Journal of World Science
Vol. 5, No. 9, Month 2026, pp 907-924
E-ISSN: 2828-9307
https://doi.org/10.58344/jws.v5i8.1702

Comparison of LLM Prompting Strategies for Identifying Indonesian-Language Hoax Narratives

Silvie Zahwa Ardiansyah
Sekolah Tinggi Manajemen Informatika dan Komputer IKMI Cirebon, Indonesia
Corresponding Author: zahwaardiansyah04@gmail.com

Keywords:
large language model; hoax detection; zero-shot prompting; few-shot prompting; text classification; Indonesian language

This is an open access article under the CC BY-SA license.

Copyright © 2026 by Author. Published by Riviera Publishing

ABSTRACT

Large language models (LLMs) can classify text through instructions alone, but their output may depend on the prompting strategy, and the effect of adding labelled examples on the identification of Indonesian-language hoax narratives has rarely been tested empirically. This study aimed to (1) compare the initial results of zero-shot and few-shot prompting with 2, 4, and 8 examples, (2) examine the relationship between the number of examples and latency, and (3) formulate a validation protocol for trustworthy HOAX/NON-HOAX evaluation. A comparative quantitative experiment applied the four configurations to a single LLM accessed through an API, using a dataset of 4,000 narratives (1,000 HOAX and 3,000 NON-HOAX). Recorded accuracy, macro-F1, and mean latency were audited analytically through single-class metric identities, 95% Wilson intervals, and a least-squares latency trend. The recorded test set contained only 400 HOAX items, and every recorded macro-F1 equalled A/(1 + A); the recorded accuracies (86.75%, 65.75%, 70.50%, and 52.00%) therefore represent hoax recall rather than binary performance. Hoax recall was highest for zero-shot, and its gap from every few-shot configuration exceeded sampling uncertainty. Mean latency rose almost linearly from 0.418 to 0.923 seconds, about 0.065 seconds per added example, reaching 2.21 times the zero-shot value. Adding examples increased computational cost without improving hoax recall, whereas binary classification performance remains undetermined. Because the model identity and few-shot examples were not preserved, the proposed protocol stratified splitting, leakage-free example selection, per-class metrics, and complete model documentation must be satisfied before any configuration is declared superior.

INTRODUCTION

The rapid flow of digital information has widened public access to news, opinions, and narratives from diverse sources. The same ecosystem, however, makes it easier for false or misleading information to spread. Hoaxes can shape public perception, amplify uncertainty, and hinder evidence-based decision-making (Aïmeur et al., 2023). Automated detection can help filter narratives that require further review, but the output of a computational system should not be treated as a substitute for fact-checking.

Conceptually, hoax detection is a text classification task that depends on the relationship between narrative content, context, reference labels, and the characteristics of the data distribution. In this study, HOAX and NON-HOAX are treated as operational labels derived from the dataset curation process, not as a claim that a model can establish the truth of

information from text alone. Verifying truth ideally requires external evidence, temporal context, claim identification, and source checking (Guo et al., 2022).

This distinction matters because text-based classifiers can learn statistical correlations that are not identical to factual evidence. Emotional words, headline patterns, text length, publisher names, and phrases common in fact-checking articles can all correlate with the label. If such features are not controlled, evaluation scores may reflect surface-pattern recognition or shortcut learning rather than an ability to assess claims semantically (Du et al., 2024).

The urgency of this problem lies in the gap between the convenience of instruction-based classification and the rigour required to evaluate it. Because an LLM can be applied without task-specific training, it is tempting to adopt a prompting configuration on the basis of a single favourable score. Yet, as noted above, text-based classifiers may rely on surface cues, and evaluation scores obtained under an unsuitable test design can overstate what a model actually does. For a socially sensitive task such as hoax identification, where misclassification can either let false narratives pass or wrongly flag legitimate news, any claim about the superiority of a prompting strategy must therefore rest on a valid and transparent evaluation protocol.

Classical approaches to fake news detection rely on lexical features, TF-IDF representations, and machine learning algorithms, while deep learning approaches learn text representations directly. Research on Indonesian news has tested a range of approaches, including SVM-based classification and one-dimensional convolutional models. Al Amin et al. (2025), for example, examined Indonesian hoax news detection with a 1D-CNN, whereas Safira and Nurlayli (2023) compared several machine learning algorithms for predicting news validity. Yefferson et al., (2024) combined IndoBERT as a feature extractor with an LSTM classification layer to detect Indonesian hoax news. These studies show that dataset characteristics, label definitions, and evaluation procedures are an essential part of any comparison between methods.

The development of large language models (LLMs) offers an alternative in the form of instruction-based classification. A model can be asked to assign a label to a text without being retrained on a task-specific dataset (P. Liu et al., 2023). This capability is not always stable. B. B. Hu et al., (2024) found that GPT-3.5 could provide useful multi-perspective rationales for fake news detection yet still underperformed a fine-tuned BERT model, suggesting that LLMs may serve better as advisors to smaller task-specific models than as their replacements. Pendyala and Hall (2024) examined the use of LLMs for detecting and explaining misinformation, and Papageorgiou et al. (2024) surveyed the use of LLMs in fake news research, highlighting both the opportunities and the challenges of the approach. Chen and Shu (2024) likewise argued that LLMs can help counter misinformation but also create new challenges related to model-generated false information.

One factor that researchers can control is the prompting strategy. In zero-shot prompting, the model receives instructions without labelled examples. In few-shot prompting, several text–label pairs are added as examples. Adding examples may clarify the expected output pattern, but it also lengthens the prompt context, increases processing time, and can introduce bias when examples are poorly chosen. Jin et al. (2025) developed a prompting approach that enriches context and adds veracity constraints, and Haupt et al. (2024) showed that role-based

changes to instructions can alter misinformation detection accuracy. Prompting strategies should therefore be tested empirically rather than assumed to improve performance.

Prompting offers a way to steer a model without fine-tuning. In zero-shot prompting, the model relies on the instruction and on knowledge acquired during pretraining. In few-shot prompting, labelled examples become additional context that can reduce ambiguity about the output format and the class definitions. Examples can also introduce order effects, class imbalance, excessive similarity to test data, and selection bias (Dong et al., 2024). The number of examples should therefore be treated as an experimental variable while the remaining content of the instruction is held constant.

