SMATIKA : STIKI Informatika Jurnal
Accredited Sinta 4 Ministry of Higher Education, Science and Technology Republic of Indonesia SK No. 79/E/KPT/2023
E-ISSN: 2580-6939 || P-ISSN: 2087-0256

Implementation of Naive Bayes Method and Natural Language Processing in Web-Based Online Hoax News Detection System

Mohammad Ainur Rofiqi[1*], Ahmad Heru Mujianto[2]

[1] *Hasyim Asy'ari University, Faculty of Information Technology, Informatics Engineering, Jl. Irian Jaya No. 55, Tebuireng, Cukir, Diwek District, Jombang Regency, East Java 61471, Indonesia*
[2] *Hasyim Asy'ari University, Faculty of Information Technology, Information Systems, Jl. Irian Jaya No. 55, Tebuireng, Cukir, Diwek District, Jombang Regency, East Java 61471, Indonesia*

Keyword

Hoax Detection; Naive Bayes; Natural Language Processing; TF-IDF; Web-Based System

****Corresponding Author:***
ainurrofiqi@mhs.unhasy.ac.id

Abstract

The rapid growth of digital media in Indonesia has accelerated the spread of hoax news, which poses serious threats to public trust and social stability. Based on data from the Ministry of Communication and Informatics of the Republic of Indonesia, a total of 12,547 hoax contents were identified from 2018 to 2023, while 1,923 new hoax contents emerged in 2024 alone. This research aims to design and build a web-based hoax news detection system by implementing the Multinomial Naive Bayes algorithm combined with Natural Language Processing (NLP) techniques. The system processes text through five NLP stages: sentence splitting, case folding, tokenizing, stopword removal, and stemming using PySastrawi. Feature weighting is performed using TF-IDF (Term Frequency–Inverse Document Frequency), and classification is executed using the Multinomial Naive Bayes algorithm enhanced with Laplace Smoothing and Log Posterior Probability. The output is converted using the Softmax function to produce probability percentages for each sentence. A manual calculation simulation using 10 training sentences and one test narrative containing four sentences was conducted to verify the algorithm. The system successfully classified the test narrative as hoax with a probability of 62.00% hoax and 38.00% non-hoax, consistent with the actual label. The system was evaluated using a Confusion Matrix on a 50-sentence test dataset, achieving Accuracy of 90.00%, Precision of 91.67%, Recall of 88.00%, and F1-Score of 89.80%. The resulting system provides a practical tool for the public to verify the credibility of news information in the digital era.

1. Introduction

Advances in digital technology alongside the expansion of internet connectivity have substantially altered the way individuals access and receive information. Data from the Indonesian Internet Service Providers Association reveals that roughly 221 million Indonesians, constituting over 79% of the national population, are currently online [1]. Although this expanded connectivity enables broad and open access to knowledge, it has equally become a channel through which hoax news spreads rapidly and without restraint.

Hoax news, commonly referred to as fake news, is characterized as fabricated or deceptive content intentionally designed to mimic genuine news reporting. According to official records from the Ministry of Communication and Informatics of the Republic of Indonesia, as many as 12,547 hoax items were found to be

circulating on digital platforms from 2018 through the end of 2023, spanning subject areas such as health, politics, and social matters [2]. Adding to this concern, the Ministry of Communication and Digital Affairs documented 1,923 additional hoax items in 2024 alone, reflecting an ongoing upward trend [3]. Research has shown that false news spreads significantly faster and more broadly than true news on social media platforms, reaching more people more quickly across all categories of information [4]. The social impact of hoax content is substantial: studies have documented that fake news can measurably shift public opinion and voting behavior, underlining the urgency of automated detection tools [5]. Studies indicate that the swift propagation of breaking news via social media has considerable effects on how the public perceives and trusts information [6]. Further research on the language of viral misinformation shows that hoax content tends to follow identifiable linguistic characteristics that allow systematic detection [7].

A number of studies have explored the use of machine learning techniques in identifying hoax content. Among these, comparisons between Naive Bayes Classifier and Support Vector Machine (SVM) for categorizing hoax news in Indonesian online media have shown that both approaches yielded competitive results [8]. Separately, the use of Naive Bayes and SVM for Indonesian-language hoax detection also highlighted how significantly text preprocessing quality influences model accuracy [9]. When multiple algorithms, including Multinomial Naive Bayes, Logistic Regression, and Passive Aggressive classifiers, were evaluated, the results indicated that while Naive Bayes remained competitive, the relative performance of each method was substantially shaped by the nature of the dataset used [10]. A critical distinction between these prior works and the present study lies in their scope and output granularity: existing systems predominantly deliver a single document-level classification label, offering no insight into which specific sentences within a news narrative carry hoax signals. Furthermore, none of the reviewed systems combined sentence-level splitting with Softmax probability interpretation to provide users with per-sentence confidence scores, a design feature that substantially enhances transparency and practical usability for non-expert end users.

Most prior works focused narrowly on algorithm evaluation with little effort to build usable tools for the public. The present study addresses this gap by constructing a fully operational web-based hoax detection system. Its novelty is threefold:

- sentence-level splitting enables fine-grained detection beyond document-level classification
- Softmax conversion of log posterior probabilities produces per-sentence confidence scores absent from prior systems
- the system is deployed as a publicly accessible web application.

