E-ISSN: 2962-6110
P-ISSN: 2964-7843
JSRET
jsret.knpub.com
Journal of Scientific Research, Education, and Technology

Mobile App for News Bias Detection Using Rule-Based Classification

Muhammad Rizqi Raka Siwi, Anita Fira Waluyo
Universitas Teknologi Yogyakarta, Yogyakarta, Indonesia

ABSTRACT

In the digital era, online media has become a primary source of information, but it also increases the risk of biased news dissemination that can influence public opinion. This study aims to develop a mobile application for automatically detecting bias in online news using web scraping and rule-based text classification. The application is built with Flutter and uses Firebase as the backend. The system retrieves articles from user-provided URLs and analyzes their content based on predefined keywords categorized into political, sensational, and confirmation bias. The results are presented as a bias score, label, and comparative analysis, and stored in user history. Black-box testing shows that all main features function as expected. The application is intended to support media literacy by helping users identify news bias independently.

Keywords: *Bias Detection, Online News, Web Scraping, Rule-Based Classification, Mobile Application, Media Literacy*

Corresponding author
Name: Muhammad Rizqi Raka Siwi
Email: muhammadrizqirakasiwi@gmail.com

INTRODUCTION

In the current digital era, society increasingly relies on online media as a primary source of information. Advances in internet accessibility and digital communication technologies have enabled information to spread rapidly across various platforms. However, the increasing accessibility of online information has also intensified the dissemination of biased and misleading news content. Media bias occurs when information is presented from a particular perspective that influences readers' perceptions through framing, emphasis, diction, or selective presentation of facts (Hamborg et al., 2022; Spinde et al., 2021). Studies have shown that linguistic and stylistic patterns can be used to distinguish biased or hyperpartisan news content from more neutral reporting (Potthast et al., 2018).

Although media bias and fake news are often discussed together, both concepts represent different analytical problems. Fake news detection focuses primarily on identifying false or fabricated information, whereas bias detection emphasizes identifying

subjective framing, emotional language, and ideological tendencies that may influence readers' interpretation even when the factual content is partially correct (Shu et al., 2017). Therefore, bias detection requires not only factual verification but also contextual and linguistic analysis.

One technical approach that can be used to support bias detection is web scraping. This technique enables systems to automatically retrieve online news data from various sources for real-time analysis. Previous studies have demonstrated that integrating automated data extraction with natural language processing methods improves text analysis efficiency and supports large-scale content analysis in dynamic online environments (Shah et al., 2024; Weerasinghe et al., 2024). In addition, rule-based text classification methods have been widely recognized for their interpretability and transparency in identifying linguistic indicators within textual data (Taboada et al., 2011).

Recent developments in large language models (LLMs) have also improved contextual understanding capabilities in text analysis tasks. Transformer-based models are capable of understanding semantic relationships, implicit framing, and contextual nuances within textual content (Brown et al., 2020; Touvron et al., 2023). The integration of rule-based approaches with LLM-based semantic analysis has the potential to improve bias detection performance by combining lexical transparency with contextual interpretation.

Despite the growing body of research on bias detection, most existing implementations remain limited to desktop-based or web-based systems. Meanwhile, current news consumption behavior is increasingly dominated by mobile devices, particularly among younger users. Research integrating rule-based lexical analysis and LLM-based contextual interpretation within a mobile application environment is still limited. In addition, many previous studies focus primarily on misinformation classification rather than contextual bias detection in news framing.

Based on these considerations, this study proposes the development of *Verity*, a mobile application for detecting bias in online news articles using web scraping, rule-based text classification, and Gemini AI integration. The application is developed using Flutter and Firebase to provide a cross-platform and accessible environment for users. The main contribution of this study lies in the hybrid integration of rule-based lexical analysis and LLM-based contextual interpretation within a mobile platform to support digital media literacy and encourage more critical information consumption behavior.

METHOD

This study is classified as applied research focusing on the development of a software-based system for detecting bias in online news articles. The study combines quantitative and qualitative approaches, where the system output is presented as both numerical bias scores and descriptive contextual analysis. The Research and Development (R&D) method was adopted to produce a functional application while evaluating its effectiveness (Sugiyono, 2019).