Jin et al. (2025) proposed CAPE-FND, which combines context, veracity-oriented constraints, and prompt optimisation; the approach shows that the quality of instructions and context is part of the design of fake news detection. Roslan and Mohd (2025) compared zero-shot with two-shot prompting on GPT-3.5-Turbo and reported small, non-significant improvements on the two datasets they tested.

This finding is a reminder that adding examples does not always yield large gains and that the effect depends on the dataset and the model.Haupt et al. (2024) showed that adding role-playing instructions can substantially change misinformation detection results in the context of their study. Hu and Tian (2025) evaluated reasoning models on rumour tasks and reported that traditional models can still outperform LLMs in several detection scenarios. Together, these studies support task- and dataset-specific evaluation and caution against assuming that larger models or longer prompts are necessarily better.

Taken together, the studies reviewed above show that prompting choices can alter detection outcomes, but they either change several prompt components at once, are based on data in other languages, or rely on supervised training. Among the studies reviewed, none isolates the number of in-context examples as the single manipulated variable on Indonesian-language narratives while holding the model and instruction constant. This gap defines the novelty of the present study.

This study focuses on comparing four configurations zero-shot and few-shot with 2, 4, and 8 examples using one and the same LLM. This focus differs from studies that compare many model architectures, because the main variable observed is the number of examples in the prompt. The evaluation was designed to consider both classification quality and time efficiency, so that a configuration is judged not only by the proportion of correct predictions but also by its behaviour on each class and by its latency. The position of this study is a controlled test of one LLM at four example levels 0, 2, 4, and 8 on Indonesian-language narratives. The expected contribution is not a universal claim about all LLMs but findings bound to the documented model, data, prompt, and evaluation protocol. Measuring macro-F1, per-class metrics, and latency allows a fuller discussion than accuracy alone.

The objectives of the study are to compare the initial results of four prompting configurations, to examine the relationship between configuration and latency, and to formulate a validation protocol that makes HOAX/NON-HOAX evaluation trustworthy. At this stage the contribution is methodological and exploratory: the article documents the initial results transparently, shows analytically what the recorded metrics actually measure, and identifies the test-set composition problems that must be resolved before any performance conclusion is drawn.

METHOD

Research Design

The study was designed as a comparative quantitative experiment in which four prompting configurations were applied to a single LLM and compared on the same recorded test set. Using one model reduces variation arising from differences in architecture or capability, so that the prompting strategy is the only manipulated factor. The initial run was executed through an API, and the latency of each request was recorded. The model name and version, inference parameters, execution dates, and API settings were not preserved in the source files of the initial run (Table 7); the results are therefore reported as a single preliminary run, and these items are specified as mandatory reporting elements of the validation protocol formulated in this study.

Dataset and Labels

The dataset consists of 4,000 Indonesian-language narratives, comprising 1,000 HOAX and 3,000 NON-HOAX items (Figure 1) [dataset source curator, year, repository/URL, and licence to be inserted by the author]. The binary label was constructed during data preparation by combining the hoax and label columns of the source data into the label strings HOAKS and NON-HOAKS, reported here as HOAX and NON-HOAX. The labels therefore follow the operational assignment of the dataset curators and were not re-annotated in this study. Checks for empty values, duplicates, very short texts, and near-identical articles across subsets were not documented for the initial run; they are therefore included as step (1) of the validation protocol (Figure 2).

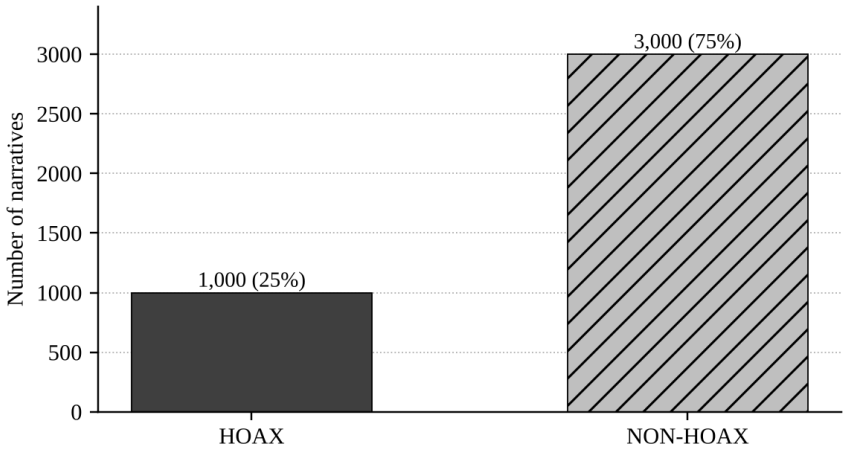


Figure 1. Distribution of the initial dataset by label (N = 4,000)
Source: Initial dataset; processed by the author (2026)

The composition of data sources is a limitation that requires attention. If a given source contributes only one class, a model can learn its stylistic patterns, article format, or characteristic vocabulary. Where possible, dditional analyses should control for source or test on data from sources that do not appear in the development data. Without such control, classification results may overstate the generalisation ability of the model. With a 1:3 class ratio, a trivial classifier that always predicts NON-HOAX would already reach 75% accuracy on a stratified test set, which is why accuracy alone cannot serve as the primary metric.

Data Splitting and Leakage Prevention

The procedure used to derive the test set in the initial run was not documented in the experiment files. The audit established that the recorded test set comprised 400 rows, all carrying the actual label HOAX (see Results and Discussion). To correct this, the validation protocol specifies a stratified 80:20 split of the 4,000 items, yielding 3,200 development and 800 test items, of which 200 are HOAX and 600 NON-HOAX if all data are retained (Figure 6). This split is part of the protocol for the re-run and was not applied in the initial run.

In the initial run, the few-shot configurations added 2, 4, or 8 labelled narrative–label pairs to the base instruction; however, the identity of these examples, their class balance, and their order were not archived (Table 7). The validation protocol therefore requires that examples be drawn exclusively from the development set, balanced by class, recorded by identifier and order, and kept fixed across runs. The test set is reserved for final evaluation and is not used to select examples, revise instructions, or adjust output-parsing rules, so that all four configurations share an identical test set, identical actual labels, and identical prediction-processing rules.

Prompting Configurations

The four configurations are summarised in Table 1. The base instruction was kept constant, and the number of examples was the manipulated variable. Because the exact prompt text of the initial run was only partly documented (Table 7), Table 2 sets out the prompt structure fixed in the validation protocol, in which the model is instructed to return exactly one label, HOAX or NON-HOAX, to reduce ambiguity during output parsing, and only the labelled examples change between configurations.

