Requirements Analysis

The analysis of the requirements for this Android-based fake news detection application is divided into two parts: input requirements and output requirements. Input requirements include information that users need to enter into the application and the text of political news headlines to be checked. Information required for account registration includes user name, email, and user password. Output requirements include the results generated by the application, specifically the result of the news check, which indicates whether the headline is classified as fake news (hoax) or real news (not a hoax).

The hoax news detection application provides login and registration features to enhance functionality and user interaction. To access the application's features, users must register an account (for new users) or log in (for existing users). After logging in, users can use the hoax detection feature and also view the history of previous news checks. To ensure that these features work smoothly, the hoax detection application is designed with two main systems: front-end and back-end. Authentication and authorization processes, such as login and registration, as well as the hoax news checking process, which uses the Multinomial Naive Bayes algorithm to analyze and detect the content of news entered by users, are handled by the back-end system. This system also stores and manages the history of news checks. Meanwhile, the front-end system provides an intuitive and responsive user interface. In addition, the front-end displays views for hoax news verification, shows previous verification history, and allows users to view account profile details.

The application development uses the Multinomial Naive Bayes algorithm as the main algorithm. To implement this algorithm for Android, an Application Programming Interface (API) is used on the back-end system running on a web server. This API is built using Python with the Flask framework and Java with the SpringBoot framework. All database-related aspects, such as user data and hoax check results, are stored using PostgreSQL. The Multinomial Naive Bayes algorithm is integrated with the backend system through APIs implemented with Flask and SpringBoot. With this technology combination, the application can perform hoax message checking using the Multinomial Naive Bayes algorithm through APIs running on a web server. In this study, the Android-based mobile application is developed using the Kotlin programming language.

2.5. Application Design

1. Flowchart

The following is the main flowchart of the HoaxGY application with Android-based Multinomial Naive Bayes algorithms. When the HoaxGY application is launched, a splash screen appears. Before entering the login page, the application will checks the userID credentials stored on the Android device using the DataStore. If the user has never logged in / is using the application for the first time, you will be redirected to the Register page first. On the other hand, if the user has previously logged in to the application, the user will be taken directly to the Home page. After successfully entering the Home page the user has the option to go to the History page or the Profile page, depending on the user's needs. Figure 2 is the main flowchart explaining the HoaxGY application process in general

2. Entity Relationship Diagram (ERD)

The diagram below explains the relationships between entities in the application. The hoax news detection application requires two entities to be used, namely entity "user" and entity "news". The relationship concept

applied between the entities is a One To Many relationship, where the userId in the news entity is a foreign key of the ID user entity as the primary key. Figure 3 is the ERD of the HoaxGY application database.

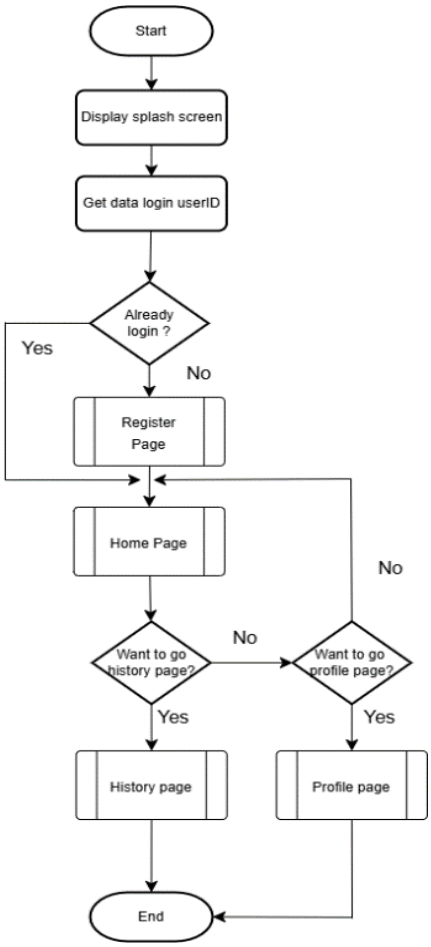


FIG 2. The main flowchart of HoaxGY application

3. User Interface

This section contains the results of the user interface design that will be implemented. When you open the application for the first time, it will display "Splash Screen". Figure 4a is the result of the splash screen interface design. Next is the interface design for the login page. There is a Hoaxgy application logo and is followed by a username text box and password to fill in the user account data. If the user does not have an account, there is a "Register" text button that will take the user to the register page. The Login button triggers the application's login function. Figure 4b shows the result of the login page interface design.