The developed system combines Multinomial Naive Bayes with NLP-based text preprocessing and TF-IDF weighting to perform sentence-level classification of user-submitted news narratives. By applying sentence splitting, each submitted narrative is broken down into individual sentences that are then processed separately through the NLP pipeline. The Softmax function subsequently transforms the resulting log posterior values into probability outputs that can be easily interpreted. This design gives users granular, sentence-by-sentence insight into hoax probability within any given news narrative.

2. Research Method

The present research adopts an applied research framework centered on system development, covering stages of literature review, data gathering, system design, coding, and performance evaluation. The technical stack comprises PHP for the web-based front-end interface, Python utilizing the Flask framework as the back-end engine for NLP and machine learning tasks, and MySQL for data storage and management. The development environment ran on XAMPP under macOS, with the Python setup incorporating PySastrawi to support Indonesian language stemming and scikit-learn to handle TF-IDF vectorization and Naive Bayes model training. The development workflow follows an organized sequence, detailed in the subsections that follow.

2.1 Data Collection

The training corpus was drawn from two distinct sources. Non-hoax sentences were taken from detik.com, a widely trusted and high-traffic Indonesian news outlet. Hoax sentences were obtained from turnbackhoax.id, an established platform dedicated to debunking misinformation, operated by MAFINDO (Masyarakat Anti Fitnah Indonesia). Data collection was carried out between January and March 2026. Sentence selection followed four explicit criteria:

- the sentence must be a complete, well-formed Indonesian-language clause
- the sentence must contain a minimum of ten words to provide sufficient linguistic context for classification
- no duplicate sentences were permitted across the dataset
- sentences were drawn from content covering health, politics, and social topics to ensure topical diversity

The labeling procedure assigned the label 'hoaks' to any sentence sourced from content verified as misinformation by turnbackhoax.id, and the label 'non-hoaks' to any sentence sourced from verified news articles on detik.com, ensuring label validity and consistency throughout the dataset. Every sentence in the dataset is an individual Indonesian-language entry, tagged manually as either 'hoaks' (hoax) or 'non-hoaks' (non-hoax). The overall dataset comprised 250 annotated sentences, divided using an 80:20 ratio, 200 for training and 50 for testing. It should be noted that the evaluation relies on a single 80:20 train-test split, which is a limitation of the present study; this partitioning strategy may yield performance estimates that vary across different data splits. Future work is encouraged to adopt k-fold cross-validation to obtain more stable and reliable performance estimates. For the purpose of algorithm verification via manual calculation, a smaller subset of 10 sentences (5 Non-Hoax and 5 Hoax) was drawn from the training pool purely for demonstration purposes; this subset is used solely to illustrate the step-by-step Naive Bayes computation and does not represent the actual training corpus of 200 sentences used by the model. These sentences were selected to represent a diversity of topics and linguistic styles within each category, enabling a step-by-step walkthrough of the Naive Bayes classification logic. User-submitted news narratives or paragraphs entered via the web interface constituted the test data. It should be acknowledged as a limitation of this study that the total dataset size of 250 sentences is relatively modest for robust machine learning evaluation; a larger and more linguistically diverse corpus would be expected to improve model generalization and reduce the risk of overfitting to domain-specific vocabulary.

2.2 NLP Preprocessing

All textual input, whether training sentences or user-submitted test narratives, was subjected to a unified NLP preprocessing workflow consisting of five ordered stages, each described below :

- Sentence Splitting: Applicable only to test narratives submitted as paragraphs, this step breaks the input into individual sentences by recognizing end-of-sentence punctuation such as periods, question marks, and exclamation marks. Each extracted sentence then proceeds through the remaining pipeline stages independently
- Case Folding: All characters in the text are converted to lowercase to ensure uniform treatment of words regardless of their original capitalization
- Tokenizing: The lowercased text is segmented into discrete word tokens for use in subsequent processing steps
- Stopword Removal: Frequently occurring Indonesian function words (e.g., 'dan', 'yang', 'di') that contribute minimal meaning to the classification task are filtered out from the token list.
- Stemming: Using the PySastrawi library, each token is reduced to its base form (e.g., 'mendaftarkan' becomes 'daftar'), thereby normalizing morphological variants into a single consistent representation.

2.3 TF-IDF Feature Weighting

Once preprocessing is complete, the resulting token sets are converted into numerical feature vectors through TF-IDF (Term Frequency–Inverse Document Frequency). TF-IDF quantifies how significant a particular word is within a given document in relation to the broader corpus [11]. The relevant formulas are presented below.

Term Frequency (TF) for term t in document d:

$$TF(t,d) = \frac{f(t,d)}{\Sigma f(k,d)} \quad (1)$$

where f(t,d) is the frequency of term t in document d, and Σf(k,d) is the total number of terms in document d.

Inverse Document Frequency (IDF) for term t:

$$IDF(t) = log(\frac{N}{df(t)}) \quad (2)$$

where N is the total number of documents and df(t) is the number of documents containing term t.

The final TF-IDF weight is:

$$TF - IDF(t,d) = TF(t,d) \times IDF(t) \quad (3)$$

When a term's TF-IDF score is high, it means the term is prominent in that particular document but seldom found across the rest of the corpus, which makes it a useful indicator for distinguishing between document classes [12].