Table 1. Prompting configurations compared

Configuration	Number of examples	Role in the experiment
Zero-shot	0	Baseline condition without labelled examples
Few-shot 2	2	Measures the configuration with two examples
Few-shot 4	4	Measures the configuration with four examples
Few-shot 8	8	Measures the configuration with eight examples

Source: Research design (2026)

Table 2. Prompt structure held constant across configurations in the re-run

Prompt component	Content held constant across configurations	Varied?
Task instruction	Classify the Indonesian narrative as HOAX or NON-HOAX	No
Label definition	Operational definitions of both labels taken from the dataset curation rules	No
Output constraint	Return exactly one label, with no explanation	No
Labelled examples	0, 2, 4, or 8 narrative–label pairs drawn from the development set	Yes
Example balance and order	Equal number per class; order fixed and recorded	No
Target narrative	One test narrative per request	No

Source: Author's protocol (2026)

Experimental Procedure

The study was carried out in two stages. In the first stage (initial run), the four configurations were executed on the recorded test set, and the actual labels of the test items, the summary accuracy and macro-F1 values, and the mean latency per configuration were stored in the experiment files. In the second stage (audit), the recorded results were examined analytically, as described under Audit Procedure. On the basis of the audit, a seven-step validation protocol was formulated (Figure 2): (1) check the label distribution, duplicates, empty values, and the consistency of actual labels; (2) split the data, in a documented way, into a development set and a test set containing both classes; (3) fix the instruction, output format, model, and inference parameters;

(4) select few-shot examples from the development data for the 2-, 4-, and 8-example configurations; (5) run all four configurations on the identical test set and store per-sample predictions; (6) record the latency of every request with a consistent measurement procedure; and (7) re-check the number of rows, the distribution of actual labels, and the predicted labels before computing metrics. Step (7) acts as a validation gate: if it fails, the split is repaired and all configurations are re-run.

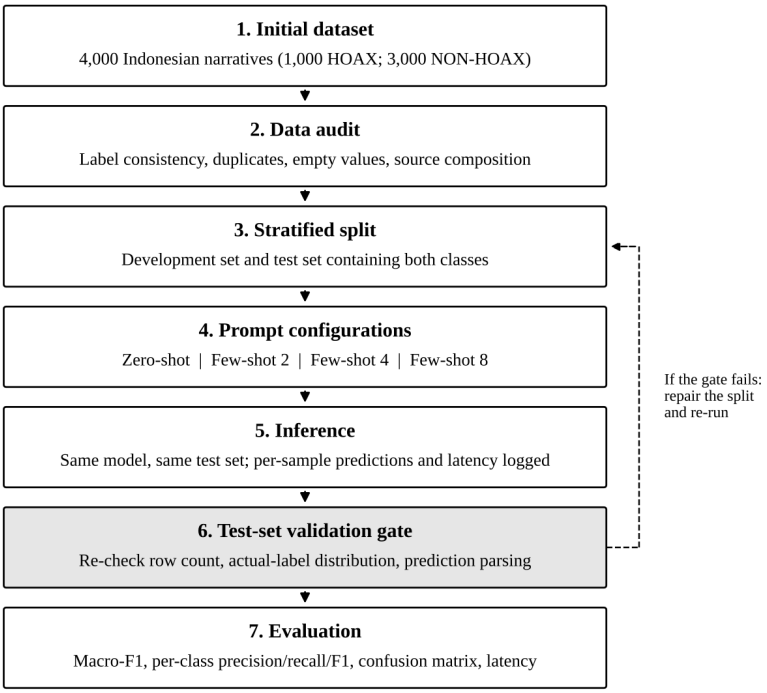


Figure 2. Proposed experimental workflow with a test-set validation gate
Source: Author's design (2026)

Evaluation Metrics

The primary planned metric is macro-F1, because it averages the F1-score of each class with equal weight. Additional metrics are accuracy, per-class precision, recall, and F1-score, the confusion matrix, and mean latency. Accuracy measures the proportion of correct predictions, while precision and recall give different information about false positives and about the ability to find instances of a given class. Per-class metrics are essential when the data are imbalanced. For a class c with true positives TP, false positives FP, and false negatives FN, and for N test items, the metrics are defined as follows.

$$Accuracy = (TP_H + TP_N) / N \quad (1)$$

$$Precision_c = TP_c / (TP_c + FP_c) \quad (2)$$

$$Recall_c = TP_c / (TP_c + FN_c) \quad (3)$$

$$F1_c = 2 \times Precision_c \times Recall_c / (Precision_c + Recall_c) \quad (4)$$

$$Macro\text{-}F1 = (F1_HOAX + F1_NON\text{-}HOAX) / 2 \quad (5)$$

In the validation protocol, metrics are computed from validated actual labels and per-sample predictions. When a test set contains only one class, aggregate values do not describe the ability to distinguish the two classes and are not used to claim binary classification performance. Latency is reported as the mean time per request, and the relative cost of configuration k is expressed as its ratio to zero-shot, L_k / L_0.

Audit Procedure

The audit of the initial run comprised four analyses. First, the distribution of actual labels in the recorded test set was counted. Second, the recorded macro-F1 values were

compared with the values implied by Equations (2)–(5) under a single-class test set (Equations 6–7), and the implied confusion counts were recovered as the recorded accuracy multiplied by the number of test items. Third, the uncertainty of each implied HOAX recall was estimated with the 95% Wilson score interval for a binomial proportion. Fourth, the dependence of mean latency on the number of examples k was summarised by the latency ratio L_k / L_0 and by an ordinary least-squares line fitted to the four configuration means. All derived values were computed from the recorded summary in Table 3, because the raw model responses were not archived.

Validity and Reproducibility

Internal validity was addressed by applying the same recorded test set and evaluation rules to all four configurations. Construct validity depends on the operational definitions of HOAX and NON-HOAX and on the quality of the curators' labels. External validity is limited by the diversity of sources, topics, and writing styles. For reproducibility, the validation protocol requires the prompt versions, the model and its parameters, the experiment dates, the raw responses, the parsed results, and the metric scripts to be archived, and repeated runs to be performed where the model settings allow, because LLM outputs can vary between calls. Table 7 reports which of these items were available for the initial run.

RESULTS AND DISCUSSION

Summary of the Initial Experimental Results

The metric summary supplied with the experiment records the results shown in Table 3 and Figure 3. These values are copied from the initial experiment summary and were not recomputed from per-sample predictions. Their presentation is therefore descriptive and must be read together with the test-set validity analysis that follows.

