

Indonesian Online News Topic Classification Using Naive Bayes and Support Vector Machine

Winando Parbo Kusuma^{1*}, Pradita Eko Prasetyo Utomo², Mutia Fadhila Putri³

^{1,2,3}Universitas Jambi, Indonesia

^{1*}winandokusuma5995@gmail.com, ²pradita.eko@unjia.ac.id, ³Mutia.fadhila@unjia.ac.id



*Corresponding Author

Article History:

Submitted: 19-06-2026

Accepted: 29-06-2026

Published: 05-07-2026

Keywords:

Indonesian online news; Naive Bayes; Support Vector Machine; Text Classification; TF-IDF; Web Scraping.

Brilliance: Research of

Artificial Intelligence is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0).

ABSTRACT

Indonesian online news portals publish large volumes of articles every day, making manual topic grouping time-consuming, inconsistent, and difficult to maintain when articles contain overlapping contexts. Objective: This study develops and compares Naive Bayes and support vector machine models for Indonesian online news topic classification and identifies the model that can be used as the basis for an automatic newsroom support tool. Methods: A dataset of 1,631 Kompas.com articles was collected through web scraping from six channels: economy, automotive, technology, lifestyle, politics, and sport. The article body was used as input, while the source channel became the class label. The text was prepared through duplicate checking, text normalization, case folding, tokenizing, filtering with Indonesian and custom stopwords, and stemming with Sastrawi. The processed texts were transformed with TF-IDF using a maximum of 10,000 unigram and bigram features, min_df=2, and max_df=0.9. The data were split into 80% training data and 20% testing data with stratification. GridSearchCV with five-fold cross validation was applied to tune Multinomial Naive Bayes and support vector machine parameters. Results: Naive Bayes achieved 97.25% testing accuracy, while support vector machine with a linear kernel, C=10, and gamma=scale achieved 98.17%. SVM also produced higher macro precision, recall, and F1-score. Remaining errors mainly appeared in technology articles because their vocabulary overlapped with lifestyle, automotive, and politics topics. Conclusion: TF-IDF with linear SVM effectively classifies Indonesian online news topics and supports automated content organization workflows.

INTRODUCTION

The development of online media has made news production and distribution faster than before. News can be published shortly after an event occurs, allowing the public to obtain information in real time through digital news portals. Online portals also provide various channels so that readers can find information according to their needs (Habib et al., 2022). In addition, online news portals function as structured information sources that collect and present news from many fields. Although news portals already provide topic channels, the grouping process can still be too general for readers who need more specific information. Manual classification also requires each article to be read and interpreted before it is assigned to the most suitable category. This process becomes inefficient when the number of articles increases and when the content of different news categories is similar. Previous studies explain that similarity between news topics can make manual classification difficult and requires extra accuracy (Nanda et al., 2022). Therefore, an automatic topic classification approach is needed to support more consistent and efficient news organization.

Along with advances in data processing and computing, classification techniques have been widely used to handle data according to its characteristics. Classification is the process of finding patterns or functions that separate one data class from another so that the model can predict the class of new data (Mayasari et al., 2024). In the context of information systems, news topic classification is not only useful for helping readers find relevant articles, but also for supporting digital content governance. Consistent topic grouping can support content indexing, article recommendation systems, personalized news pages, targeted advertising, and topic trend analysis based on reader interests (Solahuddin et al., 2023).

Text mining is one approach that can be used to solve the problem of manual news classification. In online news processing, text mining helps extract relevant information and hidden patterns from large collections of unstructured text (Afdal & Elita, 2022). Text mining also allows documents to be grouped into certain categories based on word similarity and frequently occurring terms, making document analysis faster, more efficient, and more consistent than manual processing. In this study, topic classification focuses on six classes of Indonesian online news: economy, automotive, technology, lifestyle, politics, and sport (Utami & Eka Putra, 2022).



Before classification can be performed, news data must be collected and cleaned. This study uses web scraping to collect article data from Kompas.com based on news channels. Web scraping refers to the process of retrieving semi-structured documents from the internet, such as HTML pages, and extracting specific data for further use (Djiwadikusumah & Irawan, 2021). The collected attributes include title, publication date, article body, URL, and label. The article text is then cleaned from irrelevant HTML elements, characters, and duplicate data.

The classification methods used in this study are Naive Bayes and Support Vector Machine. Naive Bayes predicts class labels using probability patterns from training data and is often selected for text classification because it is simple, fast, and useful as a baseline model (Handoko, 2021). Support Vector Machine is used as a comparison method because it can handle high-dimensional text data by finding the best separating hyperplane between classes. Previous research on online news classification showed that SVM can produce good performance after preprocessing and TF-IDF weighting (Nanda et al., 2022). Naive Bayes has also been reported to classify online news effectively with high accuracy. In that study, the algorithm was applied to online news articles that had passed several stages, including data collection, labeling, preprocessing, term weighting, and classification. The result shows that Naive Bayes can learn the distribution of words that frequently appear in each news topic and use these probability patterns to predict the category of new articles. Therefore, Naive Bayes is relevant to be used in this study as an efficient baseline model for Indonesian online news topic classification (Habib et al., 2022).

