Journal of Soft Computing Exploration

Homepage: htttps://shmpublisher.com/index.php/joscex

p-ISSN: 2746-7686
e-ISSN: 2746-0991

.

Clickbait detection in indonesian news headlines using various prompting strategies in large language models

Nandhika Rega Rohadi[1], Fajar Muslim[2], Dewi Wisnu Wardani[3],
[1,2,3] Faculty of Information Technology and Data Science, Universitas Sebelas Maret, Indonesia

Article Info

Article history:

Received May 21, 2026
Revised June 24, 2026
Accepted June 26, 2026

Keywords:

Clickbait detection
Large language model
Zero-shot
Few-shot
Self-consistency

ABSTRACT

Clickbait detection in news headlines is a critical task in Natural Language Processing (NLP) related to information quality and the credibility of digital journalism. While traditional machine learning and deep learning approaches have demonstrated impressive performance in clickbait detection, they are limited by a heavy reliance on extensive annotated datasets and significant computational requirements for model training. Unlike previous methods, Large Language Models (LLMs) do not require massive amounts of annotated data. LLMs allow classification tasks to be solved through zero-shot and few-shot prompting without additional retraining. However, the effectiveness of these models can depend significantly on prompting configuration. Despite this, linguistically-enriched prompting strategies have not been widely evaluated for non-English domains such as Indonesian news headlines. This study aims to analyze the performance of various LLM prompting strategies in detecting Indonesian-language clickbait headlines. For this purpose, this study evaluated Llama 4 on the CLICK-ID dataset using multiple combinations of plain and linguistically-enriched prompts (zero-shot and few-shot) alongside advanced inference techniques (self-consistency, weighted self-consistency, and Self-Refine). Performance was measured via Accuracy and Macro F1-scores against a fine-tuned IndoBERT as baseline model. The results show that the prompting approach in the Large Language Models (LLMs) can be used to classify Indonesian clickbait effectively. The use of linguistic prompting and few-shot managed to provide the best performance, which achieved an accuracy of 0.90 and a Macro F1-score of 0.89.

Corresponding Author:

Nandhika Rega Rohadi,
Fakultas Teknologi Informasi dan Sains Data,
Universitas Sebelas Maret,
Jalan Ir. Sutami 36A Kentingan, Jebres, Surakarta, Jawa Tengah, Indonesia.
Email: nandhikaregarohadi@student.uns.ac.id
https://doi.org/10.52465/joscex.v7i2.131

1. INTRODUCTION

Technological advancements, particularly the internet, make people prefer online news portals to access news [1]. These online news portals typically profit from advertisements displayed when news is accessed. Therefore, high traffic is crucial for online news portals. The higher the traffic, the more ads displayed, thus increasing revenue. High traffic also influences algorithms, as trending news tends to be recommended to more people. To exploit these recommendation systems and ensure their articles stand out,

online news portal often rely on attention-grabbing tactics, making clickbait one of the most popular strategies to generate high traffic [2].

Clickbait is a strategy to attract users to click on news stories by using titles that arouse curiosity [3]. Clickbait titles contain incomplete information and are structured with sensational and provocative words. The more compelling the title, the more people will read it, resulting in increased traffic. The use of clickbait titles has been shown to increase the desire to read the news [4]. While clickbait has a positive impact on online news portals in the form of increased traffic, but it also has several negative impacts. Prioritizing engagement can undermine journalistic integrity [1]. Moreover, it leads to negative user experiences, such as reader disappointment when the actual news content fails to meet the expectations derived from the headline [5]. Despite these negative impacts, the use of clickbait remains widespread because, while there are several regulations that can indirectly ensnare clickbait creators, these regulations are flawed and do not explicitly prohibit the use of clickbait [6].

Clickbait detection is one of the solutions to address the current proliferation of clickbait news. Machine learning is one of the approach frequently used for clickbait detection. Several machine learning models, such as Naïve Bayes, Logistic Regression, and Support Vector Machines, have been applied to clickbait detection [7]. Deep learning models such as Bidirectional Encoder Representation from Transformers (BERT) are also popular for clickbait detection. BERT models have higher accuracy than conventional machine learning models, but require longer training times [8]. The performance of deep learning models such as BERT, RoBERTa, and XLNet models can be improved by applying Human Semantic Knowledge [9]. Furthermore, these deep learning approaches have been successfully adapted for other languages, such as utilizing Bi-LSTM and BERT with semantic and syntactic features to improve detection in Chinese-language news [10].