Key Findings

Four findings emerge from the initial run and its audit. First, the recorded test set contained only HOAX narratives, so the recorded accuracy and macro-F1 do not measure binary classification performance. Second, the recorded accuracy is equivalent to the recall of the HOAX class, which was highest for zero-shot (0.8675) and lowest for few-shot 8 (0.5200). Third, mean latency increased with every additional example, reaching 2.21 times the zero-shot value with eight examples. Fourth, the experiment files lacked several elements required for a reproducible LLM study (Table 7). Each finding is examined below and then compared with previous studies.

Table 3. Summary of initial results from the experiment files

Configuration	Recorded accuracy	Recorded macro-F1	Mean latency	Latency ratio
Zero-shot	86.75%	0.4645	0.418 s	1.00×
Few-shot 2	65.75%	0.3967	0.486 s	1.16×
Few-shot 4	70.50%	0.4135	0.632 s	1.51×
Few-shot 8	52.00%	0.3421	0.923 s	2.21×

Source: Initial experiment summary; latency ratio computed by the author (2026)

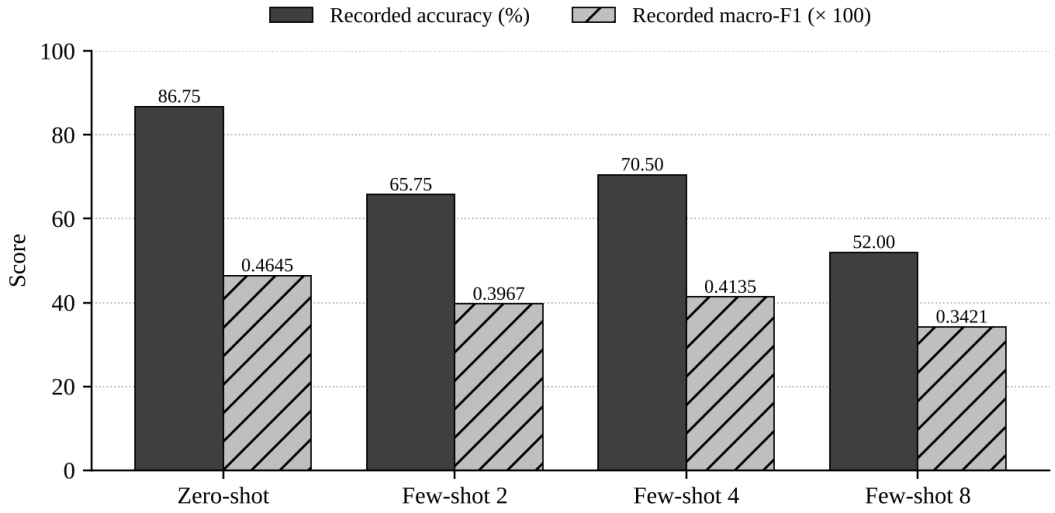


Figure 3. Recorded accuracy and macro-F1 of the four configurations (descriptive; not a validated binary evaluation)

Source: Initial experiment summary (2026)

Comparison of Recorded Accuracy and Macro-F1

Zero-shot has the highest recorded accuracy at 86.75%, followed by few-shot 4 at 70.50%, few-shot 2 at 65.75%, and few-shot 8 at 52.00%. The pattern shows no linear improvement as examples are added. Few-shot 4 scores higher than few-shot 2 but remains below zero-shot. Recorded macro-F1 follows the same general pattern: 0.4645 for zero-shot, 0.3967 for few-shot 2, 0.4135 for few-shot 4, and 0.3421 for few-shot 8.

Although zero-shot appears superior in the table, this advantage cannot be stated as a final scientific conclusion. The file audit showed that all 400 actual labels in the recorded test set are HOAX. Because NON-HOAX never occurs as an actual label in that test data, the model's performance on the NON-HOAX class cannot be properly evaluated, and accuracy and macro-F1 must be recomputed after the split has been repaired.

What the Recorded Metrics Actually Measure

The single-class composition of the test set can be verified directly from the recorded numbers. When every actual label is HOAX, there are no NON-HOAX items, so FP_H = 0 and TP_N = 0. Every HOAX prediction is then correct, and accuracy reduces to the recall of the HOAX class. Writing A for the recorded accuracy, Equations (2)–(5) give the following identities.

$$Precision_H = 1,\ Recall_H = A,\ F1_H = 2A / (1 + A) \quad (6)$$

$$F1_N = 0,\ Macro\text{-}F1 = A / (1 + A) \quad (7)$$

Equation (7) reproduces every recorded macro-F1 value to four decimal places: 0.8675/1.8675 = 0.4645, 0.6575/1.6575 = 0.3967, 0.7050/1.7050 = 0.4135, and 0.5200/1.5200 = 0.3421 (Table 4 and Figure 4b). This exact agreement independently confirms that the metrics were computed on a test set containing only HOAX narratives, and it allows the implied prediction counts to be recovered: zero-shot labelled 347 of the 400 hoax narratives as HOAX and 53 as NON-HOAX, whereas few-shot 8 labelled only 208 as HOAX

(Figure 4a). The recorded "accuracy" is therefore the hit rate on hoax narratives, and the recorded macro-F1 is bounded above by 0.5 regardless of model quality.

Table 4. Implied confusion counts and per-class metrics on the recorded single-class test set (n = 400)

Configuration	Predicted HOAX (TP)	Predicted NON-HOAX (FN)	Recall HOAX	F1 HOAX	F1 NON-HOAX	A/(1 + A)
Zero-shot	347	53	0.8675	0.9290	0.0000	0.4645
Few-shot 2	263	137	0.6575	0.7934	0.0000	0.3967
Few-shot 4	282	118	0.7050	0.8270	0.0000	0.4135
Few-shot 8	208	192	0.5200	0.6842	0.0000	0.3421

Source: Derived by the author from Table 3 using Equations (6)–(7) (2026)