This research aims to build and compare automatic classification models for Indonesian online news topics using Naive Bayes and Support Vector Machine. The research problem focuses on several stages. These stages include collecting Indonesian online news data through channel-based web scraping, preprocessing the dataset, building classification models using Naive Bayes and SVM, evaluating model performance using multiclass confusion matrix, accuracy, precision, recall, and F1-score, and identifying the best-performing model.

LITERATURE REVIEW

Online News

Online news refers to news content published through internet-based media. Compared with printed media, online news has characteristics such as fast publication, easy access, digital archiving, multimedia presentation, and interaction with readers through comments or social media (Puspitasari & Wulandari, 2024). News portals commonly provide topic channels such as economy, politics, technology, automotive, sport, and lifestyle to help readers browse information more efficiently (Insani & Zamzamy, 2023). In this study, online news is treated as textual data that can be processed and classified according to topic labels.

Text Mining

Text mining is the process of extracting useful information from text data using analytical tools and computational techniques. In general, text mining adopts several processes from data mining and applies them to unstructured text data (Joergensen E Munthe et al., 2022). Text mining, also known as text data mining or text analytics, aims to identify patterns, trends, and hidden knowledge from large collections of text, such as articles, emails, social media posts, books, reports, and other textual documents. This process combines several disciplines, including Natural Language Processing (NLP), statistics, and machine learning, to analyze text and produce insights that can support decision-making, trend understanding, and service or product improvement (Hartati et al., 2025).

In a news portal context, text mining can support editorial analytics, topic monitoring, article recommendation, and content management. A common pipeline for text mining includes text normalization, case folding, tokenizing, stopword removal, stemming, feature extraction, model training, and evaluation.



Figure 1. The data preprocessing stage (Rianto et al., 2021)

Figure 1 explains the preprocessing workflow applied to the raw news text before feature extraction. The process begins with text normalization to remove numbers, punctuation, redundant spaces, HTML-related artifacts, and irrelevant characters. After that, case folding converts all characters into lowercase, tokenizing splits the text into word tokens, filtering removes stopwords and unimportant tokens, and stemming reduces words into their root forms.

TF-IDF

Term Frequency (TF) indicates how often a word appears in a document, while Inverse Document Frequency (IDF) indicates how important the word is by considering the number of documents that contain it. A word that frequently appears in one document but rarely appears in other documents will have a higher weight. Conversely, a word that is too common and appears in many documents will have a lower weight (Asyhar et al., 2024). TF-IDF is a widely used feature weighting method for text classification. Term Frequency measures how often a term appears in a

document, while Inverse Document Frequency reduces the weight of terms that appear in many documents. Therefore, terms that are frequent in one document but less common across the corpus receive higher weights (Arifin et al., 2021).

$$TF - IDF(t, d) = TF(t, d) \times IDF(t) \quad (1)$$

$$IDF(t) = \log N / (df(t)) \quad (2)$$

$$TF - IDF_{td} = TF_{td} \times IDF_{td} \quad (3)$$

Naive Bayes

Naive Bayes is a probabilistic classifier based on Bayes theorem and the assumption of conditional independence among features. In text classification, Multinomial Naive Bayes is commonly used because it is efficient for word-frequency or weighted word-feature representations. Previous work by (Habib et al., 2022) used Naive Bayes to classify online news and reported strong classification performance, while (Astuti et al., 2024) and (Nur'Faradila et al., 2024) also showed that Naive Bayes can perform well in Indonesian text classification tasks.

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

Support Vector Machine

Support Vector Machine is a supervised learning algorithm that searches for an optimal separating hyperplane between classes. It is suitable for text classification because text data often produces high-dimensional sparse feature spaces after TF-IDF transformation. (Nanda et al., 2022) reported that SVM was effective for online news classification, and (Hartati et al., 2025) applied SVM in sentiment analysis with topic modeling. Based on these findings, this study compares Naive Bayes and SVM to determine which method performs better for six-class Indonesian online news topic classification.

$$f_x = w \cdot x + b \quad (5)$$

METHOD

This study develops an Indonesian online news topic classification model by applying Natural Language Processing and machine learning techniques. The research focuses on transforming unstructured news articles into structured textual features using TF-IDF and comparing the performance of Naive Bayes and Support Vector Machine. The research process is carried out through four main stages: data collection, text preprocessing, feature extraction and model classification, and model evaluation with prototype implementation.