2.4 Multinomial Naive Bayes Classification

Text classification in this system is performed through the Multinomial Naive Bayes algorithm, which operates on the basis of Bayes' Theorem under the assumption that features are conditionally independent of one another. The algorithm computes a posterior probability for every possible class and assigns the class with the maximum probability as the output label [13]. Bayes' Theorem is expressed as:

$$P(C|X) = P(X|C) \times \frac{P(C)}{P(X)} \quad (4)$$

where P(C|X) is the posterior probability, P(X|C) is the likelihood, P(C) is the prior probability, and P(X) is the evidence.

2.5 Laplace Smoothing

When a term in the test data has no occurrence in the training set, a zero-probability situation arises that can disrupt the classification computation. To overcome this, Laplace Smoothing is employed, which adds a value of 1 to every term's frequency count [13]. The resulting smoothed likelihood formula is:

$$P(t|C) = \frac{(f(t,C) + 1)}{(\Sigma f(k,C) + |V|)} \quad (5)$$

where f(t,C) is the frequency of term t in class C, Σf(k,C) is the total words in class C, and |V| is the total unique vocabulary across all training data.

2.6 Log Posterior Probability

Directly multiplying many small probability values can produce numerical underflow, making the results unreliable. To mitigate this, all Naive Bayes probability computations are carried out in the logarithmic domain. The Log Posterior Probability formula is:

$$log\ P(C|D)\ =\ log\ P(C)\ +\ \Sigma\ f(t,d)\ \times\ log\ P(t|C) \qquad (6)$$

where f(t,d) is the frequency of term t in test document D. The class whose log posterior value is the least negative is selected as the final classification output.

2.7 Softmax Conversion

Since log posterior values are inherently negative and cannot be directly read as percentage probabilities, the Softmax function is applied to transform them into normalized probability scores for each class. The formula is:

$$P(C) = \frac{\exp(\log P(C|D))}{\Sigma i \exp(\log P(Ci|D))} \qquad (7)$$

The output of this transformation is a set of probability values across all classes that collectively total 100%, offering users a readable confidence score for each sentence's classification result.

2.8 System Architecture

The overall system is built on a dual-component design: a PHP-driven web interface that handles user interaction and database operations, paired with a Python-Flask backend API responsible for NLP processing and classification tasks. Upon receiving a news narrative from a user, the PHP front-end transmits the input to the Flask API, which then performs text analysis, runs the classification model, and sends the resulting probability scores back to the web interface for display.

Figure 1 illustrates the complete system architecture, showing how the PHP front-end, Flask API, and MySQL database interact to process user input and return classification results.

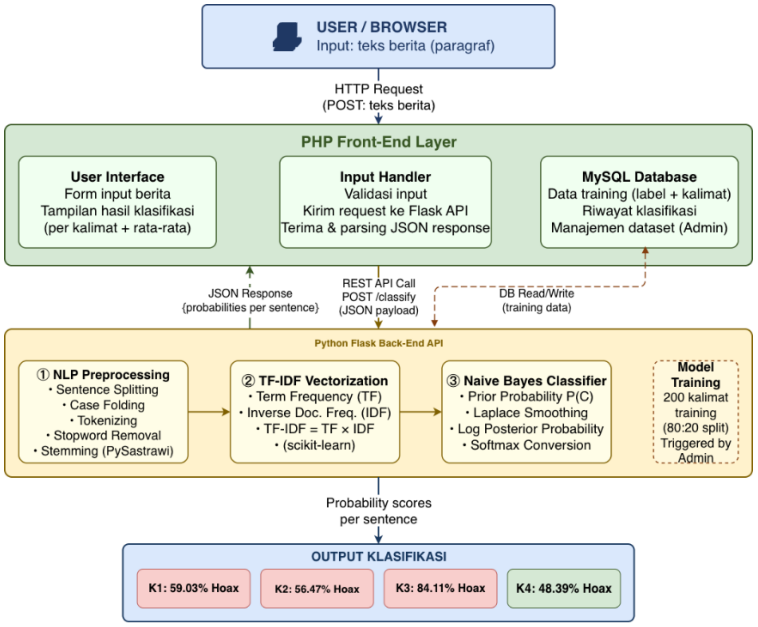

Figure 1. System Architecture: PHP Front-End ↔ Flask API ↔ MySQL Database

2.9 Use Case Diagram

The system defines two main actors: Admin and User, as shown in Figure 2. The Admin manages backend operations including dataset control, model training, and classification history. The User interacts solely with the detection interface via a standard web browser without authentication. This role separation protects core system functions while keeping the detection feature publicly accessible.

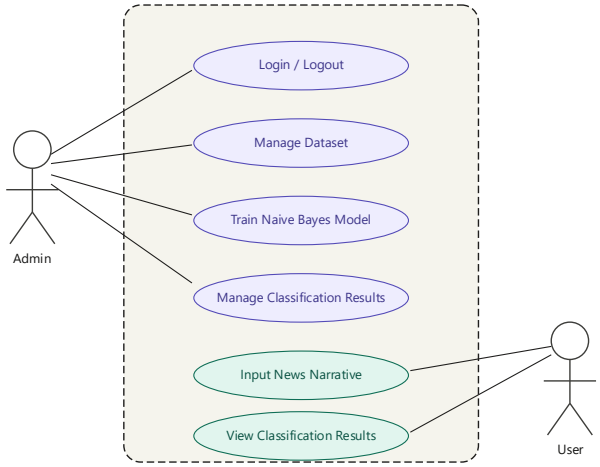

Figure 2. Use Case Diagram of Hoax News Detection System

3. Result and Discussions

3.1 Training Data

For the manual calculation simulation, 10 annotated training sentences were used: 5 classified as Non-Hoax (drawn from detik.com) and 5 classified as Hoax (drawn from turnbackhoax.id), as listed in Table 1.