The interface design results on the Register page are almost identical to those on the Login page. It's just that there is an additional email text box and there is also a button available to trigger the register function. If the user already has an account, there is a "Login" text button that takes the user to the Login page. Figure 4c is the result of the Register page interface design

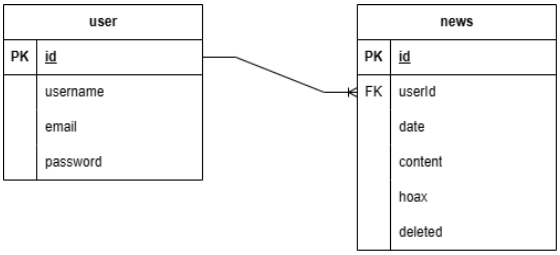
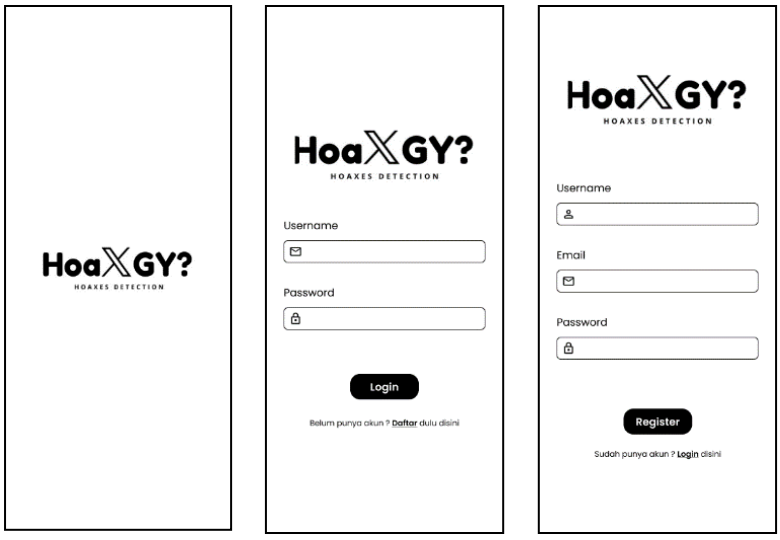


FIG 3. Entity Relationship Diagram Application application

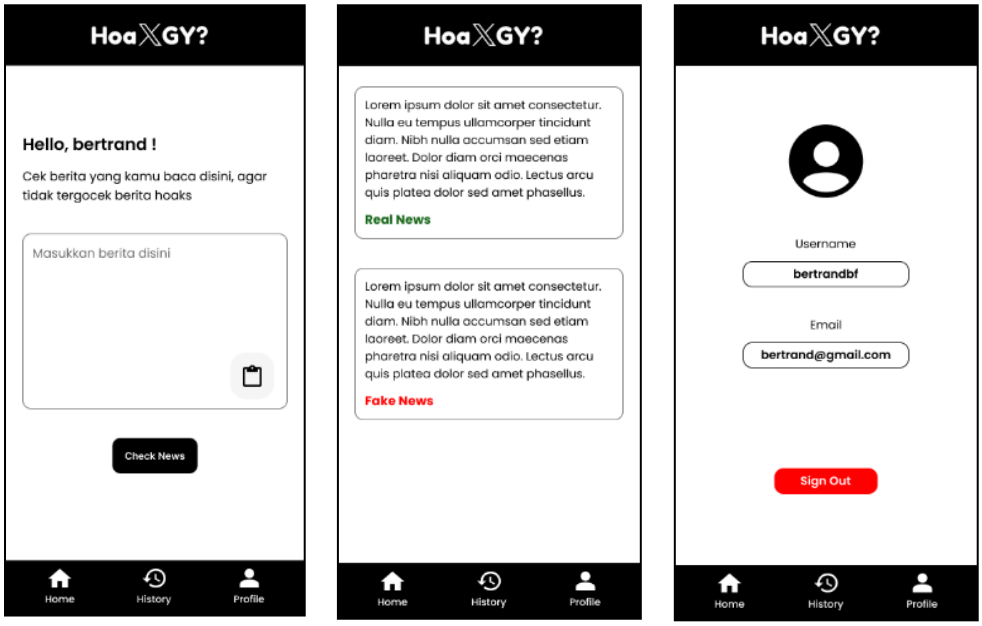


(a) (b) (c)

FIG 4. User Interface Initial Page design HoaxGY Application application

On the home page, there is a text box to fill in the title of the news will be detected. There is also a paste button to make it easier for users to paste the news title text. After entering the news title text, the user only needs to press the Check News button to perform the news detection function. Figure 5a shows the interface design of the home page.

The History page contains news titles that have been detected. This page displays the news title and the results of the hoax check. Figure 5b shows the interface design of the History page. The Profile page contains account information in the form of email and registered username. There is also a button for logging out of the account. Figure 5c shows the interface design of the Profile page.