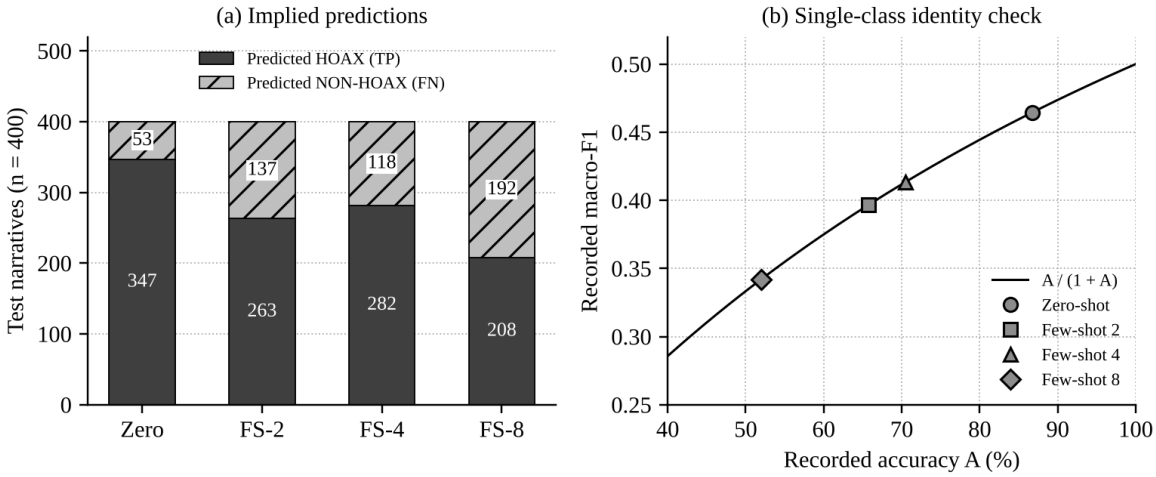


Figure 4. (a) Implied predictions on the 400 hoax test narratives; (b) recorded macro-F1 against recorded accuracy with the single-class identity A/(1 + A)
Source: Derived by the author (2026)

Read in this way, the initial results carry one interpretable signal: on hoax narratives, adding examples reduced the proportion flagged as HOAX, from 86.75% with zero-shot to 52.00% with eight examples, so that the miss rate rose from 13.25% to 48.00%. The 95% Wilson intervals for HOAX recall—83.1–89.7% for zero-shot, 61.0–70.2% for few-shot 2, 65.9–74.8% for few-shot 4, and 47.1–56.9% for few-shot 8 indicate that the gap between zero-shot and each few-shot configuration exceeds the sampling uncertainty of a 400-item set, whereas few-shot 2 and few-shot 4 cannot be distinguished.

Because all configurations were evaluated on the same items, these independent-sample intervals are approximate; a paired test such as McNemar's requires the per-sample predictions. The interval for few-shot 8 includes 50%, the hoax recall expected from random assignment of the two labels. One hypothesis is that the examples shifted the model towards the NON-HOAX label, as could occur if the example sets were dominated by NON-HOAX items or if ordering effects were present.

This hypothesis cannot be tested with the present data, because the examples were not archived and no NON-HOAX items were tested: a configuration that labels fewer narratives

as HOAX may also produce fewer false alarms on genuine news. The trade-off between hoax recall and false-alarm rate is precisely what the repaired test set must measure.

Latency and Context Cost

Recorded mean latency increased with the number of examples: 0.418 s for zero-shot, 0.486 s for few-shot 2, 0.632 s for few-shot 4, and 0.923 s for few-shot 8 (Figure 5), so that few-shot 8 was about 2.21 times slower than zero-shot (0.923/0.418). A least-squares line fitted to the four means gives an increase of approximately 0.065 s per added example (intercept 0.387 s; $R^2 = 0.98$); the marginal cost was 0.034 s per example from zero to two examples and about 0.073 s per example beyond two. This near-linear growth is consistent with the expectation that each example lengthens the input context that must be processed.

The absolute values, however, depend on factors that were not recorded the model, API settings, network conditions, service load, and output length and the fit rests on only four configuration means from a single run. The latency comparison is therefore robust in direction and relative magnitude within this run, but its absolute values should not be generalised to other deployments. Unlike the quality metrics, it does not depend on the label composition of the test set, which makes it the most robust finding of the initial run.

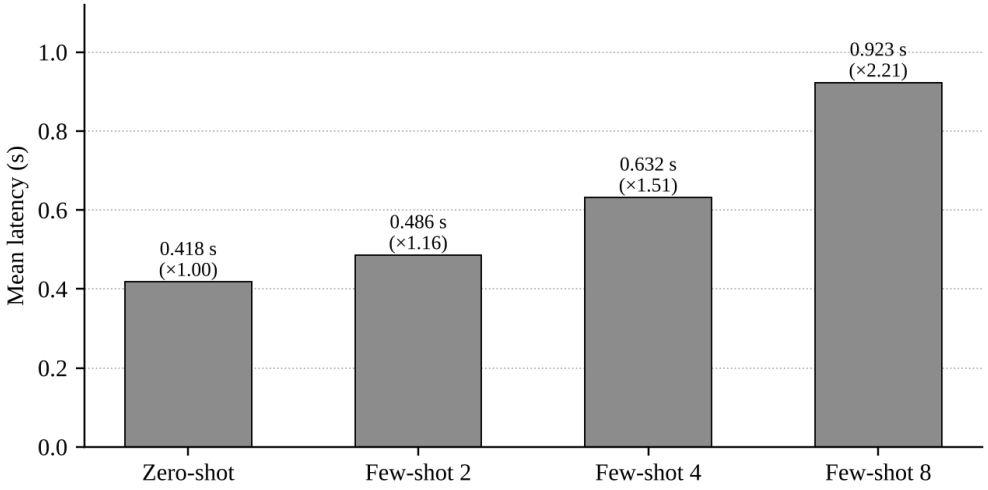


Figure 5. Mean latency per request and ratio to zero-shot
Source: Initial experiment summary; ratios computed by the author (2026)

The practical implication is that the number of examples must be weighed as a trade-off between prediction quality and efficiency. If no performance gain is demonstrated, adding examples increases time and cost without a clear benefit. The cost of long prompts is a recognised concern in LLM inference and has motivated prompt-compression methods such as LLMLingua (Jiang et al., 2023). Because classification quality in the initial results has not been validated, however, choosing zero-shot on the basis of these figures alone would be premature.

Test-Set Validity as the Central Issue

The available initial results show differences in recorded accuracy between strategies. Further inspection, however, found that all 400 rows of the recorded test set carry the HOAX label, which is a fundamental threat to a binary classification evaluation. This article does not

conceal the problem. Instead, the initial results are used as audit material and to clarify the conditions that must be met before final results can be published.