Table 1. Training Dataset (10 Labeled Sentences)

No	Label	Training Sentence
1	Non-Hoax	BMKG mengingatkan masyarakat untuk waspada terhadap potensi cuaca ekstrem berupa hujan lebat disertai petir dan angin kencang yang berpotensi menyebabkan bencana banjir dan tanah longsor.
2	Non-Hoax	BMKG menyatakan bahwa cuaca ekstrem berpotensi meningkatkan risiko bencana banjir dan tanah longsor, serta mengimbau masyarakat untuk waspada dan mengikuti informasi resmi.
3	Non-Hoax	Pemerintah melalui Kementerian Perhubungan menyiapkan fasilitas transportasi untuk menghadapi lonjakan penumpang dan mengimbau masyarakat agar merencanakan perjalanan dengan baik serta mematuhi aturan keselamatan.
4	Non-Hoax	Kementerian Pendidikan mendorong pemanfaatan teknologi digital dalam pembelajaran untuk memudahkan akses materi dan meningkatkan kualitas pendidikan.
5	Non-Hoax	Pemerintah daerah bekerja sama dengan berbagai instansi untuk meningkatkan layanan publik di bidang kesehatan dan pendidikan, serta membangun fasilitas dan program edukasi bagi masyarakat.
6	Hoax	Beredar informasi di media sosial yang menyebutkan bahwa Bank Indonesia akan menarik seluruh uang rupiah dan menggantinya dengan mata uang baru, serta meminta masyarakat menukarkan uang lama sebelum batas waktu yang ditentukan.
7	Hoax	Sebuah pesan berantai menyebutkan bahwa memakai masker dalam waktu lama dapat menyebabkan keracunan karbon dioksida yang berbahaya bagi kesehatan manusia.
8	Hoax	Beredar informasi yang menyebutkan bahwa seluruh ponsel akan disadap oleh pemerintah melalui jaringan 5G tanpa sepengetahuan dan tanpa kegunaan yang jelas bagi pengguna.

No	Label	Training Sentence
9	Hoax	Beredar kabar yang menyebutkan bahwa vaksin COVID-19 mengandung microchip untuk mengendalikan manusia.
10	Hoax	Pesan viral mengklaim bahwa meminum air garam hangat setiap hari dapat menyembuhkan semua penyakit tanpa perlu obat maupun penanganan medis.

3.2 NLP Preprocessing Results

All training sentences and the test narrative were individually run through the NLP preprocessing pipeline. Table 2 shows the tokenization output for each of the 10 training sentences after preprocessing.

Table 2. Preprocessing Results for Training Data

No	Label	Tokens After Preprocessing
D1	Non-Hoax	bmkg | ingat | masyarakat | waspada | potensi | cuaca | ekstrem | hujan | lebat | petir | angin | kencang | sebab | bencana | banjir | tanah | longsor
D2	Non-Hoax	bmkg | nyata | cuaca | ekstrem | potensi | tingkat | risiko | bencana | banjir | tanah | longsor | imbau | masyarakat | waspada | ikut | informasi | resmi
D3	Non-Hoax	perintah | kementerian | hubung | siap | fasilitas | transportasi | hadap | lonjak | tumpang | imbau | masyarakat | rencana | jalan | baik | patuh | atur | selamat
D4	Non-Hoax	kementerian | didik | dorong | manfaat | teknologi | digital | ajar | mudah | akses | materi | tingkat | kualitas | didik
D5	Non-Hoax	perintah | daerah | kerja | instansi | tingkat | layan | publik | sehat | didik | bangun | fasilitas | program | edukasi | masyarakat
D6	Hoax	edar | informasi | media | sosial | sebut | bank | indonesia | tarik | seluruh | uang | rupiah | ganti | mata | uang | baru | masyarakat | tukar | uang | lama | batas | waktu
D7	Hoax	pesan | sebut | masker | lama | sebab | racun | karbon | dioksida | bahaya | sehat | manusia
D8	Hoax	edar | informasi | sebut | ponsel | sadap | pemerintah | jaringan | tanpa | tahu | guna
D9	Hoax	kabar | sebut | vaksin | covid | kandung | microchip | kontrol | manusia
D10	Hoax	pesan | klaim | minum | air | garam | hangat | sembuh | semua | sakit | tanpa | obat | medis

Based on Table 2, the resulting corpus statistics are as follows: the Non-Hoax class contains 84 total words; the Hoax class contains 62 total words; and the combined unique vocabulary |V| spans 108 distinct terms.