(a) (b) (c)

FIG 5. User Interface Page Contents design HY HoaxGY Application application

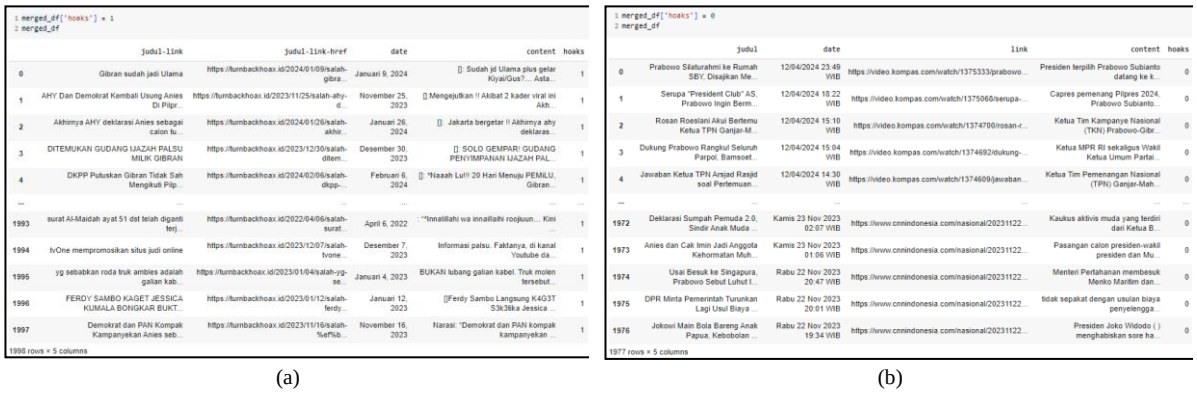
3. RESULTS AND DISCUSSION

3.1. Data Collection

In the data collection phase for developing the Multinomial Naive Bayes algorithm model in the Hoaxgy application, news articles will be scraped from Kompas, CNN, Detik as real news data and from TurnBackHoax as hoax news data. The scraping process for real news articles will use the BeautifulSoup library using Python. Meanwhile, the scraping of hoax news articles is facilitated by the Webscraper.io extension.

The collected news articles are exclusively about politics and span from November 2023 to April 2024. A total of 3,975 articles were collected and used as the dataset for building the algorithm model, consisting of 1,977 genuine news articles and 1,998 hoax news articles. The real news dataset includes 583 articles from Kompas, 674 from Detik, and 720 from CNN, each with 4 features: title, date, link, and content. Similarly, the hoax news dataset also contains 4 features: title, title-link-href, date, and content.

Next, in the data labeling phase, a column "hoax" was added, where the value 1 denotes hoax news and 0 denotes real news. After labeling, the number of features for each dataset was increased to 5 due to the addition of the hoax column. The labeled data results are shown in Figure 6a for fake news and Figure 6b for real news.



(a)

	judul	date	link	content	hoaks
0	Prabowo Silaturahmi ke Rumah SBY, Disajikan Me...	12/04/2024 23:49 WIB	https://video.kompas.com/watch/1375333/prabowo...	Presiden terpilih Prabowo Subianto datang ke k...	0
1	Serupa "President Club" AS, Prabowo Ingin Berm...	12/04/2024 18:22 WIB	https://video.kompas.com/watch/1375066/serupa-...	Capres pemenang Pilpres 2024, Prabowo Subianto...	0
2	Rosan Roeslani Akui Bertemu Ketua TPN Ganjar-M...	12/04/2024 15:10 WIB	https://video.kompas.com/watch/1374700/rosan-r...	Ketua Tim Kampanye Nasional (TKN) Prabowo-Gibr...	0
3	Dukung Prabowo Rangkul Seluruh Parpol, Bamsoet...	12/04/2024 15:04 WIB	https://video.kompas.com/watch/1374692/dukung-...	Ketua MPR RI sekaligus Wakil Ketua Umum Partai...	0
4	Jawaban Ketua TPN Arsjad Rasjid soal Pertemuan...	12/04/2024 14:30 WIB	https://video.kompas.com/watch/1374609/jawaban...	Ketua Tim Pemenangan Nasional (TPN) Ganjar-Mah...	0
...	...	...	...	...	...
1972	Deklarasi Sumpah Pemuda 2.0, Sindir Anak Muda ...	Kamis 23 Nov 2023 02:07 WIB	https://www.cnnindonesia.com/nasional/20231122...	Kaukus aktivis muda yang terdiri dari Ketua B...	0
1973	Anies dan Cak Imin Jadi Anggota Kehormatan Muh...	Kamis 23 Nov 2023 01:06 WIB	https://www.cnnindonesia.com/nasional/20231122...	Pasangan calon presiden-wakil presiden dan Mu...	0
1974	Usai Besuk ke Singapura, Prabowo Sebut Luhut I...	Rabu 22 Nov 2023 20:47 WIB	https://www.cnnindonesia.com/nasional/20231122...	Menteri Pertahanan membesuk Menko Maritim dan...	0
1975	DPR Minta Pemerintah Turunkan Lagi Usul Biaya ...	Rabu 22 Nov 2023 20:01 WIB	https://www.cnnindonesia.com/nasional/20231122...	tidak sepakat dengan usulan biaya penyelengga...	0
1976	Jokowi Main Bola Bareng Anak Papua, Keboboian ...	Rabu 22 Nov 2023 19:34 WIB	https://www.cnnindonesia.com/nasional/20231122...	Presiden Joko Widodo ( ) menghabiskan sore ha...	0