In the specific domain of Indonesian-language news, several studies have conducted extensive clickbait detection. An ensemble stacking model consisting of three classifiers: Multinomial Naive Bayes, Logistic Regression, and Support Vector Machine achieved an accuracy of 0.77 [11]. Furthermore, hyperparameter tuning was applied to several LSTM models, with the best model achieving a validation accuracy of 0.80 [12]. Transformer-based models, namely fine-tuned RoBERTa and IndoBERT, have very high performance on Indonesian clickbait detection tasks, with an accuracy of 0.92 [13]. Other study used M-Bert resulting in accuracy and f1-score of 0.91 [14]. Additionally, a hybrid architecture combining BERT with a Convolutional Neural Network (CNN) has been explored, demonstrating promising and competitive performance with precision, recall, and F1-score values of 0.91, 0.86, and 0.89, respectively [15].

While deep learning models can achieve high accuracy, their primary drawback is the necessity for significant computational resources and large-scale labeled datasets to train optimally, which poses severe challenges in resource-constrained or limited-data scenarios. One of relevant approach to address these drawback is the application of Large Language Models (LLMs) via zero-shot and few-shot prompting. Zero-shot prompting is a method where the LLMs performs a task without being given any examples. Few-shot prompting, on the other hand, involves the LLMs performing a task with only a few examples. Not only do they address the computational resource requirement issue, but they are also very useful when available data is limited [16].

Large Language Models (LLMs) have demonstrated remarkable success when tested on various NLP tasks, such as zero-shot mathematical reasoning, text summarization, machine translation, information extraction, and sentiment analysis [17]. Furthermore, LLMs have been applied to clickbait detection such as in a study using a Chinese dataset [18]. However, LLMs have not yet been able to surpass the performance of state-of-the-art models in clickbait detection [19]. The reasoning ability of LLMs can be improved with methods such as self-consistency, which replaces greedy decoding by generating multiple reasoning paths and selecting the most consistent answer [20]. As an extension, this study also utilizes weighted self-consistency, requiring the LLM to assign confidence scores to each path to choose the most reliable label. Furthermore, another relevant approach is the self-refine method, which mimics human revision by having the LLM evaluate its initial label to produce a final assessment one [21]. This study will systematically apply these advanced prompting strategies, as they have been widely proven to enhance LLM reasoning, to achieve higher accuracy in clickbait classification.

The application of Large Language Models (LLMs) for clickbait detection in Indonesian news remains relatively underexplored. To address this gap, this study aims to evaluate the performance of LLMs in identifying clickbait within Indonesian-language contexts. This study systematically utilize several approaches, encompassing prompt types such as plain and linguistically enriched prompts, basic prompting strategies like zero-shot and few-shot setups, and inference refinement techniques, namely self-consistency, weighted self-consistency, and self-refine. Ultimately, this research is expected to provide valuable insights into how each approach influences LLM performance in clickbait detection.

2. METHOD

In this study, the clickbait detection methodology involves several key stages, namely dataset collection and preparation, prompt engineering and LLM inference, the development of a fine-tuned BERT baseline, performance evaluation, and a final comparative analysis. The research framework is shown in Figure 1.