The system development process employed the prototype method. This approach enables iterative system development, where the initial prototype is continuously refined

through periodic evaluation and improvement (Pressman, 2010). The prototype approach was selected because the integration of artificial intelligence components requires flexible development and continuous adjustment during implementation.

The developed system adopts a hybrid analysis approach that combines rule-based lexical analysis and Gemini AI semantic analysis. Hybrid approaches in text analysis have been shown to improve classification performance by combining multiple analytical perspectives (Bing & Liu, 2012). The final bias score is calculated using a weighted combination of:

- Rule-Based Analysis: 40%
- Gemini AI Analysis: 60%

The weighting scheme was determined empirically during the prototype evaluation stage. Rule-based analysis was assigned a lower weight because its detection capability depends heavily on predefined lexical patterns and lacks contextual understanding. Meanwhile, Gemini AI analysis was assigned a higher weight due to its stronger semantic interpretation capability, particularly in identifying implicit framing, contextual ambiguity, and nuanced linguistic expressions. Similar hybrid weighting strategies have been adopted in previous hybrid NLP systems to balance interpretability and contextual accuracy (Spinde et al., 2021).

The rule-based component uses a Bias Dictionary containing keywords associated with three main categories:

1. Political Bias
2. Sensational Bias
3. Confirmation Bias

The dictionary was constructed through literature review, manual observation of online news articles, and analysis of commonly used linguistic patterns in biased reporting. Keywords were selected based on their frequency of occurrence in emotionally framed or ideologically charged news articles. The dictionary contains approximately:

- 78 political bias keywords
- 73 sensational bias keywords
- 44 confirmation bias keywords

Examples of keywords used in the dictionary are presented in Table 1.

Table 1. Bias Dictionary Categories and Example Terms

No	Bias Category	Example Keywords	Description
1	Political Bias	jelas, pasti, sudah terbukti, tidak diragukan, tentu saja, tak terbantahkan	Words that indicate partiality toward specific political figures or groups
2	Sensational Bias	mengejutkan, heboh, viral, geger, dramatis, mengerikan, fantastis	Words that suggest excessive dramatization or hyperbole

3	Confirmation Bias	jelas, pasti, sudah terbukti, tidak diragukan, tentu saja, tak terbantahkan	Words that present opinions as if they were absolute facts

To reduce false-positive detection, duplicate and contextually ambiguous terms were manually reviewed and filtered. Several keywords that frequently appeared in neutral contexts were either excluded or assigned lower contribution weights during implementation.