1977 rows × 5 columns

(b)

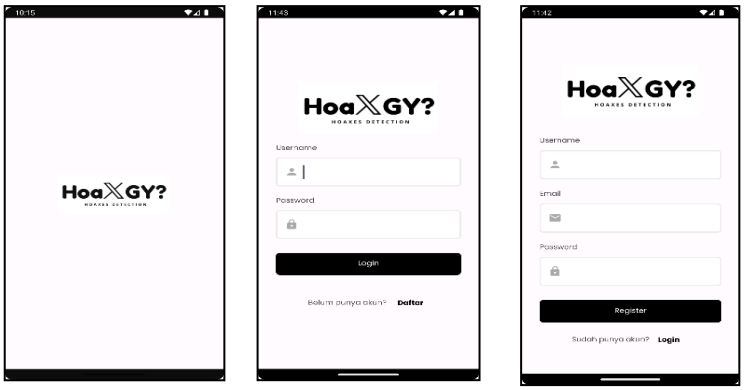
FIG 6. Labeling data fake and data real news

4.2. Implementation of the Hoax News Detection Application

1. Implementing the interface design

When you open the Hoaxgy application for the first time, it will immediately with a splash screen that will last for 3 seconds before finally switching to another page. The results of the splash screen implementation are shown in Figure 7a. When the splash screen is complete, it will redirect to the "Login Page". This page contains the Hoaxgy application logo as the application identity. There is also is also a form that users can fill out to access features in the application. If the user wants to log in, he/she can fill in the username and password form, after which he/she can immediately press the Login button. If the user doesn't have an account yet, there is a text button on this page that redirects to the Register page, which is useful for creating a new account. Figure 7b shows the results of the login page design implementation.

The "Register Page" screen has 3 input forms, starting with Username, Email and Password. This information is required to create a new account for the application. An account is created when the Register button is clicked. Implementation The Register Page is shown in Figure 7c.



(a) (b) (c)

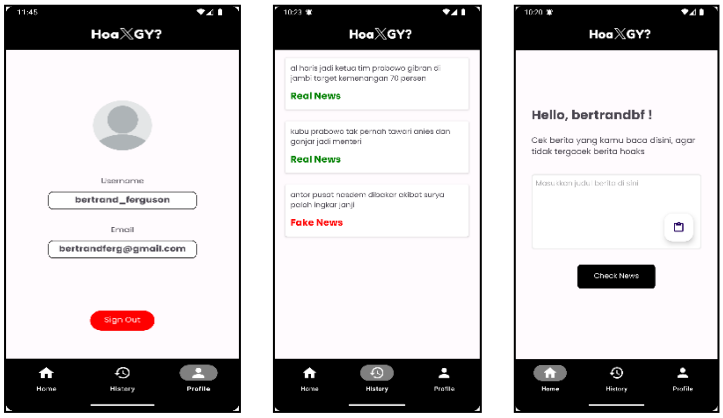
.

FIG 7. Implementaion Result of User Interface Initial Page design HoaxGY

The home page is the main feature of the Hoaxgy application, where users can check hoax news by entering the title of the news they want to detect in the text field provided. On the user side user experience, added a paste button to make things easier for users to paste the text you want to enter. After entering the news title to be detected, the user can immediately press the "Check News" button to run the Hoax News detection model is running. Each time you check the news, the detection result data will be displayed in the history page. Figure 8a shows the home page of Hoaxgy application.

On the History page, displays the results of checking news that has been done by the user based on the username of the user who has successfully logged into the application. The history feature was created so that users can double-check news titles that have already been detected, so the user doesn't have to re-check news that has already been detected. The page view history is shown in Figure 8b.