The most important methodological finding is the mismatch in test-set composition: all 400 recorded rows have the actual label HOAX (Figure 6). For a binary HOAX/NON-HOAX task, the test set should contain examples of both classes; otherwise the model is never tested on recognising NON-HOAX as an actual class. The F1-score of a class with no actual instances cannot be interpreted as a meaningful measure of performance. Opitz, (2024) likewise cautions that macro-averaged metrics depend on class prevalence and should be selected and interpreted explicitly rather than reported by default. The size of the recorded test set also departs from the planned design: 400 items correspond neither to 20% of the full dataset (800) nor to 20% of the HOAX class (200).

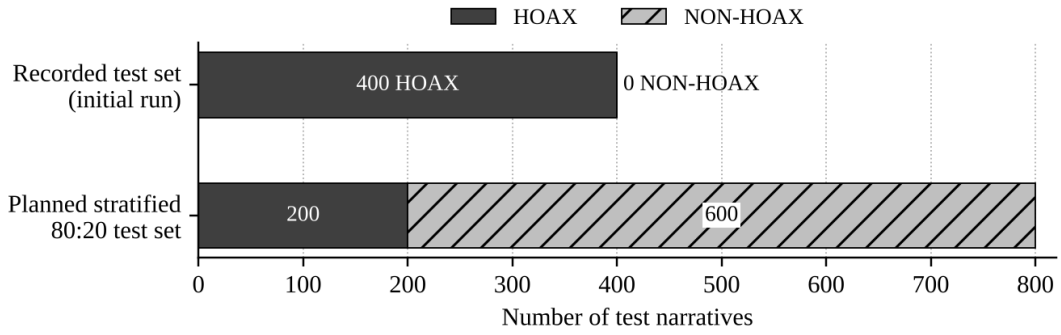


Figure 6. Composition of the recorded test set compared with the planned stratified 80:20 test set

Source: Experiment file audit and research design (2026)

The problem may originate at several stages of the pipeline, such as a filter that inadvertently keeps a single label, an incorrect index selection, sorting before slicing, or an incorrect file merge. Table 5 lists the diagnostic check and the expected outcome for each candidate cause. Such pipeline errors resemble the forms of data leakage that Kapoor & Narayanan, (2023) identified as a widespread source of irreproducible results in machine-learning-based science. Once the cause is found, all configurations need to be re-run on the same representative test set.

Table 5. Diagnostic checks for the single-class test set

Possible cause in the pipeline	Diagnostic check	Expected result after repair
Filter keeps only one label	Label counts before and after every filtering step	Both labels present after filtering
Sorting before slicing	Inspect whether data were ordered by label before the test slice was taken	Random or stratified selection, not ordered slicing
Incorrect index selection	Compare test indices with the split output	Indices match the stratified split
Mismatched file merge	Join text, actual label, and prediction on a unique ID	One-to-one rows; no label shift
Wrong label construction	Re-derive labels from the hoax and label columns	One consistent label per narrative

Source: Author's audit (2026)

Potential Source Label Confounding

The dataset underlying the design contains 4,000 narratives distributed as 1,000 HOAX and 3,000 NON-HOAX items (stored with the Indonesian label strings HOAKS and NON-HOAKS). Initial exploratory data analysis also revealed a potential link between data source and label: HOAX and NON-HOAX narratives come from different groups of sources. This situation can produce source label confounding, in which a model recognises characteristics of the source or writing style rather than the substance of the narrative. Data splitting and validity checks therefore need careful design.

The initial dataset contains 1,000 HOAX and 3,000 NON-HOAX narratives. Besides class imbalance, the EDA notes indicate that the sources of the two label groups do not fully overlap, so the label may correlate with the source. If texts from one portal are always NON-HOAX and texts from another source are always HOAX, a model can exploit the site name, writing format, or source-specific vocabulary as indirect cues. The recommended strategy is to review the sources, remove irrelevant metadata where necessary, and carry out cross-source evaluation when the amount of data allows. For an LLM used without fine-tuning, this risk enters mainly through the few-shot examples, which should therefore be drawn from several sources within each class.

Comparison with Previous Studies

Table 6 positions this study relative to the literature cited. Jin et al. (2025) explored prompting enriched with context, verification constraints, and prompt optimisation for fake news detection; their approach is broader than varying the number of examples and suggests that what the context contains matters more than how long it is. Haupt et al. (2024) showed that changes to the instructions can reduce misinformation detection accuracy in the context of their study. The decline in hoax recall observed here when examples were added is directionally consistent with that observation: modifying the prompt did not automatically improve detection and, on hoax items, reduced it.

Table 6. Position of this study relative to previous work

Study	Data and language	Approach	Relevance to this study
Al Amin et al. (2025)	Indonesian hoax news	Supervised 1D-CNN	Supervised baseline family for Indonesian text
Safira & Nurlayli (2023)	Indonesian news validity	Comparison of machine learning algorithms	Shows the role of dataset and evaluation design
Roslan & Mohd (2025)	PolitiFact and LIAR (English)	GPT-3.5-Turbo, zero-shot vs two-shot	Small, non-significant gains from two examples
Jin et al. (2025)	Fake news benchmarks	CAPE-FND: context, veracity constraints, prompt optimisation	Prompt design changes more than example count
Haupt et al. (2024)	Misinformation statements	Role-playing prompts for ChatGPT	Prompt changes can lower accuracy
Y. Hu & Tian, (2025)	Rumour tasks	Reasoning LLMs vs traditional models	Traditional models can still outperform LLMs
This study	4,000 Indonesian narratives	One LLM; 0, 2, 4, 8 examples	Valid only for HOAX recall (0.52–0.87) and latency (1.00–2.21×); binary performance not yet measurable

Source: Compiled by the author from the cited studies (2026)

Interpretation in the Context of the Literature

The initial results show no monotonic relationship between the number of examples and HOAX recall: the value fell from zero-shot to few-shot 2, rose slightly at few-shot 4—a difference within the overlap of their 95% intervals and fell sharply at few-shot 8. Combined with the monotonic increase in latency, this pattern means that, in this run, additional context raised the computational cost without a corresponding gain on the class that was measured. Because the test set contains no actual NON-HOAX labels, however, the pattern cannot be taken as evidence that zero-shot is the better binary classifier; it shows only that zero-shot was the most sensitive configuration for hoax narratives.