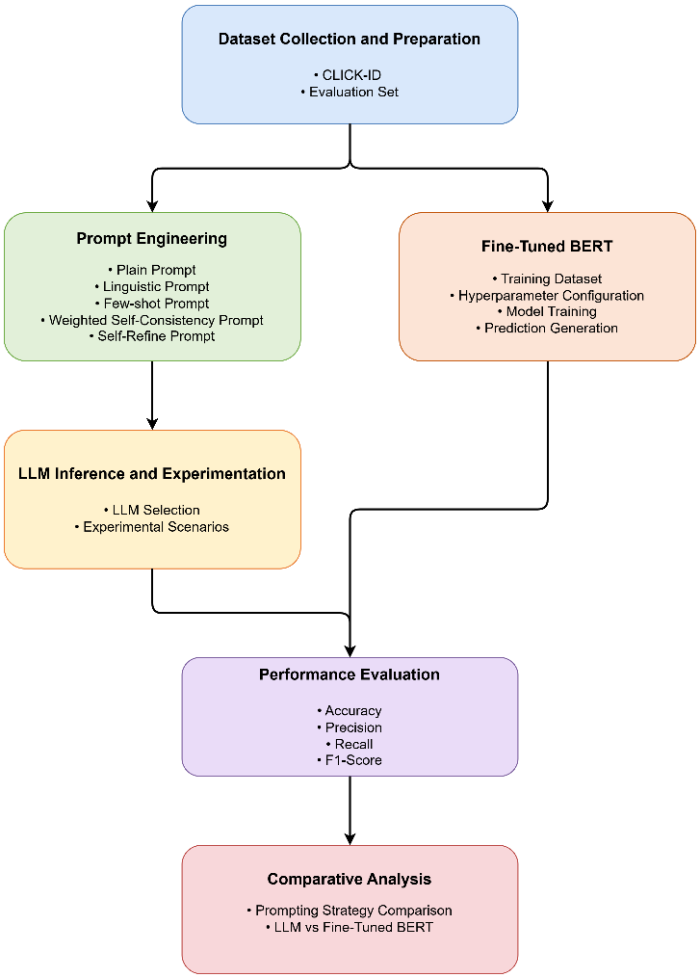


Figure 1. Research flowchart

Dataset Collection and Preparation

This study used CLICK-ID, a dataset containing Indonesian-language news titles taken from 12 news media outlets for use in clickbait detection. The news media taken are detikNews, Fimela, Kapanlagi, Kompas, Liputan6, Republika, Sindonews, Tempo, Tribunnews, Okezone, Wowkeren, and Posmetro-Medan. In total, there are 15,000 news titles consisting of 8,710 non-clickbait titles and 6,290 clickbait titles. The label annotation process was carried out by three annotators, with the majority vote being used as the ground truth [22].

From this comprehensive dataset, a subset of 300 news titles was randomly selected to serve as the evaluation sample. This specific sample size was chosen to ensure experimental efficiency, considering that the study involved several prompting strategies requiring iterative inference processes on a Large Language Model. Furthermore, limited computing resources and time constraints were also major considerations when determining this sample size.

Prompt Engineering

In the prompt engineering stage, this study implements several prompting strategies to optimize the Large Language Model's reasoning capabilities, ranging from basic approaches to advanced inference refinement techniques. The baseline strategies utilized include zero-shot and few-shot prompting. Zero-shot is an approach where the model performs a task without being given any prior examples, relying solely on the knowledge gained during pretraining [23]. The primary advantage of this approach is that it requires neither labeled data nor additional training. Few-shot learning allows the model to recognize patterns from a small amount of training data [16]. By mimicking humans, who can learn effectively from limited examples, few-shot learning focuses on the model's ability to generalize to new tasks. This approach also successfully overcomes the problems of large computational resources and long training times because it avoids the extensive training data required by traditional methods [24].

To further enhance the model's reasoning capabilities, this study applies advanced inference refinement techniques, starting with self-consistency. Self-consistency is a decoding strategy developed to replace the greedy decoding typically used in chain-of-thought (CoT) prompting. By creating multiple reasoning paths, the most consistent answer emerges and is determined by a majority vote [25]. In standard self-consistency, the most frequently occurring label is chosen without considering the model's confidence for each prediction [20]. To address this specific limitation, this study also tests weighted self-consistency, where the model assigns a confidence score to each reasoning path, and the label with the highest accumulated score is selected. However, because both methods require generating multiple reasoning paths, weighted self-consistency inherently shares the same drawbacks as standard self-consistency, namely higher computational overhead and longer inference times [20]. Alternatively, this study also incorporates self-refine, an iterative approach that circumvents parallel generation by drawing inspiration from human behavior. Reflecting the tendency to revise initial assessments after receiving feedback, the LLM generates an initial output and then evaluates it to provide constructive feedback. This feedback is subsequently used to improve the initial output, resulting in a better final prediction. This approach is highly efficient as it does not require additional training because the same LLM functions seamlessly as the generator, feedback provider, and refiner [21]. To implement the basic and advanced prompting strategies discussed previously, specific instruction templates were formulated. The prompt designs utilized in this study consist of several types, which are detailed below.

Plain Prompt

This prompt is used for two scenarios: plain-zero-single and plain-zero-sc. In this prompt, the model is only instructed to classify news headlines as CLICKBAIT or NOT_CLICKBAIT without being given any labeled data examples or other specific instructions. This prompt is suitable for evaluating the model's basic ability to detect clickbait headlines because the model can only utilize the knowledge learned during pre-training. This baseline capability is tested using the minimalistic instruction format demonstrated in Figure 2.

Plain Prompt

```
Klasifikasikan judul berita dibawah ini sebagai clickbait atau non-clickbait

JUDUL:
"{title}"

ATURAN PENTING:
- Jangan bersikap netral.
- Pilih salah satu secara tegas.

Output HARUS hanya salah satu:
CLICKBAIT
atau
NOT_CLICKBAIT
Tanpa penjelasan tambahan.
```

Figure 2. Plain prompt

Few-Shot Prompt

This prompt is used for two scenarios: plain-few-single and plain-few-sc. In this prompt, the model is given several labeled data examples. This prompt evaluates the model's ability to learn from newly given examples (in-context-learning) to help complete the task. As shown in Figure 3, this prompt incorporates a perfectly balanced subset of 10 examples consisting of 5 clickbait and 5 non-clickbait headlines, to prevent data leakage, the labeled examples provided in the prompt are explicitly excluded from the evaluation dataset, this study did not experiment with different combinations of few-shot examples.