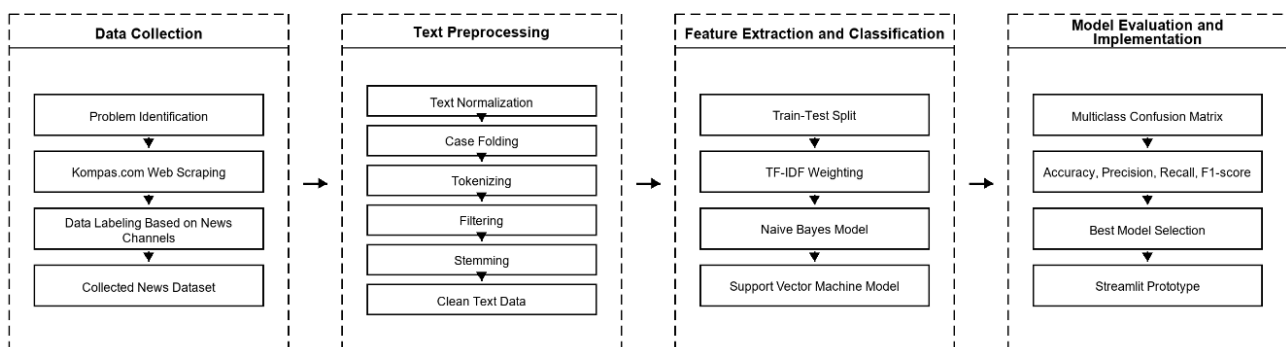


Figure 2. Research Methodology Workflow

Data Collection

```

import requests
from bs4 import BeautifulSoup
from urllib.parse import urljoin, urlsplit, urlunsplit
import re
import pandas as pd

base_url = "https://lifestyle.kompas.com/fashion"
  
```

Figure 3. Web Scraping code

This study used an experimental text classification workflow. The dataset was collected from Kompas.com using Python-based web scraping. Six channels were selected as class sources: economy, automotive, technology, lifestyle, politics, and sport. Each article collected from a channel was automatically labeled according to the channel topic. The collected attributes included title, publication date, article body, URL, and label, but only the article body and label were used as the main classification variables.

The scraping process used Requests to access webpages, BeautifulSoup to parse HTML content, URL processing utilities to normalize article links, regular expressions to match URL and date patterns, and Pandas to store the dataset. Article URLs were filtered so that only valid article pages were collected. The article body was extracted from the main content section, while irrelevant paragraphs such as recommendations or advertisement text were excluded. Duplicate data were checked using the URL column, and the final dataset contained 1,631 unique news articles.

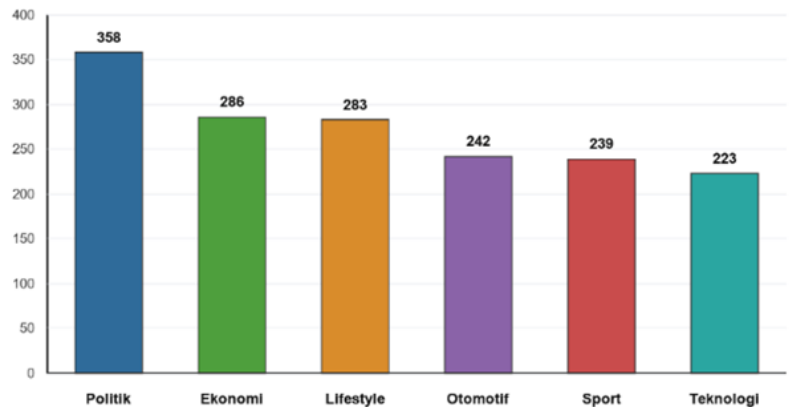


Figure 4. Distribution Dataset

Text Preprocessing

Text preprocessing was applied before feature extraction. First, text normalization removed numbers, punctuation, redundant spaces, HTML-related artifacts, and irrelevant characters. Second, case folding converted all text to lowercase. Third, tokenizing split the article body into word tokens. Fourth, filtering removed Indonesian stopwords and additional custom stopwords such as portal names, editorial labels, advertisement words, and short tokens with little classification value. Finally, stemming used Sastrawi to convert Indonesian words into their base forms.

Table 1. Text Preprocessing Pipeline

Preprocessing	Before	After
Text Normalization	KOMPAS.com - Di era media sosial, membandingkan diri dengan orang lain menjadi kebiasaan yang sulit dihindari.	Di era media sosial, membandingkan diri dengan orang lain menjadi kebiasaan yang sulit dihindari.
Case Folding	Di era media sosial, membandingkan diri dengan orang lain menjadi kebiasaan yang sulit dihindari.	di era media sosial membandingkan diri dengan orang lain menjadi kebiasaan yang sulit dihindari
Tokenizing	di era media sosial membandingkan diri dengan orang lain menjadi kebiasaan yang sulit dihindari	["di", "era", "media", "sosial", "membandingkan", "diri", "dengan", "orang", "lain", "menjadi", "kebiasaan", "yang", "sulit", "dihindari"]
Filtering	["di", "era", "media", "sosial", "membandingkan", "diri", "dengan", "orang", "lain", "menjadi", "kebiasaan", "yang", "sulit", "dihindari"]	["era", "media", "sosial", "membandingkan", "diri", "orang", "lain", "menjadi", "kebiasaan", "sulit", "dihindari"]
Stemming	["era", "media", "sosial", "membandingkan", "diri", "orang", "lain", "menjadi", "kebiasaan", "sulit", "dihindari"]	["era", "media", "sosial", "banding", "diri", "orang", "biasa", "sulit", "hinder"]

Table 1 describes the text preprocessing pipeline used in this study by showing the transformation of news text at each stage. The table illustrates how raw article text is gradually cleaned through text normalization, case folding, tokenizing, filtering, and stemming. This step-by-step transformation helps reduce noise, standardize word forms, and prepare the article content for TF-IDF feature extraction.