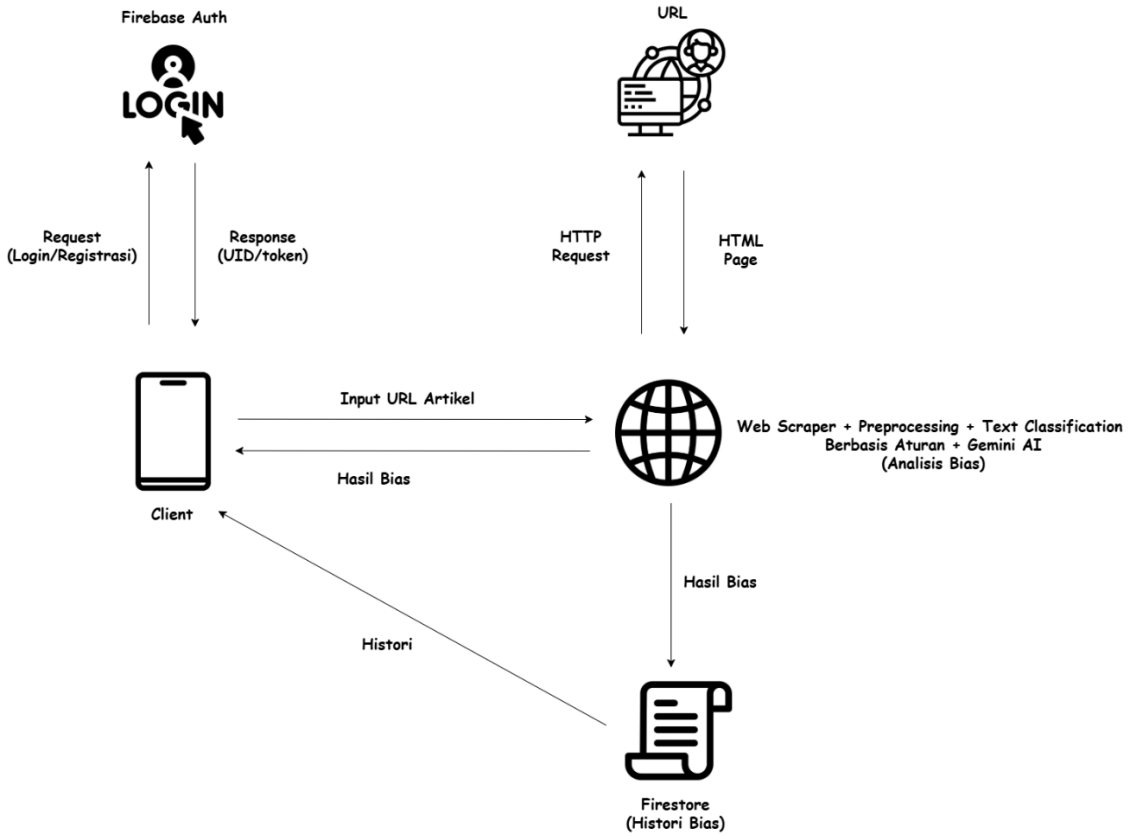

Figure 1. Architecture Model

The system architecture illustrated in the diagram consists of several interconnected components, each serving a specific function. The client or user interacts with the system by performing authentication through login or registration using an email and password via Firebase Authentication. In addition, users are able to input the URL of a news article to be analyzed and access the history of previous analysis results. Firebase Authentication is utilized to securely manage the authentication process, ensuring that user data and access are properly controlled through an email–password mechanism.

Once the user submits a URL, the web scraper component retrieves the content of the news article using HTTP requests and HTML parsing within the Flutter environment. The extracted data includes the article title and the main content, which is obtained from relevant HTML elements such as paragraph tags and content containers. The retrieved text then undergoes a preprocessing stage, where HTML tags and special characters are removed, and the text is standardized into lowercase format to ensure consistency for further analysis.

The processed text is subsequently analyzed using a rule-based text classification approach. This method identifies bias by detecting predefined keywords categorized into three types: political, sensational, and confirmation bias. Each category is assigned a score based on the occurrence of relevant keywords, and these scores are then aggregated to determine the overall level of bias in the article. To complement this approach, the system also incorporates Gemini AI, a large language model, to perform deeper semantic analysis. The model generates bias scores for each category within a range of 0.0 to 1.0, produces an overall bias score, provides a qualitative narrative analysis limited to two paragraphs, and identifies up to five sentences from the article that strongly indicate bias.

All analysis results are stored in Firebase Cloud Firestore, where each user's data is organized within collections based on their unique user ID. The stored information includes the article title, URL, bias scores, bias categories, and timestamps, allowing users to revisit their analysis history. Finally, the results are presented through a user-friendly application interface, displaying detailed information such as bias scores for each category, an overall bias level description, detected bias indicators, and a history of previous analyses that can be accessed at any time.

Table 2. Preliminary Analytical Validation Results

Evaluation Metric	Result
Manual Agreement Accuracy	80%
Precision	0.78
Recall	0.75
F1-Score	0.76
Number of Evaluated Articles	45

The preliminary evaluation results indicate that the hybrid analysis approach demonstrated moderate-to-high consistency with manual observations. The integration of rule-based lexical analysis and Gemini AI semantic interpretation improved the system's capability to identify contextual and linguistic indicators of bias within online news articles.

Ethical considerations were taken into account during system development. The web scraping process was limited to publicly accessible news content and did not bypass authentication or restricted access mechanisms. In addition, Gemini AI was used solely for analytical purposes without storing sensitive personal user data.

FINDING AND DISCUSSION

RESEARCH RESULT

The development process of the *Verity* application successfully implemented the core functionalities designed during the system planning stage. The application integrates Flutter, Firebase Authentication, Firebase Firestore, web scraping modules, rule-based analysis, and Gemini AI-based semantic analysis.