3.3 Test Data Preprocessing

The test narrative applied in this simulation was a confirmed hoax circulating about a government social assistance program, presented as follows:

"Yang belum daftar buruan daftar sekarang. Ternyata NIK KTP berisi bantuan sosial dari pemerintah senilai Rp7 juta sampai Rp50 juta yang akan dibagikan kepada masyarakat pada akhir bulan ini. Program bantuan ini disebut dapat dicairkan dengan melakukan pendaftaran melalui tautan yang dibagikan di media sosial. Masyarakat diminta segera mendaftarkan diri agar tidak kehilangan kesempatan mendapatkan bantuan tersebut."

Upon applying sentence splitting, the narrative was divided into four discrete sentences (K1–K4). Each sentence then underwent independent NLP preprocessing, and the resulting token sets are shown in Table 3.

Table 3. Preprocessing Results for Test Sentences (K1-K4)

Sent.	Original Sentence	Tokens After Preprocessing	Count
K1	Yang belum daftar buruan daftar sekarang.	daftar | buru | daftar	3
K2	Ternyata NIK KTP berisi bantuan sosial...	nik | ktp | isi | bantu | sosial | perintah | nilai | juta | juta | bagi | masyarakat | akhir | bulan	13
K3	Program bantuan ini disebut dapat dicairkan...	program | bantu | cair | daftar | tautan | bagi | media | sosial	8
K4	Masyarakat diminta segera mendaftarkan diri...	masyarakat | segera | daftar | diri | hilang | sempat | bantu	7

3.4 Prior Probability Calculation

The prior probability for each class is derived from its relative frequency within the training dataset:

$$\mathrm{P}(\mathrm{Non}-\mathrm{Hoax}) = \frac{5}{10} = 0.5 \quad \Rightarrow \quad log\ P(NH) = log10(0.5) = -0.3010 \tag{8}$$

$$\mathrm{P}(\mathrm{Hoax}) = \frac{5}{10} = 0.5 \quad \Rightarrow \quad log\ P(H) = log10(0.5) = -0.3010 \tag{9}$$

3.5 Likelihood with Laplace Smoothing

Using Equation (5), the denominators are: Non-Hoax denominator = 84 + 108 = 192; Hoax denominator = 62 + 108 = 170. Table 4 presents the likelihood calculation for the 15 most relevant terms appearing in the test data. Note that the TF (test) column represents the aggregate term frequency across all four test sentences (K1–K4) combined, used for the overall narrative-level likelihood reference.

Table 4. Likelihood Calculation with Laplace Smoothing

No	Term	P(t|NH) Formula	P(t|NH)	P(t|H) Formula	P(t|H)	TF (test)
1	daftar	(0+1)/192 = 1/192	0.005208	(0+1)/170 = 1/170	0.005882	4
2	bantu	(0+1)/192 = 1/192	0.005208	(0+1)/170 = 1/170	0.005882	3
3	sosial	(0+1)/192 = 1/192	0.005208	(1+1)/170 = 2/170	0.011765	2
4	masyarakat	(4+1)/192 = 5/192	0.026042	(1+1)/170 = 2/170	0.011765	2
5	perintah	(2+1)/192 = 3/192	0.015625	(0+1)/170 = 1/170	0.005882	1
6	bagi	(0+1)/192 = 1/192	0.005208	(0+1)/170 = 1/170	0.005882	2
7	media	(0+1)/192 = 1/192	0.005208	(1+1)/170 = 2/170	0.011765	1
8	program	(1+1)/192 = 2/192	0.010417	(0+1)/170 = 1/170	0.005882	1
9	juta	(0+1)/192 = 1/192	0.005208	(0+1)/170 = 1/170	0.005882	2
10	tautan	(0+1)/192 = 1/192	0.005208	(0+1)/170 = 1/170	0.005882	1
11	nik	(0+1)/192 = 1/192	0.005208	(0+1)/170 = 1/170	0.005882	1
12	ktp	(0+1)/192 = 1/192	0.005208	(0+1)/170 = 1/170	0.005882	1
13	segera	(0+1)/192 = 1/192	0.005208	(0+1)/170 = 1/170	0.005882	1
14	cair	(0+1)/192 = 1/192	0.005208	(0+1)/170 = 1/170	0.005882	1
15	buru	(0+1)/192 = 1/192	0.005208	(0+1)/170 = 1/170	0.005882	1

3.6 Log Posterior Probability per Sentence

Applying Equation (6), the log posterior probability for each of the four sentences is computed separately. The outcomes per sentence are presented in Table 5.

Table 5. Log Posterior Probability per Sentence

Sent.	Log P(Non-Hoax|Sentence) Calculation	Log P(Hoax|Sentence) Calculation	Result
K1	-0.3010 + 2×(-2.2833) + 1×(-2.2833) = -7.1509	-0.3010 + 2×(-2.2304) + 1×(-2.2304) = -6.9922	NH: -7.1509 H: -6.9922
K2	-0.3010 + Σ TF×log P(t|NH) = -28.8079	-0.3010 + Σ TF×log P(t|H) = -28.6948	NH: -28.8079 H: -28.6948
K3	-0.3010 + Σ TF×log P(t|NH) = -18.2664	-0.3010 + Σ TF×log P(t|H) = -17.5426	NH: -18.2664 H: -17.5426
K4	-0.3010 + Σ TF×log P(t|NH) = -15.5852	-0.3010 + Σ TF×log P(t|H) = -15.6131	NH: -15.5852 H: -15.6131

The detailed step-by-step calculation for K1 is used as a verification example; results for K2–K4 follow the same procedure and are summarized in Table 5.