On the Profile page, users can see the username data along with the email registered on this account. There is also a sign out button if the user wants to leave the account or switch to another account. Figure 8c is a screenshot of the profile page of the Hoaxgy application.



(a) (b) (c)

FIG 8. Implementation Result of User Interface Page Contents design HoaxGY

2. Implementation of Multinomial Naive Bayes Algorithm Model

The first stage of building the algorithm model involves dropping unused columns and aligning column names between the real and hoax news datasets. This is because the model building process will use only 2 features (title and hoax) from the original dataset, which initially contained 5 features (title, link, date, content, hoax). After that, the data is combined using functions concat on a dataset of real news and hoax news. Next, do randomization data in the dataset using "sample(frac=1)" like shown in Figure 9.

To ensure that the data in the data set does not contain empty values (NaN), use the dropna function, then delete the symbol in the title data and change all data to lowercase. Finally, the data as shown in Figure 10 as a dataset for creating algorithm models.

```
1 dataset = pd.concat([real, fake]).sample(frac=1, random_state=42)
```

```
1 dataset
```

	judul	hoaks
378	Video Raffi Ahmad mendukung AMIN	1
495	Rekapitulasi Nasional: Saksi Partai Ummat Ungk...	0
1916	Video Atas Perintah Jokowi, Serangan TNI Bumi ...	1
1177	Demo Pemakzulan Jokowi diskualifikasi Prabowo ...	1
1178	Gibran Salim Mega, PAN: Semoga Pemilu 2024 Tak...	0
...	...	...
1130	Serba-serbi Hari Anak Sedunia 2023: Tema hingg...	0
1294	48 LSM Somasi Jokowi soal Kecurangan Pemilu, S...	0
860	Blak-blakan Perangkat Desa Sebut Ada Paslon Be...	0
1530	Mario Dandy dan Agnes Divonis Mati	1
1197	Erick Thohir Resmi Jadi Cawapres Prabowo	1

3975 rows × 2 columns

FIG 9. Dataset after concatting and merging data

```
1 dataset.dropna(subset=['judul'], inplace=True)
```

```
1 dataset['judul'] = dataset['judul'].apply(lambda x: re.sub('\W+', '
```

```
1 dataset
```

	judul	hoaks
378	video raffi ahmad mendukung amin	1
495	rekapitulasi nasional saksi partai ummat ungka...	0
1916	video atas perintah jokowi serangan tni bumi h...	1
1177	demo pemakzulan jokowi diskualifikasi prabowo ...	1
1178	gibran salim mega pan semoga pemilu 2024 tak t...	0
...	...	...
1130	serba serbi hari anak sedunia 2023 tema hingga...	0
1294	48 lsm somasi jokowi soal kecurangan pemilu si...	0
860	blak blakan perangkat desa sebut ada paslon be...	0
1530	mario dandy dan agnes divonis mati	1
1197	erick thohir resmi jadi cawapres prabowo	1

3975 rows × 2 columns

FIG 10. Result Dataset Pre-processing

Before building the algorithm model, the data was divided into two, namely training data and test data, using the attributes "title" and "hoax". The data is divided into training data (X train, y train) and test data (X test, y test). Distribution training and test data using the scikit-learn library, namely train test split with a test size of 0.2 or 20% of the total data (3,975), so the number of test data size is 795, while the training size is 1-0.2 = 0.8 (80%) or a total of 3,180 data. The code snippet for splitting data can be written as follow:

Splitting data into training and test data

```
# Train-test split
x_train, x_test, y_train, y_test = train_test_split(dataset['judul'], dataset['hoaks'], test_size=0.2,
random_state=42)
```

Next, a pipeline for text processing and building a classification model is created. The pipeline consists of 3 steps, namely CountVectorizer to convert text into a frequency vector, TfidTransformer to convert a frequency vector into a TF-IDF vector, and MultinomialNB as a Naive Bayes classification model. The code snippet to create a Naive Bayes model can be written as follow:

Pipeline naive bayes model

```
Text_clf = Pipeline([
    ('vect', countvectorizer()),
    ('tfidf, tfidfTransformer()),
    ('clf', multinomialNB()),
])
```

In the model training phase, the classification model learns patterns from the textual headlines in the training data and appropriate labels, so that it can make accurate predictions on test data. The model training process is performed by the "fit()" method in "text clf" as shown in Figure 11.