TF-IDF

TF-IDF was used to convert text into numerical features. The implementation used scikit-learn TfidfVectorizer with `max_features = 10000`, `ngram_range = (1,2)`, `min_df = 2`, and `max_df = 0.9`. The feature extractor was fitted only on training data and then applied to testing data to avoid data leakage. The dataset was split into 80% training data and 20% testing data using `train_test_split` with `random_state = 42` and `stratify = y` so that the class proportions remained represented in both subsets.

Table 2. Dataset Distribution and Stratified Train-Test Split

Class	Split Dataset		Total
	Training 80%	Testing 20%	
Politik	286	72	358
Ekonomi	229	57	286
Otomotif	194	48	242
Lifestyle	226	57	283
Sport	191	48	239
Teknologi	178	45	223
	1304	327	1631

Table 2 shows the dataset distribution and the stratified train-test split for each news class. Each class was divided into 80% training data and 20% testing data, resulting in 1,304 training articles and 327 testing articles. The stratified split was used to ensure that each topic class maintained a similar proportion in both subsets, making the model evaluation more representative and fair.

Algorithm Implementation

The algorithm implementation stage was conducted after the Indonesian online news articles had been collected, cleaned, and prepared through several text preprocessing steps. At this stage, the textual data that originally consisted of unstructured article content was transformed into a structured numerical representation so that it could be processed by machine learning algorithms.

This study implemented two supervised classification algorithms, namely Multinomial Naive Bayes and Support Vector Machine, to classify Kompas.com news articles into six topic categories: economy, automotive, technology, lifestyle, politics, and sport. The comparison between these two algorithms was carried out to determine which model produced better classification performance on Indonesian online news text. The implementation process included data splitting, TF-IDF feature extraction, hyperparameter tuning, model training, prediction, and model evaluation.

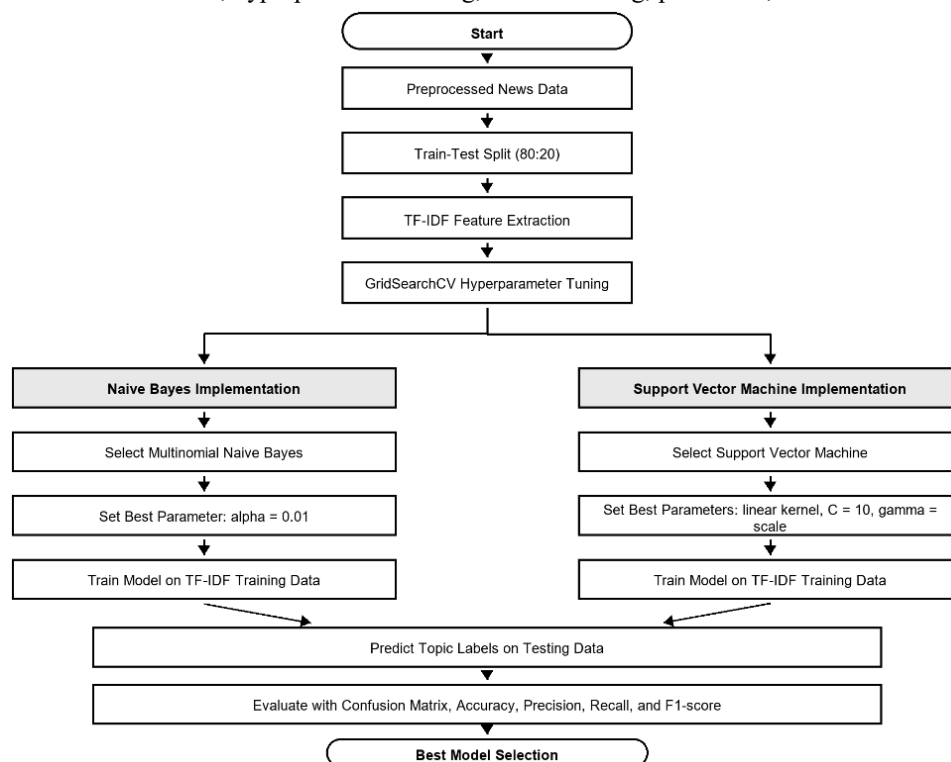


Figure 5. Flowchart Algorithm Implementation

The implementation process begins with preprocessed news data that has passed text normalization, case folding, tokenizing, filtering, and stemming stages. The clean dataset is then divided into training and testing subsets using an 80:20 stratified split, where the training data are used to build the classification models and the testing data are used to measure model performance on unseen news articles. After the data split, TF-IDF is applied to convert the processed text into numerical feature vectors by assigning weights to terms based on their frequency in a document and their importance across the entire dataset.