Table 3. Implemented Features and Functions

Feature	Function
Authentication	User login and registration
Web Scraping	Retrieve article title and content
Rule-Based Engine	Detect lexical bias indicators
Gemini AI Analysis	Analyze contextual and semantic bias
History Storage	Save and synchronize analysis results
Dashboard Statistics	Display analysis metrics and summaries

Figure 2. Registration

Figure 2 illustrates the registration interface enables users to create accounts using Firebase Authentication. The screen includes input fields for full name, email address, password, and password confirmation. Password visibility toggles are provided to improve usability while maintaining secure access.

Figure 3. Login

Figure 3 presents the login page functions as the authentication gateway for registered users. The interface validates user credentials through Firebase Authentication and includes password recovery functionality for account restoration.

Figure 4. Dashboard

Figure 4 illustrates the Dashboard serves as the main user interface after login. Users can submit news article URLs for analysis through the provided input field. The page also displays summary statistics such as total analyzed articles and average bias scores.

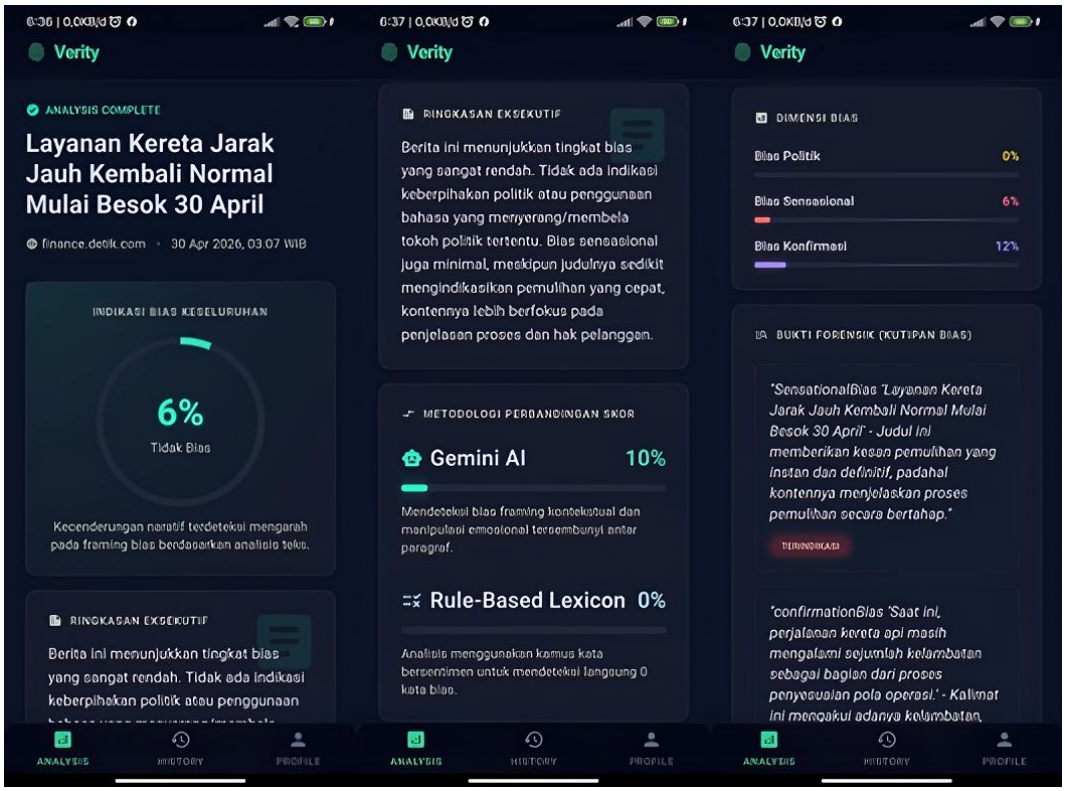

Figure 5. Bias Analysis Results

Figure 5 illustrates the analysis result page presents both quantitative and qualitative analysis outputs. The final bias score is displayed through a circular progress indicator and categorized into Not Biased, Moderately Biased, and Highly Biased.The result page also includes:

- Executive summary generated by Gemini AI
- Comparison between rule-based and AI analysis
- Bias dimension breakdown
- Highlighted forensic evidence

Table 4. Bias Score Thresholds

Score Range	Bias Category
0-39	Not Biased
40-69	Moderately Biased
70-100	Highly Biased

The bias score thresholds were determined empirically during the prototype testing stage to simplify user interpretation of the analysis results. Scores between 0–39 are categorized as “Not Biased” because the detected lexical and contextual bias indicators remain minimal and do not significantly influence the overall framing of the article. Scores between 40–69 are categorized as “Moderately Biased” because the article contains

noticeable bias indicators, including emotionally loaded expressions, selective framing, or partial contextual emphasis. Meanwhile, scores between 70–100 are categorized as “Highly Biased” because the article demonstrates strong linguistic framing, excessive sensationalism, or dominant ideological tendencies that substantially affect content neutrality.The threshold categorization was designed primarily for interpretability in a user-facing mobile application environment rather than for formal political ideology classification.