3.7 Softmax Conversion and Final Classification

Using Equation (7), the log posterior scores for each sentence are transformed into percentage probabilities. Table 6 details the complete Softmax conversion outcomes along with the final averaged classification result.

Table 6. Softmax Conversion Results and Final Classification

Sent.	Log Post. NH	Log Post. H	exp(NH)	exp(H)	P(NH)	P(H)
K1	-7.1509	-6.9922	7.064×10^{-8}	1.018×10^{-7}	40.97%	59.03%
K2	-28.8079	-28.6948	1.557×10^{-29}	2.019×10^{-29}	43.53%	56.47%
K3	-18.2664	-17.5426	5.415×10^{-19}	2.867×10^{-18}	15.89%	84.11%
K4	-15.5852	-15.6131	2.599×10^{-16}	2.437×10^{-16}	51.61%	48.39%
Average					38.00%	62.00%
Final	HOAX (62.00%) | Actual Label: Hoax | Status: CORRECT					

The mean probabilities across all four sentences are calculated as:

$$\mathrm{P(Non-Hoax)} = \frac{(40.97\% + 43.53\% + 15.89\,\% + 51.61\%)}{4} = \frac{152.00\%}{4} = 38.00\% \quad (10)$$

$$\mathrm{P(Hoax)} = \frac{(59.03\% + 56.47\% + 84.11\% + 48.39\%)}{4} = \frac{248.00\%}{4} = 62.00\% \quad (11)$$

Following normalization, the narrative is ultimately labeled as Hoax with a 62.00% probability, which aligns with its true label. This outcome confirms that the model successfully captured language cues associated with hoax content, including urgent calls to action ('segera daftar') and references to social media link sharing ('tautan yang dibagikan di media sosial') [14]. A closer look at sentence K4 shows that it was individually labeled as Non-Hoax (P(NH)=51.61%), even though it contains the hoax-associated term 'daftar'. This is attributable to the presence of the word 'masyarakat' in K4, which has a considerably stronger Non-Hoax likelihood (P(masyarakat|NH)=0.026042) relative to its Hoax likelihood (P(masyarakat|H)=0.011765), thus tilting the log posterior marginally toward the Non-Hoax class for that sentence. This outcome is a known and expected behavior of the Naive Bayes model and has no bearing on the correct overall classification of the narrative.

3.8 System Testing Results

To verify that the developed web-based system operates correctly as a functional application, Black Box Testing was conducted on all major system features. Testing was performed by providing various input scenarios and comparing actual system outputs with expected outputs. All 13 test scenarios passed successfully, covering:

- Admin login with valid credentials, system redirected to dashboard
- Admin login with invalid credentials, system displayed an error message
- User submitting a valid news narrative, system returned per-sentence hoax and non-hoax probability percentages with a final average
- User submitting an empty input, system displayed a mandatory field warning
- Admin adding a complete dataset entry, data saved and appeared in the dataset table
- Admin adding an incomplete dataset entry, system displayed a mandatory field warning
- Admin editing a dataset entry, data updated correctly
- Admin deleting a single dataset entry, data removed from the dataset table
- Admin running model training, system displayed accuracy, precision, recall, F1-Score, and confusion matrix results
- Admin deleting a single classification result, data removed
- Admin deleting all classification results, all records removed
- Admin exporting data to CSV, CSV file downloaded with complete records
- Admin logout, session terminated and login page displayed. These results confirm that all functional requirements of the system are met and that the application operates as designed

Model performance was evaluated using a Confusion Matrix on a separate test dataset consisting of 50 sentences (25 Hoax, 25 Non-Hoax), split with an 80:20 ratio from a total of 250 labeled sentences collected from detik.com and turnbackhoax.id. The evaluation metrics Accuracy, Precision, Recall, and F1-Score are computed as follows using Equations (12)–(15):

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN) \quad (12)$$

$$\text{Precision} = TP / (TP + FP) \quad (13)$$

$$\text{Recall} = TP / (TP + FN) \quad (14)$$

$$\text{F1} - \text{Score} = 2 \times (Precision \times Recall) / (Precision + Recall) \quad (15)$$

Table 7. Confusion Matrix Evaluation Results

Class	TP	TN	FP	FN	Total
Hoax	22	23	2	3	25
Non-Hoax	23	22	3	2	25
Metric	Accuracy	Precision	Recall	F1-Score	Total Test
Value	90.00%	91.67%	88.00%	89.80%	50

The figures in Table 7 confirm that the system attained Accuracy of 90.00%, Precision of 91.67%, Recall of 88.00%, and F1-Score of 89.80%, with 45 of 50 test sentences correctly classified. Comparing these results with prior studies, reported a Naive Bayes accuracy of approximately 79% on Indonesian hoax news data[8], while other study achieved 84% using a similar approach [9]. Previous study reported Multinomial Naive Bayes accuracy in the range of 85–90% in a multi-algorithm comparison[10]. The present system's 90.00% accuracy is therefore competitive and situated at the upper end of this range, which can be attributed to the quality of the NLP preprocessing pipeline, the use of reliably labeled training data from detik.com and turnbackhoax.id, and the per-sentence classification design that prevents a single strongly hoax-indicative sentence from being diluted by neutral ones. Regarding error analysis, the 5 misclassified sentences (5 False Positives and 5 False Negatives across both classes) are likely attributable to two causes: sentences that contain hoax-associated vocabulary (e.g., 'beredar', 'klaim') despite coming from credible sources, and sentences from hoax content that use factual or neutral phrasing that the bag-of-words model cannot distinguish from non-hoax language. This is an inherent limitation of the Naive Bayes assumption of feature independence, which ignores semantic