Comparison with Roslan and Mohd (2025) must be made with care. They reported small, non-significant improvements from zero-shot to two-shot prompting with GPT-3.5-Turbo on PolitiFact and LIAR, whereas the present run recorded a decline in hoax recall from zero-shot to two-shot. Both results contradict the assumption that adding examples reliably improves detection, but they differ in direction, which may reflect differences in language, domain, label definitions, test protocol, or example selection. Because the present model configuration was not recorded and the test set contained a single class, the comparison is qualitative: the literature indicates that few-shot effects can be small or variable, and the present data are consistent with that variability rather than serving as a direct numerical baseline.

CAPE-FND by Jin et al. (2025) changes more than the number of examples, as it introduces context and prompt optimisation mechanisms. The present study therefore cannot test the effectiveness of CAPE-FND; it focuses on the effect of example count when the core

prompt components are kept constant. Follow-up experiments may test balanced example selection, similarity-based example retrieval (J. Liu et al., 2022), or verification instructions as separate factors, but such changes should not be mixed into the main experiment without a clear design.

The test-set limitation is a more fundamental issue than the differences between configurations. If all actual labels are HOAX, the model is never tested on NON-HOAX as an actual class, so no conclusion can be drawn about false positives on NON-HOAX narratives or about the ability to separate the two classes. The appropriate step is to return to the pipeline that builds the test set, confirm the label distribution after splitting, and re-run every configuration.Pendyala and Hall (2024) examined the use of LLMs for misinformation detection together with approaches for explaining model output. Papageorgiou et al. (2024) reviewed the use of LLMs in fake news research, and Chen and Shu (2024) discussed the opportunities and challenges of countering misinformation in the LLM era.

In the Indonesian context, Al Amin et al. (2025) tested a 1D-CNN for hoax news detection and Safira and Nurlayli (2023) compared machine learning algorithms for news validity, while Hu and Tian (2025) found that traditional models can still outperform LLMs on rumour detection. These studies differ in dataset, model, and experimental design, so their performance figures are not compared directly with the initial results of this article. Their common implication is that supervised models remain a necessary reference point, which is why the validation protocol includes a supervised baseline evaluated on the same split. Ziems et al., (2024) reached a similar conclusion, reporting that LLMs used without fine-tuning failed to outperform the best fine-tuned models on taxonomic labelling tasks.

Reporting Requirements and Follow-Up Experiments

Several items required for a reproducible LLM experiment were missing from the source files (Table 7). Balloccu et al., (2024) reported that unfair or missing baseline comparisons and reproducibility problems are common in evaluations of closed-source LLMs, which underlines the need for such documentation. The first follow-up step is to repair the test set and re-run every configuration. Second, the results should report the confusion matrix and the precision, recall, and F1-score of each class. Third, the selection of few-shot examples should be documented and consistent, so that differences are not caused by examples that are too easy or unrepresentative. Fourth, latency should be measured over multiple requests under equivalent conditions. Where possible, repeated runs should be performed to assess the stability of the results, and a supervised baseline such as TF-IDF with a linear classifier should be evaluated on the same split to anchor the LLM results.

Table 7. Reproducibility checklist for the LLM prompting experiment

Reporting item	Status in the source files	Required action
Model name and version	Not documented	Record from API logs
Inference parameters (temperature, max tokens)	Not documented	Record and fix across runs
Execution dates and API settings	Not documented	Record per run
Exact prompt text per configuration	Partly documented	Archive verbatim prompts
Few-shot examples and their order	Not documented	Archive example IDs and order
Final split and class distribution	Inconsistent (400 HOAX only)	Rebuild stratified split
Raw responses and parsing rules	Not documented	Store raw outputs and parser
Per-class metrics and confusion matrix	Not reported	Compute after re-run

Source: Author's audit of the experiment files (2026)

Once the experiment has been validated, the article can be updated with final results and a discussion that answers whether few-shot examples provide a meaningful improvement over zero-shot. Until then, the figures presented here should be regarded as preliminary audit results and should not be used to declare a definitive best configuration.

Synthesis with Respect to the Research Objectives

With respect to the first objective, the initial results of the four configurations can be compared only as HOAX recall, which ranked zero-shot above few-shot 4 and few-shot 2, which were statistically indistinguishable, and few-shot 8 last; binary classification performance could not be determined. With respect to the second objective, latency rose with the number of examples in a near-linear manner, so that each additional example carried a measurable time cost.

With respect to the third objective, the audit produced a validation protocol (Figure 2 and Tables 2, 5, and 7) that specifies the conditions under which a valid HOAX/NON-HOAX comparison can be made. Whether few-shot prompting improves the identification of Indonesian hoax narratives therefore remains an open question that the re-run must answer.

Limitations

This manuscript is based on initial experimental results. The model name, inference parameters, execution dates, final data split, and per-class prediction details have not been fully confirmed in the source documents and must be added from the original experiment logs. The implied confusion counts in Table 4 are derived analytically from the recorded summary and should be checked against the per-sample prediction files, and the Wilson intervals treat the configurations as independent samples although they were evaluated on the same items. The initial figures must not be treated as final results until the audit and the re-run are complete.

CONCLUSION

This study compared four prompting configurations—zero-shot and few-shot with 2, 4, and 8 examples for identifying Indonesian-language hoax narratives with a single LLM. Regarding the first objective, the audit showed that the recorded test set contained only HOAX items; the recorded accuracies (86.75%, 65.75%, 70.50%, and 52.00%) therefore represent hoax recall, which declined as examples were added, while binary performance remains undetermined. Regarding the second objective, mean latency increased almost linearly with the number of examples, reaching 2.21 times the zero-shot value. Regarding the third objective, the study formulated a validation protocol comprising stratified splitting, leakage-free example selection, per-class metrics, and complete model documentation. The practical implication is that few-shot prompting should not be assumed beneficial for Indonesian hoax detection and must be justified empirically against its computational cost before deployment.

FUNDING

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors. The author declares that no external financial support was received for conducting the research or preparing and publishing this manuscript.

ACKNOWLEDGEMENTS

The author expresses sincere appreciation to the providers of the Indonesian-language hoax and news narratives used in this study. Gratitude is also extended to colleagues who contributed through their review of the experimental files and provided constructive

feedback. Their contributions supported the evaluation of the experimental procedures and the identification of methodological limitations in comparing zero-shot and few-shot prompting strategies.

CONFLICT OF INTEREST

The author declares that there are no financial, commercial, professional, or personal relationships that could be perceived as potential conflicts of interest concerning this research. The study was conducted independently, and the interpretation and presentation of the findings were not influenced by any external interests.