GridSearchCV with five-fold cross validation is then used to find the best hyperparameter combination for each algorithm. In the Naive Bayes branch, the Multinomial Naive Bayes model is trained using the best alpha value of 0.01. In the Support Vector Machine branch, the model is trained using the best parameters consisting of a linear kernel, C=10, and gamma=scale. After training is completed, both models predict the topic labels of the testing data. The prediction results are then evaluated using a multiclass confusion matrix, accuracy, precision, recall, and F1-score. The model with the highest and most stable evaluation results is selected as the best classification model for Indonesian online news topic classification.

Visualization Implementation

The visualization interface displays the predicted news category and the classification score for each topic class, including economy, automotive, technology, lifestyle, politics, and sport. This output helps users observe not only the final prediction but also the confidence tendency of the model across all available categories. Therefore, the visualization prototype can support users in understanding how the classification model works and can be further developed as a newsroom support tool for article categorization, topic monitoring, and digital content organization.

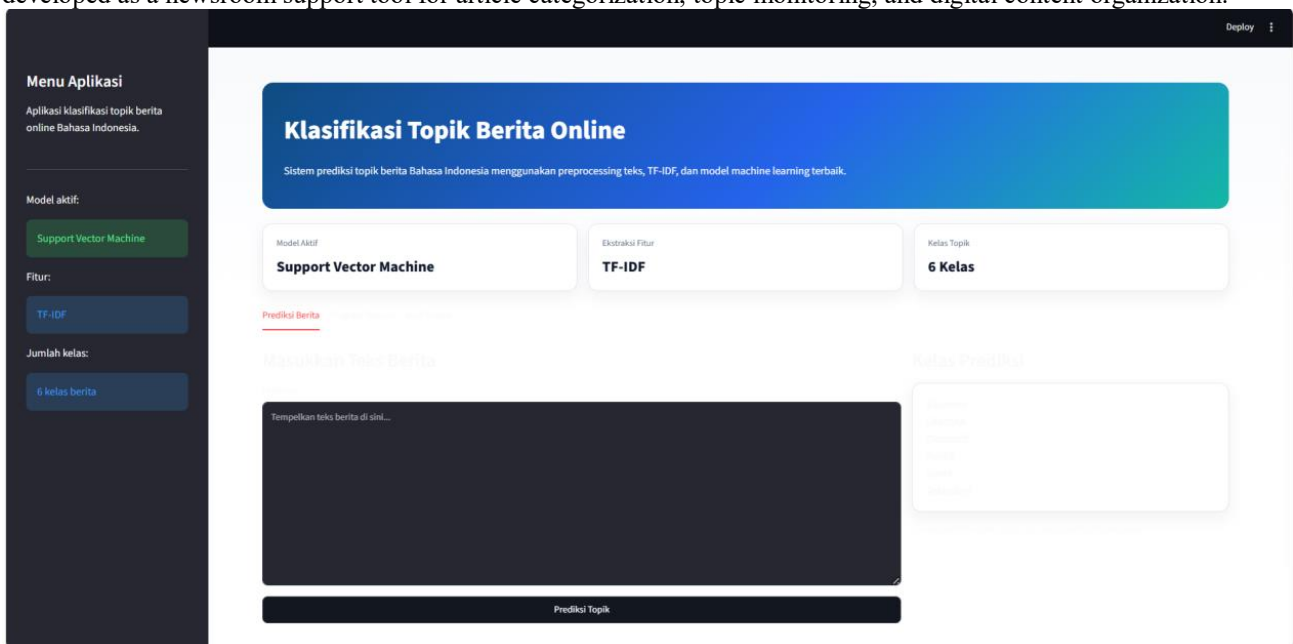


Figure 6. Streamlit-Based Visualization of News Topic Classification Results

Evaluation

The evaluation metrics used in this study include accuracy, precision, recall, and F1-score. Accuracy measures the overall proportion of correctly classified news articles. Precision measures how many articles predicted as a certain class actually belong to that class, while recall measures how many articles from a certain class are correctly identified by the model. F1-score is used to balance precision and recall, especially when evaluating multiclass classification performance.

$$Accuracy = \frac{\sum \text{correct prediction}}{\text{Total data}} \quad (6)$$

$$Precision\ y = \frac{1}{k} \sum_{i=1}^k \frac{TP_i}{TP_i + FP_i} \quad (7)$$

$$Recall\ y = \frac{1}{k} \sum_{i=1}^k \frac{TP_i}{TP_i + FN_i} \quad (8)$$

$$F1 - Score\ y = 2 \frac{(Precision \times Recall)}{(Precision + Recall)} \quad (9)$$

- k : number of classes
- y : class index
- TP_i : true positive for class i
- FP_i : false positive for class i
- FN_i : false negative for class i
- TN_i : true negative for class i

The evaluation results are then summarized using macro average and weighted average. Macro average calculates the mean value of a metric across all classes without considering the number of data in each class. Therefore, each class has the same contribution to the final score.

$$\text{Macro Average} = \frac{1}{k} \sum m_i \quad (10)$$

Weighted average calculates the mean value of a metric by considering the number of data in each class. Classes with more testing data have a greater influence on the final weighted score.

$$\text{Weighted Average} = \frac{\sum(n_i \times m_i)}{\sum(n_i)} \quad (11)$$

- m_i = metric value for class i
- n_i = number of testing data in class i
- k = total number of classes

The use of macro average and weighted average is important in this study because the number of articles in each news topic is not completely balanced. Macro average is used to evaluate the model performance equally across all classes, while weighted average is used to evaluate model performance based on the proportion of data in each class.