The final stage, by creating a function for testing the model with the "test model()" function, which can take news headlines as input and predict as input, predicts whether the news is real or fake, and returns the and return the predicted value. This function was created to be used on the Hoaxgy application on the Android platform. The code snippet of the model testing function can be written as follow:

Model testing function

```
Def test_model(text):
   testing_text = [text]
   prediction_proba = text_clf.preditc_proba(testing_text)
   #mengambil probabilitas prediksi kelas positif (fake news)
   prob_fake_news = prediction_proba[0][1]

   #Mengkonversi probabilitas menjadi string dengan 3 angka di belakang koma
   prob_str = "{:.3f}".format{prob_fake_news}

   #Mengkonversi nilai probabilitas menjadi float
   prob_float = float(prob_str)

   if prob_float > 0.47:
      news = "Fake News"
   else:
      news = "Real News"

   return news, prob_float
```

3. Algorithm Model Evaluation

The Multinomial Naive Bayes algorithm model was evaluated using test data with Confusion Matrix. Confusion Matrix is a table used to evaluate the performance of the classification model by comparing the model prediction results with the actual values from the test data. The code snippet for model evaluation can be written as follows:

Confusion Matrix Code Snippet

```
predicted = text_clf.predict(x_test)
conf_matrix = metrics.confusion_matrix(y_test, predicted)

print("Confusion Matrix: ")
print(conf_matrix)
```

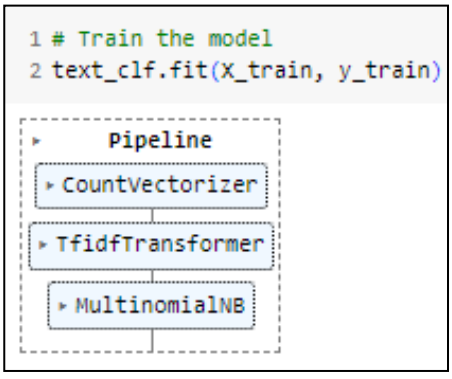


FIG 11. Training model naive bayes

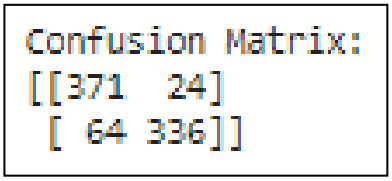


FIG 12. Confusion Matrix Results

Based on the results of the confusion matrix code snippet shown in Figure 12, the confusion matrix results are obtained as shown in Figure 12, from these results the confusion matrix can be concluded that:

- A total of 336 data are (TP) True Positive, which is actually a hoax and correctly predicted as hoax news by the model
- A total of 371 data are (TN) True Negative, which are actually news and correctly predicted by the model as real news.
- A total of 64 data are (FP) False Positive, which are actually hoax news but predicted by the model to be real news.

A total of 24 data are (FN) False Negative, which are actually real news but predicted by the model as hoax news. After evaluating the model with the Confusion Matrix, this research also calculates the model accuracy by determining the values of Accuracy, Precision, Recall and F1-Score. On the calculation manually obtained the following values:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{336+371}{336+371+24+64} = 0{,}889 \quad (1)$$

$$Precision = \frac{TP}{TP+FP} = \frac{336}{336+24} = \frac{338}{361} = 0{,}933 \quad (2)$$

$$Recall = \frac{TP}{TP+FN} = \frac{336}{336+64} = 0{,}84 \quad (3)$$

$$F1 = \frac{2\times(Precision\times Recall)}{Precision+Recall} = \frac{2\times 0{,}933\times 0{,}84}{0{,}933+0{,}84} = 0{,}884 \quad (4)$$

Then, the accuracy, precision, recall and F1 scores were calculated in Python using the "metrics" module from scikit-learn, which produces values, the code snippet can be written as follow:

```
Splitting data into training and test data
accuracy = metrics.accuracy_score(y_test, predicted)
precision = metrics.precision_score(y_test, predicted)
recall = metrics.recall_score(y_test, predicted)
f1_score = metrics.f1_score(y_test, predicted)

print("Accuracy:", accuracy)
print("Precision:", precision)
print("Recall:", recall)
print("F1 Score:", f1_score)

output:
Accuracy: 0.889308176100629
Precision: 0.933333333333333
Recall: 0.84
F1 Score: 0.8842105263157894
```

The results of calculating the Accuracy, Precision, Recall and F1 Score values, whether done manually or using Python, also produce the same values as in Table 2.