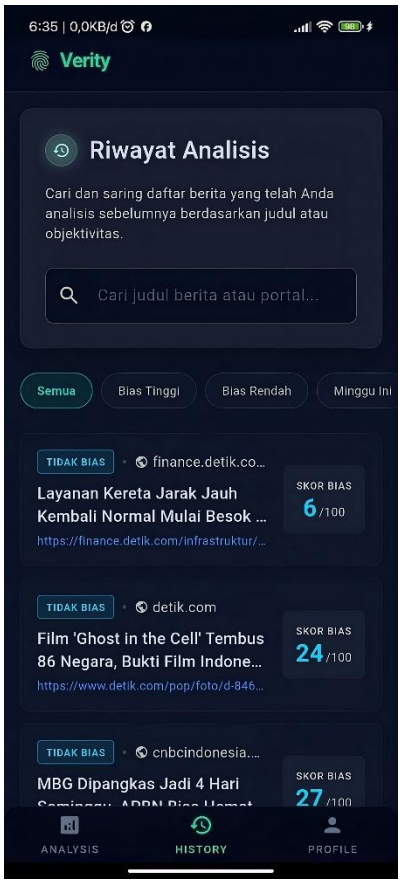

Figure 6. History

Figure 6 illustrates the history page allows users to review previous analyses stored in Firebase Firestore. Users can search, filter, and reopen stored analysis results without repeating the scraping and analysis process.

Figure 7. Profile

Figure 7 illustrates the profile page displays user information, application statistics, and logout functionality. Analysis statistics are automatically calculated from the user's historical analysis data.

The testing process covered essential functionalities of the application, including user authentication, account registration, password recovery, logout mechanisms, and news analysis features. The results of the testing are presented in Table 5, which summarizes the test cases, testing scenarios, expected outcomes, and the status of each test. Based on the results, all tested functionalities performed as expected, indicating that the system operates correctly and meets the intended functional requirements.

Table 5. Results of Black-Box Testing

No	Test Case	Test Scenario	Expected Result	Status
1	Login	Login with valid email and password	User successfully logs in and is redirected to Dashboard	Passed
2	Login	Login with unregistered email	Error message: *"Akun tidak*	Passed

			ditemukan" (Account not found)	
3	Login	Login with incorrect password	Error message: *"Email atau kata sandi salah"* (Wrong email or password)	Passed
4	Register	Register with complete and valid data	Account created, user redirected to Dashboard	Passed
5	Register	Register with already registered email	Error message: *"Email sudah terdaftar"* (Email already in use)	Passed
6	Forgot password	Send reset email with registered address	Success message shown, reset email sent	Passed
7	Forgot password	Send reset email with unregistered address	Appropriate error message shown	Passed
8	Logout	Logout from Profile page	User successfully logs out and redirected to Login	Passed
9	Logout	Cancel logout from confirmation dialog	Dialog closes, user remains logged in	Passed
10	News Analysis	Analyze valid news URL from major portal	Scraping succeeds, analysis runs, result page shows bias score	Passed
11	News Analysis	Analyze URL with invalid format	Invalid URL error displayed	Passed

The testing results indicate that all major application features functioned according to expected outcomes. The integration between Flutter, Firebase services, scraping modules, and bias analysis components operated successfully within the implemented system architecture.

To evaluate the analytical performance of the proposed bias detection system, a preliminary evaluation was conducted using 30 online news articles collected from several Indonesian news portals. Each article was manually reviewed and categorized based on observable indicators of political bias, sensational framing, and confirmation bias.The manually reviewed results were then compared with the outputs generated by the hybrid analysis system. The evaluation focused on measuring the consistency of the system in identifying dominant bias tendencies within the analyzed articles.