RESULT

Hyperparameter Tuning

GridSearchCV with five-fold cross validation was used to search for the best parameter configuration for each model. This process was carried out to ensure that the model was trained using the most suitable parameter values before being evaluated on the testing data. For Multinomial Naive Bayes, the tested parameter was alpha, which controls the smoothing value used to handle word features with very small or zero probability. Meanwhile, for Support Vector Machine, the tested parameters were C, kernel, and gamma, which influence the model's ability to form an optimal decision boundary in the TF-IDF feature space.



Figure 7. Hyperparameter tuning results using GridSearchCV

The hyperparameter tuning results for both classification models using GridSearchCV with five-fold cross validation. In the Naive Bayes model, six alpha values were tested, resulting in 30 training processes. The best performance was obtained when alpha = 0.01, with a cross-validation accuracy of 96.55%.

The Support Vector Machine model, 45 parameter combinations were tested using the parameters C, kernel, and gamma, resulting in 225 training processes. The best result was obtained using C = 10, kernel = linear, and gamma = scale, with a cross-validation accuracy of 97.39%. These results show that SVM produced better validation performance

than Naive Bayes, so SVM was selected for the final model evaluation stage.

Table 3. Hyperparameter Tuning Results

Model	Tuned Parameters	Best Parameter	CV Accuracy
Multinomial Naive Bayes	alpha = 0.01, 0.1, 0.5, 1.0, 2.0, 5.0	alpha = 0.01	96.55%
Support Vector Machine	C = 0.1, 1, 10; kernel = linear, poly, rbf; gamma = scale, auto, 0.001, 0.01, 0.1	C = 10, kernel = linear, gamma = scale	97.39%

Naïve Bayes Evaluation

The best alpha value was obtained, the Naive Bayes model was trained using the training data and tested using the testing data. The confusion matrix in Fig. 8 shows that Naive Bayes correctly classified 318 of 327 testing articles. The model classified all automotive, politics, and sport articles correctly, while the remaining errors appeared in economy, lifestyle, and technology classes.

	Predicted Class					
	Economy	Lifestyle	Automotive	Politics	Sport	Technology
Economy	55	0	0	2	0	0
Lifestyle	0	55	1	0	0	1
Automotive	0	0	48	0	0	0
Politics	0	0	0	72	0	0
Sport	0	0	0	0	48	0
Technology	1	1	2	1	0	40

Figure 8. Confusion matrix of Naive Bayes model

Naive Bayes achieved 97.25% accuracy. The technology class obtained the lowest recall value of 0.89 because several technology articles were predicted as other classes. This indicates that technology articles contain vocabulary that overlaps with economy, lifestyle, automotive, and politics topics.

Table 4. Classification report of Naive Bayes

	Precision	Recall	F1-Score	Support
Ekonomi	0.98	0.96	0.97	57
Lifestyle	0.98	0.96	0.97	57
Otomotif	0.94	1.00	0.97	48
Politik	0.96	1.00	0.98	72
Sport	1.00	1.00	1.00	48
Teknologi	0.98	0.89	0.93	45
Accuracy			0.97	327
Macro Avg	0.97	0.97	0.97	327
Weighted Avg	0.97	0.97	0.97	327

Support Vector Machine Evaluation

The best SVM model was evaluated using the same testing data. Fig. 9 shows that SVM correctly classified 321 of 327 testing articles. Similar to Naive Bayes, SVM classified all automotive, politics, and sport articles correctly. However, SVM produced fewer errors in economy, lifestyle, and technology articles.

	Predicted Class					
	Economy	Lifestyle	Automotive	Politics	Sport	Technology
Economy	56	0	0	1	0	0
Lifestyle	0	56	1	0	0	0
Automotive	0	0	48	0	0	0
Actual Class						
Politics	0	0	0	72	0	0
Sport	0	0	0	0	48	0
Technology	0	1	2	1	0	41

Figure 9. Confusion matrix of Support Vector Machine model

SVM achieved 98.17% accuracy. The recall of the technology class increased from 0.89 in Naive Bayes to 0.91 in SVM. This result indicates that SVM was slightly better at separating semantically overlapping news topics in the TF-IDF feature space.