TABLE 1. Calculation of Accuracy, Precision, Recall and F1-score values

	Manual	**Python**
Accuracy	0,889	0.889308176100629
Precision	0,933	0.9333333333333333
Recall	0,84	0,84
F1-score	0,884	0.8842105263157894

4. CONCLUSIONS

Conclusions that can be drawn from the research that has been completed are as follows; first. hoax news detection application designed and built using the Android-based Multinomial Naive Bayes algorithm successfully created. An application called Hoaxgy which is built using Kotlin programming language created using the Android Studio IDE. This application has the ability to detect hoax news using news headline text. And in the testing process using a confusion matrix, a detection model hoax news with use news headline text algorithm achieves a level of accuracy of 88.9%. In this case, it means that the algorithm model can correctly predict 88.9% of all classifications. Besides Therefore, a precision value of 93.33% shows all the predicted news as hoax news / fake news by the model, 93.33% of the predictions is really a hoax. Test results also show that the application has a recall rate of 84%, which means the model was successful in detecting it 84% of all existing hoax news. The last indicator, f1 which is a combination of precision and recall, shows a value of 88.4%. By looking at these values, it can be concluded that the hoax news detection model using the Multinomial Naive Bayes algorithm that have been created have quite good performance in detecting hoax news, with high accuracy, solid precision and good recall is quite good and has the potential to contribute to the environment online by the Indonesian people by verifying the information first before sharing it on social media

ACKNOWLEDGMENT

Thank you to Universitas Multimedia Nusantara in Indonesia for providing a space for academics to conduct this journal research. Hopefully, this research will contribute significantly to the growth of technology in Indonesia

REFERENSI

[1] R. Agusiady, D. Saepudin, and Z. Aripin, "The influence of social media communication on consumer perceptions of brands and purchase intentions in the pandemic and post-pandemic era: an analytical study," *J. Jabar Econ. Soc. Netw. Forum*, vol. 2, no. 1, pp. 16–30, 2024.

[2] R. Huerta-Álvarez, J. J. Cambra-Fierro, and M. Fuentes-Blasco, "The interplay between social media communication, brand equity and brand engagement in tourist destinations: An analysis in an emerging economy," *J. Destin. Mark. Manag.*, vol. 16, pp. 1–34, 2020, doi: 10.1016/j.jdmm.2020.100413.

[3] E. Aïmeur, S. Amri, and G. Brassard, *Fake news, disinformation and misinformation in social media: a review*, vol. 13, no. 1. Springer Vienna, 2023. doi: 10.1007/s13278-023-01028-5.

[4] Y. M. Rocha, G. A. de Moura, G. A. Desidério, F. D. de Oliveira, C. H., Lourenço, and L. D. de Figueiredo Nicolete, "The impact of fake news on social media and its influence on health during the COVID-19 pandemic: A systematic review," *J. Public Heal. From Theory to Pract.*, pp. 1007–1016, 2021, [Online]. Available: https://doi.org/10.1007/s10389-021-01658-z

[5] M. L. Enabler, "Hasil Survey Wabah HOAX Nasional 2019," mastel.id. Accessed: May 20, 2024. [Online]. Available: https://mastel.id/hasil-survey-wabah-hoax-nasional-2019/

[6] B. H. K. Kominfo, "Sampai Mei 2023, Kominfo Identifikasi 11.642 Konten Hoaks," kominfo.go.id. Accessed: May 21, 2024. [Online]. Available: https://www.kominfo.go.id/content/detail/49914/siaran-pers-no123hmkominfo062023-tentang-sampai-mei-2023-kominfo-identifikasi-11642-konten-hoaks/0/siaran_pers

[7] Bskdn.kemendagri.go.id, "Riset: 44 Persen Orang Indonesia Belum Bisa Mendeteksi Berita Hoax," bskdn.kemendagri.go.id. Accessed: May 21, 2024. [Online]. Available: https://bskdn.kemendagri.go.id/website/riset-44-persen-orang-indonesia-belum-bisa-mendeteksi-berita-hoax/

[8] D. N. Irwanto, "Identifikasi Berita Hoax Menggunakan Kombinasi Metode K-Nearest Neigbor (KNN) dan TF-IDF Berbasis Web Dengan Menggunakan Framework Codeigniter," UNIVERSITAS WIJAYA PUTRA, 2021.

[9] M. Audina, A. E. Karyawati, I. W. Supriana, I. K. G. Suhartana, I. G. S. Astawa, and I. W. Santiyasa, "Klasifikasi Berita Hoaks Covid-19 Menggunakan Kombinasi Metode K-Nearest Neighbor dan Information Gain," *J. Elektron. Ilmu Komput. Udayana*, vol. 10, no. 4, p. 319, 2022, doi: 10.24843/jlk.2022.v10.i04.p02.

[10] T. Ige and S. Adewale, "AI Powered Anti-Cyber Bullying System using Machine Learning Algorithm of Multinomial Naïve Bayes and Optimized Linear Support Vector Machine Interception of Cyberbully Contents in a Messaging System by Machine Learning Algorithm," *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 5, pp. 5–9, 2022, doi: 10.14569/IJACSA.2022.0130502.