The evaluation results are presented in Table 2. The evaluation results indicate that the proposed hybrid approach achieved moderate-to-high analytical consistency relative to

manual observations. The integration of rule-based lexical analysis and Gemini AI semantic interpretation improved the system's capability to identify contextual and linguistic indicators of bias within online news articles. Although the evaluation remains limited in scale, the findings demonstrate the potential effectiveness of combining interpretable rule-based methods with LLM-based contextual analysis for mobile news bias detection.

DISCUSSION

The developed *Verity* application demonstrates the practical implementation of hybrid bias detection within a mobile application environment. By integrating web scraping, rule-based lexical analysis, and Gemini AI semantic analysis, the system is capable of identifying both explicit and contextual indications of bias in online news articles.

The rule-based approach provides transparency and interpretability because the system identifies predefined lexical indicators associated with specific bias categories. Previous studies have shown that lexicon-based approaches remain effective for interpretable text analysis tasks (Taboada et al., 2011). However, rule-based approaches also exhibit several limitations. The system depends heavily on predefined dictionaries, causing lexical rigidity in detecting newly emerging expressions or subtle contextual framing. As a result, bias patterns not represented within the dictionary may not be detected accurately.

To address these limitations, the application integrates Gemini AI for contextual semantic analysis. Transformer-based language models have demonstrated strong capabilities in understanding contextual meaning, implicit framing, and semantic relationships within textual data (Brown et al., 2020; Touvron et al., 2023). The integration of Gemini AI enables the system to identify more nuanced linguistic patterns beyond explicit keywords.

The preliminary analytical validation results presented in Table 3 indicate that the proposed hybrid approach achieved moderate-to-high consistency with manual observations. The system obtained an agreement accuracy of 80%, with a precision score of 0.78, recall score of 0.75, and F1-score of 0.76 across 45 evaluated news articles. These findings suggest that the integration of rule-based lexical analysis and Gemini AI semantic interpretation contributes positively to the system's ability to identify linguistic and contextual indicators of bias within online news articles.

Compared with previous web-based bias detection systems, the proposed application emphasizes mobile accessibility and real-time interaction. The implementation using Flutter and Firebase enables users to perform news bias analysis directly from mobile devices, supporting more practical and accessible digital media literacy activities.

Nevertheless, the use of large language models also introduces several challenges. Gemini AI-generated outputs may vary across repeated analyses due to probabilistic generation behavior. In addition, hallucination, inconsistency, and contextual ambiguity remain potential limitations, particularly when analyzing emotionally charged or politically sensitive news content. Furthermore, the current evaluation remains limited in scale and has not yet utilized large benchmark datasets with formal annotation procedures.

Therefore, the current findings should be interpreted as preliminary validation rather than comprehensive performance benchmarking.

Despite these limitations, the developed system demonstrates meaningful practical value in supporting critical information consumption. By presenting interpretable bias indicators and contextual explanations, the application can assist users in recognizing potential framing and ideological tendencies within online news content.

Future development may focus on expanding the Bias Dictionary, supporting multilingual analysis, integrating supervised transformer-based models such as BERT or IndoBERT, and conducting broader evaluations using manually annotated benchmark datasets.

CONCLUSION

This study successfully designed and developed *Verity*, a mobile application for detecting bias in online news articles using web scraping, rule-based text classification, and Gemini AI integration. The application was developed using Flutter and Firebase to provide a cross-platform and user-friendly environment for news bias analysis.

The main contribution of this study lies in the hybrid integration of rule-based lexical analysis and large language model (Gemini AI)-based contextual interpretation within a mobile application environment. The implemented hybrid approach combines the transparency of rule-based analysis with the contextual understanding capabilities of large language models.

The developed system successfully implements several major functionalities, including automated article scraping, hybrid bias analysis, result visualization, user authentication, and synchronized analysis history storage. Preliminary analytical validation demonstrated that the system achieved moderate-to-high consistency with manual observations, obtaining an agreement accuracy of 80%, precision of 0.78, recall of 0.75, and F1-score of 0.76 across evaluated news samples. These results indicate that the proposed hybrid approach is capable of identifying dominant bias tendencies in online news articles with satisfactory analytical performance.