Table 5. Classification report of Support Vector Machine

	Precision	Recall	F1-Score	Support
Ekonomi	1.00	0.98	0.99	57
Lifestyle	0.97	0.98	0.97	57
Otomotif	0.96	1.00	0.98	48
Politik	0.97	1.00	0.99	72
Sport	1.00	1.00	1.00	48
Teknologi	1.00	0.91	0.95	45
Accuracy			0.98	327
Macro Avg	0.98	0.98	0.98	327
Weighted Avg	0.98	0.98	0.98	327

Best Model Selection

The final model selection was not only based on accuracy, but also on macro average and weighted average. Macro average is important because it gives equal weight to each class, while weighted average considers the number of testing articles in each class.

Table 6. Final comparison based on macro and weighted average

Model	Accuracy	Precision Macro	Recall Macro	F1-Score Macro
Naive Bayes	97,25%	97%	97%	97%
SVM	98,17%	98%	98%	98%

Support Vector Machine achieved the highest accuracy, macro precision, macro recall, macro F1-score, and weighted F1-score. Therefore, SVM was selected as the best model for Indonesian online news topic classification. The better performance of SVM is related to its ability to construct an optimal separating hyperplane in a high-dimensional sparse TF-IDF feature space.

DISCUSSION

This study has several limitations. First, the dataset was collected only from Kompas.com, so the model may not fully represent the writing style and topic distribution of other Indonesian news portals. Second, the labels were assigned automatically based on news channels, which may still contain articles with overlapping contexts between classes. Third, the study only used TF-IDF as the feature extraction method, so contextual word meaning was not captured as deeply as in embedding-based or transformer-based models. Therefore, future studies can use multiple news portals, improve label validation, and compare TF-IDF with more contextual text representation methods.

The experimental results indicate that TF-IDF can provide a strong textual representation for Indonesian online news topic classification. Both Naive Bayes and SVM achieved high accuracy because the six news channels contain distinguishable topical vocabularies. However, SVM performed slightly better because it is able to construct a more

optimal decision boundary in a high-dimensional sparse feature space produced by TF-IDF.

The difference between the two models was not large, but it was consistent across accuracy, macro average, and weighted average. Naive Bayes remained effective as a baseline model because it is simple and computationally efficient. Nevertheless, its conditional independence assumption may limit its ability to capture relationships between words. SVM does not rely on this assumption and can separate classes more effectively when many TF-IDF features are involved.

The misclassification analysis shows that most errors occurred in semantically overlapping topics. Economy articles can contain political terms when they discuss government policy, tax, subsidies, or public budgets. Lifestyle articles can be close to automotive or technology when they discuss digital habits, applications, mobility, or vehicle-related lifestyles. Technology articles produced the highest number of errors because this class often intersects with economy, lifestyle, automotive, and politics contexts.

The Streamlit implementation demonstrates how the best model can be used as an interactive prototype for news topic prediction. Beyond single-article prediction, the application can be extended into a newsroom analytics dashboard that monitors daily topic trends, supports automatic content organization, and assists editorial or digital marketing teams in making data-driven decisions.

Compared with previous studies, the result of this study shows an improvement in classification performance. (Nanda et al., 2022) applied Support Vector Machine for online news classification using 510 news articles and obtained the highest accuracy of 88% with a linear kernel. In this study, SVM achieved a higher accuracy of 98.17% using 1,631 Indonesian online news articles and TF-IDF features. In addition, (Habib et al., 2022) reported that Naive Bayes achieved 94% accuracy in online news classification, while the Naive Bayes model in this study achieved 97.25% accuracy. These comparisons indicate that the combination of a larger dataset, preprocessing stages, TF-IDF feature extraction, and hyperparameter tuning can improve the performance of Indonesian online news topic classification.

CONCLUSION

This study successfully developed and evaluated Indonesian online news topic classification models using Naive Bayes and Support Vector Machine. A total of 1,631 Kompas.com news articles from six topic classes were processed through duplicate checking, text normalization, case folding, tokenizing, filtering, stemming, TF-IDF feature extraction, and stratified train-test splitting. GridSearchCV was used to determine the best model parameters.

The evaluation results show that both models achieved strong classification performance. Naive Bayes obtained 97.25% testing accuracy, while SVM obtained 98.17% testing accuracy. Based on accuracy, macro average, weighted average, and confusion matrix results, SVM was selected as the best model. The remaining classification errors mainly occurred in technology news because its vocabulary overlaps with lifestyle, automotive, and politics topics.

The best model was implemented in a Streamlit-based visualization interface, allowing users to input news text and view the predicted topic along with the prediction score distribution. Future studies can expand the dataset using multiple news portals, compare TF-IDF with contextual embedding methods, and develop the Streamlit prototype into a more complete newsroom business intelligence dashboard.