AUTHOR CONTRIBUTIONS

The author was responsible for the conceptualization, research methodology, data curation, experimental design and implementation, formal analysis, and interpretation of the findings. The author also contributed to data visualization, manuscript preparation, critical review, and editing. The author reviewed and approved the final version of the manuscript and assumes full responsibility for its scientific integrity and content.

GENERATIVE AI STATEMENT

The author acknowledges the use of generative artificial intelligence (AI) tools to assist in improving the linguistic clarity, grammatical accuracy, and overall readability of the manuscript. The use of generative AI was limited to supporting the writing and editing process and did not replace the author's intellectual contributions, research analysis, or scientific judgment. The author reviewed and verified the AI-assisted content and assumes full responsibility for the accuracy, originality, integrity, and final content of the manuscript.

REFERENCES

Aïmeur, E., Amri, S., & Brassard, G. (2023). Fake news, disinformation and misinformation in social media: a review. *Social Network Analysis and Mining*, *13*(1), 30. https://doi.org/10.1007/s13278-023-01028-5

Al Amin, M. Z., Furqon, M. A., & Wijonarko, D. (2025). Indonesian hoax news detection using one-dimensional convolutional neural network. *Jurnal Nasional Teknik Elektro Dan Teknologi Informasi*, *14*(2), 161–169. https://doi.org/10.22146/jnteti.v14i2.19050

Balloccu, S., Schmidtová, P., Lango, M., & Dusek, O. (2024). Leak, Cheat, Repeat: Data Contamination and Evaluation Malpractices in Closed-Source LLMs. *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, 67–93. https://doi.org/10.18653/v1/2024.eacl-long.5

Chen, C., & Shu, K. (2024). Combating misinformation in the age of LLMs: Opportunities and challenges. *AI Magazine*, *45*(3), 354–368. https://doi.org/10.1002/aaai.12188

Dong, Q., Li, L., Dai, D., Zheng, C., Ma, J., Li, R., Xia, H., Xu, J., Wu, Z., Chang, B., Sun, X., Li, L., & Sui, Z. (2024). A Survey on In-context Learning. *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 1107–1128. https://doi.org/10.18653/v1/2024.emnlp-main.64

Du, M., He, F., Zou, N., Tao, D., & Hu, X. (2024). Shortcut Learning of Large Language Models in Natural Language Understanding. *Communications of the ACM*, *67*(1), 110–120. https://doi.org/10.1145/3596490

Guo, Z., Schlichtkrull, M., & Vlachos, A. (2022). A Survey on Automated Fact-Checking. *Transactions of the Association for Computational Linguistics*, *10*, 178–206. https://doi.org/10.1162/tacl_a_00454

Haupt, M. R., Yang, L., Purnat, T., & Mackey, T. (2024). Evaluating the influence of role-playing prompts on ChatGPT's misinformation detection accuracy: Quantitative study.

JMIR Infodemiology, *4*, e60678. https://doi.org/10.2196/60678

Hu, B., Sheng, Q., Cao, J., Shi, Y., Li, Y., Wang, D., & Qi, P. (2024). Bad Actor, Good Advisor: Exploring the Role of Large Language Models in Fake News Detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, *38*(20), 22105–22113. https://doi.org/10.1609/aaai.v38i20.30214

Hu, Y., & Tian, X. (2025). Evaluating reasoning large language models on rumor generation, detection, and debunking tasks. *IScience*, *28*(11), 113690. https://doi.org/10.1016/j.isci.2025.113690

Jiang, H., Wu, Q., Lin, C.-Y., Yang, Y., & Qiu, L. (2023). LLMLingua: Compressing Prompts for Accelerated Inference of Large Language Models. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 13358–13376. https://doi.org/10.18653/v1/2023.emnlp-main.825

Jin, W., Gao, Y., Tao, T., Wang, X., Wang, N., Wu, B., & Zhao, B. (2025). Veracity-oriented context-aware large language models-based prompting optimization for fake news detection. *International Journal of Intelligent Systems*, *2025*(1), 5920142. https://doi.org/10.1155/int/5920142

Kapoor, S., & Narayanan, A. (2023). Leakage and the reproducibility crisis in machine-learning-based science. *Patterns*, *4*(9), 100804. https://doi.org/10.1016/j.patter.2023.100804

Liu, J., Shen, D., Zhang, Y., Dolan, B., Carin, L., & Chen, W. (2022). What Makes Good In-Context Examples for GPT-3? *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, 100–114. https://doi.org/10.18653/v1/2022.deelio-1.10

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., & Neubig, G. (2023). Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. *ACM Computing Surveys*, *55*(9), 1–35. https://doi.org/10.1145/3560815

Opitz, J. (2024). A Closer Look at Classification Evaluation Metrics and a Critical Reflection of Common Evaluation Practice. *Transactions of the Association for Computational Linguistics*, *12*, 820–836. https://doi.org/10.1162/tacl_a_00675

Papageorgiou, E., Chronis, C., Varlamis, I., & Himeur, Y. (2024). A survey on the use of large language models (LLMs) in fake news. *Future Internet*, *16*(8), 298. https://doi.org/10.3390/fi16080298

Pendyala, V. S., & Hall, C. E. (2024). Explaining misinformation detection using large language models. *Electronics*, *13*(9), 1673. https://doi.org/10.3390/electronics13091673

Roslan, M. N. S., & Mohd, M. (2025). Performance comparison of zero-shot and two-shot prompting in detecting fake news using large language models. *Malaysian Journal of Computer Science*, *38*, 130–140. https://doi.org/10.22452/mjcs.vol38spc.8

Safira, R. E., & Nurlayli, A. (2023). Comparative analysis of Indonesian news validity detection accuracy using machine learning. *Journal of Engineering and Applied Technology*, *4*(1), 40–51. https://doi.org/10.21831/jeatech.v4i1.58791

Yefferson, D. Y., Lawijaya, V., & Girsang, A. S. (2024). Hybrid model: IndoBERT and long short-term memory for detecting Indonesian hoax news. *IAES International Journal of Artificial Intelligence (IJ-AI)*, *13*(2), 1913. https://doi.org/10.11591/ijai.v13.i2.pp1913-1924

Ziems, C., Held, W., Shaikh, O., Chen, J., Zhang, Z., & Yang, D. (2024). Can Large Language Models Transform Computational Social Science? *Computational Linguistics*, *50*(1), 237–291. https://doi.org/10.1162/coli_a_00502