context. From a practical standpoint, the system delivers transparent, per-sentence probability outputs that enable users to identify which specific parts of a news narrative carry hoax signals, a design feature absent from most prior systems that report only a single document-level label. This transparency supports media literacy efforts by helping non-expert users understand the linguistic characteristics of misinformation rather than simply receiving a binary verdict. The admin interface, which allows continuous dataset updates and model retraining, further ensures the system can adapt as hoax patterns evolve over time. These outcomes align with evidence from related studies showing Multinomial Naive Bayes reliably achieves strong results in Indonesian text classification [15]. The competitive performance can be attributed to distinctive hoax vocabulary patterns, sensationalist verbs, unverified source references, and urgent calls to action, that serve as reliable class-discriminating features under the bag-of-words assumption, amplified by TF-IDF weighting [16]. Nevertheless, with only 200 training sentences, the model may overfit to domain-specific vocabulary from detik.com and turnbackhoax.id; linguistic diversity such as regional dialects, code-switching, and sarcasm remains an unaddressed challenge. The per-sentence output approach offers a more detailed analytical perspective compared to prior works reporting only a single aggregate result [17]. The selection of Multinomial Naive Bayes as the primary classification method was deliberate: the algorithm offers high computational efficiency suitable for real-time web-based deployment, strong interpretability because per-word probabilities are explicitly traceable, and established effectiveness for high-dimensional TF-IDF text data in the Indonesian-language context. The absence of experimental comparisons with other classifiers such as SVM, Logistic Regression, or Random Forest is acknowledged as a limitation of the present study; such comparisons are recommended for future work to identify the most optimal approach for this task. Likewise, integrating transformer-based language models built upon the BERT architecture may further improve contextual understanding beyond TF-IDF-based representations [19]. Similar improvements through the combination of NLP techniques and Multinomial Naive Bayes have also been reported in previous studies [20]. Regarding evaluation metrics, the present study reports Accuracy, Precision, Recall, and F1-Score, which are appropriate for binary classification on balanced datasets. ROC Curve and AUC metrics were not included because the system produces Softmax-converted probability percentages per sentence rather than a single binary decision score, making standard ROC construction less straightforward in this per-sentence aggregation setting. Future studies that adopt a configurable decision threshold could meaningfully incorporate ROC analysis to provide additional insight into the sensitivity-specificity trade-off of the model.

4. Conclusions and Future Works

This study has successfully developed and deployed a web-based hoax news detection system built on the Multinomial Naive Bayes algorithm combined with Natural Language Processing methods. The system runs text through a five-stage NLP pipeline covering sentence splitting, case folding, tokenizing, stopword removal, and stemming, followed by TF-IDF feature extraction and Multinomial Naive Bayes classification with Laplace Smoothing. Log posterior values are then processed via the Softmax function to yield interpretable probability scores per sentence.

A manual step-by-step computation using 10 annotated training sentences and one test narrative confirmed that the system accurately labeled the narrative as hoax with 62.00% probability (ground truth: hoax), thereby validating the algorithm implementation. The sentence-by-sentence detection enabled by the splitting stage provides users with finer-grained insight into hoax indicators within any narrative, setting this work apart from prior studies that deliver only document-level classification outputs.

Performance evaluation via Confusion Matrix on a 50-sentence test set yielded Accuracy of 90.00%, Precision of 91.67%, Recall of 88.00%, and F1-Score of 89.80%, confirming the reliability of the approach. For future work, expanding the training corpus would improve generalization, and integrating real-time news API feeds would allow the dataset to be continuously refreshed. Supporting multi-label classification by topic category (health, politics, social affairs) could further improve system utility. Practical limitations must also be acknowledged: evolving language patterns, code-switching, sarcasm, and irony pose challenges to keyword-based probabilistic models that the Naive Bayes bag-of-words assumption cannot address. Future iterations

could explore integration of transformer-based models such as IndoBERT, built upon the BERT architecture and pre-trained on a large Indonesian-language corpus, to capture contextual semantics and subtle linguistic cues that TF-IDF weighting alone cannot detect.

5. References

[1] Asosiasi Penyelenggara Jasa Internet Indonesia, "Jumlah pengguna internet Indonesia tembus 221 juta orang," 2024.

[2] Kementerian Komunikasi dan Informatika Republik Indonesia, "Siaran Pers No. 02/HM/KOMINFO/01/2024 tentang Hingga Akhir Tahun 2023, Kominfo Tangani 12.547 Isu Hoaks," 2024.

[3] Kementerian Komunikasi dan Digital Republik Indonesia, "Komdigi Identifikasi 1.923 Konten Hoaks Sepanjang Tahun 2024," 2025.

[4] H. Allcott and M. Gentzkow, "Social Media and Fake News in the 2016 Election," *J. Econ. Perspect.*, vol. 31, no. 2, pp. 211–236, 2017, doi: 10.1257/jep.31.2.211.