REFERENCES

- Afdal, M., & Elita, L. R. (2022). PENERAPAN TEXT MINING PADA APLIKASI TOKOPEDIA MENGGUNAKAN ALGORITMA K-NEAREST NEIGHBOR. *Jurnal Ilmiah Rekayasa dan Manajemen Sistem Informasi*, 8(1), 78. <https://doi.org/10.24014/rmsi.v8i1.16595>
- Arifin, N., Enri, U., & Sulistiyowati, N. (2021). Penerapan Algoritma Support Vector Machine (SVM) dengan TF-IDF N-Gram untuk Text Classification. *STRING (Satuan Tulisan Riset dan Inovasi Teknologi)*, 6(2), 129. <https://doi.org/10.30998/string.v6i2.10133>
- Astuti, Y. P., Wibowo, A. R., Kartikadarma, E., Subhiyakto, E. R., Sri Winarsih, N. A., & Rohman, M. S. (2024). Penerapan Metode Naïve Bayes Classifier Untuk Klasifikasi Sentimen Pada Judul Berita. *LogicLink*, 1–12. <https://doi.org/10.28918/logiclink.v1i1.7684>
- Asyhar, E. S., Wijoyo, S. H., & Setiawan, N. Y. (2024). Analisis Sentimen dan Pemodelan Topik Terhadap Ulasan Aplikasi Jenius Menggunakan Metode Support Vector Machine dan Latent Dirichlet Allocation.
- Djiwadikusumah, F., & Irawan, G. H. (2021). *WEB SCRAPING SITUS E-COMMERCE MENGGUNAKAN TEKNIK PARSING DOM*.
- Habib, S. M., Haerani, E., Gusti, S. K., & Ramadhani, S. (2022). Klasifikasi Berita Menggunakan Metode Naïve Bayes Classifier. *Jurnal Nasional Komputasi dan Teknologi Informasi (JNKTI)*, 5(2), 248–258. <https://doi.org/10.32672/jnkti.v5i2.4191>
- Handoko, M. R. (2021). SISTEM PAKAR DIAGNOSA PENYAKIT SELAMA KEHAMILAN MENGGUNAKAN METODE NAIVE BAYES BERBASIS WEB. *Jurnal Teknologi dan Sistem Informasi*, 2(1).
- Hartati, D., Khaira, U., & Bintana, R. R. (2025). Penerapan Support Vector Machine dan Latent Dirichlet Allocation



- dalam Analisis Sentimen Terhadap Pengalaman Pengguna Aplikasi Alfagift. 8(5).*
- Insani, D. F., & Zamzamy, A. (2023). *Analisis Framing Pemberitaan Media Online CNBC Indonesia.com dan Kompas.com Mengenai Dampak Lingkungan Pemindahan Ibu Kota Negara.*
- Joergensen E Munthe, C., Astuti Hasibuan, N., & Hutabarat, H. (2022). Penerapan Algoritma Text Mining Dan TF-RF Dalam Menentukan Promo Produk Pada Marketplace. *Resolusi : Rekayasa Teknik Informatika dan Informasi*, 2(3), 110–115. <https://doi.org/10.30865/resolusi.v2i3.309>
- Mayasari, N., Chairul Rizal, & Fitri Wulandari. (2024). Penerapan Metode Klasifikasi Karakteristik Kepribadian Berbasis Website. *JOURNAL ZETROEM*, 6(2), 27–30. <https://doi.org/10.36526/ztr.v6i2.3755>
- Nanda, R., Haerani, E., Gusti, S. K., & Ramadhani, S. (2022). Klasifikasi Berita Menggunakan Metode Support Vector Machine. *Jurnal Nasional Komputasi dan Teknologi Informasi (JNKTI)*, 5(2), 269–278. <https://doi.org/10.32672/jnkti.v5i2.4193>
- Nur'Faradila, D. A., Magdalena, L., & Febima, M. (2024). Penerapan Naïve Bayes Classifier Untuk Klasifikasi Postingan Berita Hoaks Di Instagram Cirebon Saber Hoaks. *Media Jurnal Informatika*, 16(2), 243. <https://doi.org/10.35194/mji.v16i2.4616>
- Puspitasari, A. D., & Wulandari, E. R. (2024). Proses Produksi Berita dalam Konvergensi Jurnalistik di Website Suarasurabaya.net. *DIGICOM : Jurnal Komunikasi dan Media*, 4(2), 146–151. <https://doi.org/10.37826/digicom.v4i2.815>
- Rianto, Mutiara, A. B., Wibowo, E. P., & Santosa, P. I. (2021). Improving the accuracy of text classification using stemming method, a case of non-formal Indonesian conversation. *Journal of Big Data*, 8(1), 26. <https://doi.org/10.1186/s40537-021-00413-1>
- Solahuddin, M., Purnamasari, A. I., & Dikananda, A. R. (2023). *Klasifikasi Kualitas Berita Pada Majalah Menggunakan Metode Decision Tree. 1(2).*
- Utami, N. W., & Eka Putra, I. G. J. (2022). TEXT MINIG CLUSTERING UNTUK PENGELOMPOKAN TOPIK DOKUMEN PENELITIAN MENGGUNAKAN ALGORITMA K-MEANS DENGAN COSINE SIMILARITY. *Jurnal Informatika Teknologi dan Sains*, 4(3), 255–259. <https://doi.org/10.51401/jinteks.v4i3.1907>