[11] Yuyun, Nurul Hidayah, and Supriadi Sahibu, "Algoritma Multinomial Naïve Bayes Untuk Klasifikasi Sentimen Pemerintah Terhadap Penanganan Covid-19 Menggunakan Data Twitter," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 4, pp. 820–826, 2021, doi: 10.29207/resti.v5i4.3146.

[12] W. B. Zulfikar, A. R. Atmadja, and S. F. Pratama, "Sentiment Analysis on Social Media Against Public Policy Using Multinomial Naive Bayes," *Sci. J. Informatics*, vol. 10, no. 1, pp. 25–34, 2023, doi: 10.15294/sji.v10i1.39952.

[13] C. S. Sriyano and E. B. Setiawan, "Pendeteksian Berita Hoax Menggunakan Naive Bayes Multinomial Pada Twitter dengan Fitur Pembobotan TF-IDF," *Repos. Telkom Univ.*, vol. 8, no. 2, p. 3396, 2021.

[14] A. Rusli, A. Suryadibrata, S. B. Nusantara, and J. C. Young, "A Comparison of Traditional Machine Learning Approaches for Supervised Feedback Classification in Bahasa Indonesia," *IJNMT (International J. New Media Technol.*, vol. 7, no. 1, pp. 28–32, 2020, doi: 10.31937/ijnmt.v1i1.1485.

[15] S. Nurhidayah, "A Multinomial Naïve Bayes Decision Support System For Covid-19 Detection," *FUDMA J. Sci.*, vol. 5, no. 1, p. 55, 2020, doi: 10.33003/fjs-2020-0402-331.

[16] N. Y. Hutama, K. M. Lhaksmana, and I. Kurniawan, "Text Analysis of Applicants for Personality Classification Using Multinomial Naïve Bayes and Decision Tree," *J. Infotel*, vol. 12, no. 3, pp. 72–81, 2020, doi: 10.20895/infotel.v12i3.505.

[17] N. Giandomenico , Di Domenicoa; Jason. Sita; Alessio, Ishizakab; Daniel, "Fake News, Social Media and Marketing: A Systematic Review," *J. Bus. Res.*, pp. 0–48, 2020, doi: https://doi.org/10.1016/j.jbusres.2020.11.037.

[18] M. Zhikri and W. Istiono, "Handling Class Imbalance for Indonesian Twitter Sentiment Analysis A Comparative Study of Algorithms," *J. Syst. Manag. Sci.*, vol. 14, no. 10, pp. 170–179, 2024, doi: 10.33168/JSMS.2024.1010.

[19] Sajid Khan, Mehmoon Anwar, Huma Qayyum, Farooq Ali, and Marriam Nawaz, "Fake News Classification using Machine Learning: Count Vectorizer and Support Vector Machine," *J. Comput. Biomed. Informatics*, vol. 4, no. 01, 2023, doi: 10.56979/401/2022/85.

[20] K. M. Suryaningrum, "Comparison of the TF-IDF Method with the Count Vectorizer to Classify Hate Speech," *Eng. Math. Comput. Sci. J.*, vol. 5, no. 2, pp. 79–83, 2023, doi: 10.21512/emacsjournal.v5i2.9978.

[21] A. R. Lubis, M. K. M. Nasution, O. S. Sitompul, and E. M. Zamzami, "The effect of the TF-IDF algorithm in times series in forecasting word on social media," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 22, no. 2, pp. 976–984, 2021, doi: 10.11591/ijeecs.v22.i2.pp976-984.

[22] L. Xiang, "TFIDF-Application of an Improved TF-IDF Method in Literary Text Classification.pdf," *Hindawi*, 2022, doi: https://doi.org/10.1155/2022/9285324.

[23] 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," *Int. J. Inf. Technol.*, vol. 14, no. 7, pp. 3629–3635, 2022, doi: 10.1007/s41870-022-01096-4.

[24] H. D. Abubakar and M. Umar, "Sentiment Classification: Review of Text Vectorization Methods: Bag of Words, Tf-Idf, Word2vec and Doc2vec," *SLU J. Sci. Technol.*, vol. 4, no. 1&2, pp. 27–33, 2022, doi: 10.56471/slujst.v4i.266.

[25] J. Xu, Y. Zhang, and D. Miao, "Three-way confusion matrix for classification: A measure driven view," *Inf. Sci. (Ny).*, vol. 507, pp. 772–794, 2020, doi: 10.1016/j.ins.2019.06.064.

[26] D. Krstinić, M. Braović, L. Šerić, and D. Božić-Štulić, "Multi-label Classifier Performance Evaluation with Confusion Matrix," pp. 01–14, 2020, doi: 10.5121/csit.2020.100801