[5] S. Vosoughi, D. Roy, and S. Aral, "The spread of true and false news online," *Science (80-.).*, vol. 359, no. 6380, pp. 1146–1151, 2018, doi: 10.1126/science.aap9559.

[6] C. Pelau, M. I. Pop, M. Stanescu, and G. Sanda, "The Breaking News Effect and Its Impact on the Credibility and Trust in Information Posted on Social Media," *Electronics*, vol. 12, no. 2, 2023, doi: 10.3390/electronics12020423.

[7] S. H. Daulay, D. N. Aulia, and N. A. Zahra, "Framing The Lie: A Linguistic Analysis of Viral Fake News Discourse," *BASIS J. Bhs. dan Sastra Ingg.*, vol. 12, no. 2, pp. 253–264, 2025, doi: 10.33884/basisupb.v12i2.10000.

[8] R. R. Sani, Y. A. Pratiwi, S. Winarno, E. D. Udayanti, and F. Al Zami, "Analisis Perbandingan Algoritma Naive Bayes Classifier dan Support Vector Machine untuk Klasifikasi Hoax pada Berita Online Indonesia," *J. Masy. Inform.*, vol. 13, no. 2, 2022, doi: 10.14710/jmasif.13.2.47983.

[9] N. E. Febriyanty, M. A. Hariyadi, and C. Crysdian, "Hoax Detection News Using Naive Bayes and Support Vector Machine Algorithm," *Int. J. Adv. Data Inf. Syst.*, vol. 4, no. 2, pp. 191–200, 2023, doi: 10.25008/ijadis.v4i2.1306.

[10] G. Airlangga, "Comparative Analysis of Machine Learning Algorithms for Detecting Fake News: Efficacy and Accuracy in the Modern Information Ecosystem," *J. Comput. Networks, Archit. High Perform. Comput.*, vol. 6, no. 1, pp. 354–363, 2024, doi: 10.47709/cnahpc.v6i1.3466.

[11] W. Hidayat, J. Ong, H. Irsyad, and A. Rahman, "Ekstrasi Berita Hoax Pada Turn Back Hoax Berbasis Pendekatan TF-IDF \& Cosine Similarity," *J. Ilm. Comput. Insight*, vol. 7, no. 2, 2025, doi: 10.30651/comp_insight.v7i2.26678.

[12] M. F. Ramadhan, "Klasifikasi Topik dan Sentimen Judul Berita dengan Augmentasi dan TF-IDF," *RIGGS J. Artif. Intell. Digit. Bus.*, vol. 4, no. 2, pp. 6732–6741, 2025, doi: 10.31004/riggs.v4i2.1692.

[13] T. F. Mustafa and H. Alfianti, "Klasifikasi Berita Palsu Berbahasa Indonesia Menggunakan Algoritma Naive Bayes Berbasis Web," *J. Sains Inform. Terap.*, 2025, [Online]. Available: https://doi.org/10.62357/jsit.v4i3.564

[14] M. F. Ansyori and A. H. Mujianto, "Penerapan Natural Language Processing (NLP) dengan Metode Cosine Similarity pada Sistem E-Monev untuk Pencarian Program Pembangunan Daerah," *J. Software, Hardw. Inf. Technol.*, vol. 5, no. 2, pp. 84–102, 2025, doi: 10.24252/shift.v5i2.183.

[15] D. Rifaldi and others, "Evaluasi Sentimen Pengguna ChatGPT Menggunakan Naive Bayes: Tinjauan dari Confusion Matrix dan Classification Report," *J. Ris. Sist. dan Teknol. Inf.*, vol. 3, no. 2, pp. 81–89, 2025, [Online]. Available: https://doi.org/10.30787/restia.v3i2.1990

[16] O. N. Cahyani and F. Budiman, "Performa Logistic Regression dan Naive Bayes dalam Klasifikasi Berita Hoax di Indonesia," *Edumatic J. Pendidik. Inform.*, vol. 9, no. 1, pp. 60–68, 2025, doi: 10.29408/edumatic.v9i1.28987.

[17] A. Fardhina, R. M. Siregar, M. R. W. Br Sibarani, I. C. Br Ginting, and A. Pratama, "Sistem Deteksi Berita Hoaks berbasis Algoritma Natural Language Processing (NLP) menggunakan BERT," *J. Manaj. Inform. Sist. Inf. Dan Teknol. Komput.*, vol. 4, no. 1, pp. 450–461, 2025, doi: 10.70247/jumistik.v4i1.156.

[18] B. Wilie *et al.*, "IndoNLU: Benchmark and Resources for Evaluating Indonesian Natural Language Understanding," in *Proc. 1st Conf. of the Asia-Pacific Chapter of the Association for Computational Linguistics*, 2020, pp. 843–857. doi: 10.18653/v1/2020.aacl-main.85.

[19] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proc. 2019 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2019, pp. 4171–4186. doi: 10.18653/v1/N19-1423.

[20] R. Fernando, Y. D. Proboningrum, S. D. Supriati, and Nurmalitasari, "NLP Implementation for AI Generated Text Detection (ChatGPT) Using Naive Bayes Method," *J-INTECH (Journal of Information and Technology)*, vol. 13, no. 2, 2025, doi: 10.32664/j-intech.v13i02.2026.