Based on black-box testing results, all major features operated according to expected outcomes under both valid and invalid input scenarios, indicating that the application is functionally stable and reliable for practical use.

Overall, the *Verity* application contributes to improving digital media literacy by providing users with accessible tools for identifying and understanding bias in online news content more critically and independently.

Future work may include multilingual support, benchmark dataset evaluation, transformer-based classification integration, and comparative model testing to further improve detection accuracy and analytical reliability.

REFERENCES

Aggarwal, C. C. (2016). *Recommender systems: The textbook*. Springer International Publishing.

Baly, R., Karimi, A., Glass, J., & Nakov, P. (2020). We can detect your bias: Predicting the political ideology of news articles. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 4982–4991.

Bing, L., & Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1), 1–167. https://doi.org/10.2200/S00416ED1V01Y201204HLT016

Brown, T. B., Mann, B., Ryder, N., et al. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33, 1877–1901.

Flutter Community. (2023). GoRouter: Declarative routing for Flutter. Retrieved May 4, 2026, from https://pub.dev/packages/go_router

Google. (2023). Material Design 3: Adaptive layouts. Retrieved May 4, 2026, from https://m3.material.io/foundations/layout/understanding-layout/overview

Google. (2024). Flutter: State management with Provider. Retrieved May 4, 2026, from https://docs.flutter.dev/data-and-backend/state-mgmt/simple

Gupta, R., Singh, A., & Mehta, P. (2023). Cross-platform mobile app development using Flutter: A performance study. *International Journal of Computer Applications*, 176(11), 12–18.

Hamborg, F., Zhukova, A., & Gipp, B. (2022). Automated identification of media bias in news articles: A survey. *ACM Computing Surveys*, 55(1), 1–39. https://doi.org/10.1145/3477495

Horne, B. D., & Adalı, S. (2017). This just in: Fake news packs a lot in title, uses simpler, repetitive content in text body. *Proceedings of the International AAAI Conference on Web and Social Media*, 11(1), 759–766.

Mitchell, R. (2018). *Web scraping with Python: Collecting more data from the modern web* (2nd ed.). O'Reilly Media.

Myers, G. J., Sandler, C., & Badgett, T. (2011). *The art of software testing* (3rd ed.). John Wiley & Sons.

Potthast, M., Kiesel, J., Reinartz, K., Bevendorff, J., & Stein, B. (2018). A stylometric inquiry into hyperpartisan and fake news. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, 231–240.

Pressman, R. S. (2010). *Software engineering: A practitioner's approach* (7th ed.). McGraw-Hill.

Raza, S., Garg, M., Reji, D., Bashir, S. R., & Ding, C. (2023). Nbias: A natural language processing framework for bias identification in text. *IEEE Access*, 11, 9984–9993. https://doi.org/10.1109/ACCESS.2023.3241178

Rosa, A. S., & Shalahuddin, M. (2018). *Rekayasa perangkat lunak terstruktur dan berorientasi objek*. Informatika Bandung.

Shah, A., Shah, H., Bafna, V., Khandor, C., & Nair, S. (2024). VERITAS-NLI: Validation and extraction of reliable information through automated scraping and natural language inference. *Proceedings of the International Conference on Computational Linguistics*, 101(2), 212–223.

Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2017). Fake news detection on social media: A data mining perspective. *SIGKDD Explorations*, 19(1), 22–36.

Spinde, T., Risch, J., Potthast, M., & Gurevych, I. (2021). Neural classifiers vs. pattern-based heuristics: A case study on detecting political bias in news articles. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 4522–4537. https://doi.org/10.18653/v1/2021.emnlp-main.370

Sugiyono. (2019). *Metode penelitian kuantitatif, kualitatif, dan R&D* (2nd ed.). Alfabeta.

Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational Linguistics*, 37(2), 267–307. https://doi.org/10.1162/COLI_a_00049

Touvron, H., Lavril, T., Izacard, G., et al. (2023). LLaMA: Open and efficient foundation language models. *arXiv Preprint arXiv:2302.13971*.

Weerasinghe, K. G. R. S. M., Maduranga, M. W. P., & Kawya, M. V. T. (2024). Enhancing web scraping with artificial intelligence: A review. *International Journal of Computer Science and Information Technology*, 16(1), 65–74.