Plain Few-Shot Prompt

```
Klasifikasikan judul berita dibawah ini sebagai clickbait atau non-clickbait

Berikut beberapa contoh:

JUDUL: "Masuk Radar Pilwalkot Medan, Menantu Jokowi Bertemu DPW NasDem Sumut"
OUTPUT: NOT_CLICKBAIT

JUDUL: "Malaysia Sudutkan RI: Isu Kabut Asap hingga Invasi Babi"
OUTPUT: NOT_CLICKBAIT

JUDUL: "Viral! Driver Ojol di Bekasi Antar Pesanan Makanan Pakai Sepeda"
OUTPUT: CLICKBAIT

JUDUL: "Ada Motor Nyangkut di Atas Bambu di Sleman, Kok Bisa?"
OUTPUT: CLICKBAIT

JUDUL: "Pesan Gamblang Poyuono Menolak Revisi UU KPK"
OUTPUT: CLICKBAIT

JUDUL: "Kocak! Maling di Rumah Mewah Jakut Terekam CCTV Bingung Cari Jalan Kabur"
OUTPUT: CLICKBAIT

JUDUL: "Viral Video Diduga Baku Tembak di Sleman, Ini Kata Polisi"
OUTPUT: CLICKBAIT

JUDUL: "Kemensos Salurkan Rp 7,3 M bagi Korban Kerusuhan Sosial di Papua"
OUTPUT: NOT_CLICKBAIT

JUDUL: "Mayat Pria Ditemukan di Kalimalang Bekasi, Diduga Korban Tenggelam"
OUTPUT: NOT_CLICKBAIT

JUDUL: "KPK Tanggapi DPR: Penegakan Hukum Tak Boleh Terikat Komitmen Politik"
OUTPUT: NOT_CLICKBAIT

Sekarang klasifikasikan judul berikut:

JUDUL: "{title}"

JUDUL:
"{title}"

ATURAN PENTING:
- Jangan bersikap netral.
- Pilih salah satu secara tegas.

Output HARUS hanya salah satu:
CLICKBAIT
atau
NOT_CLICKBAIT
Tanpa penjelasan tambahan.
```

Figure 3. Few-shot prompt

Linguistic Prompt

This prompt is used in eight scenarios: linguistic-zero-single, linguistic-few-single, linguistic-zero-sc, linguistic-few-sc, linguistic-zero-wsc, linguistic-few-wsc, linguistic-zero-self-refine, and linguistic-few-self-refine. Similar to few-shot, the model's ability to learn from the linguistic features in the prompt will be used to improve classification quality. Specific grammatical and stylistic examples are explicitly defined to aid the model's reasoning within the structure of this linguistic prompt, as outlined in Figure 4.

Linguistic Prompt

```
Klasifikasikan judul berita dibawah ini sebagai clickbait atau non-clickbait

PETUNJUK KLASIFIKASI:

- Fokus pada gaya bahasa judul, bukan isi beritanya.

- CLICKBAIT biasanya ditandai oleh satu atau lebih ciri berikut:
1. Tanda baca atau struktur yang menarik perhatian berlebihan, misalnya tanda
seru, tanda tanya, ellipsis, atau pola yang terasa dramatis.
2. Forward reference, yaitu judul yang sengaja menahan informasi dan memancing
rasa penasaran, misalnya:
   "ini alasannya", "begini", "ternyata", "inilah", "simak", "fakta", "daftar", "yang perlu
kamu tahu".
3. Kata-kata emosional atau sensasional, misalnya:
   "bikin", "heboh", "ngakak", "terkuak", "mengerikan", "panik", "kaleng-kaleng",
"kaget", "bangga", "tajir".
4. Struktur listicle atau numerik, misalnya judul diawali angka:
   "3 ...", "5 ...", "10 ...", "11 langkah ...".
5. Unsur CTA (call-to-action), misalnya:
   "simak", "cek", "yuk", "lihat", "baca", "jangan", "ikuti".

- NOT_CLICKBAIT biasanya berupa judul yang langsung, deskriptif, dan informatif
tanpa unsur memancing penasaran yang kuat.
- Jika judul memuat beberapa ciri clickbait di atas, pilih CLICKBAIT.
- Jika judul netral, lugas, dan hanya menyampaikan fakta atau kejadian, pilih
NOT_CLICKBAIT.
- Jangan bersikap netral.
- Jangan beri penjelasan tambahan.

Sekarang klasifikasikan judul berikut:

JUDUL:
"{title}"

ATURAN PENTING:
- Jangan bersikap netral.
- Pilih salah satu secara tegas.

Output HARUS hanya salah satu:
CLICKBAIT
atau
NOT_CLICKBAIT
Tanpa penjelasan tambahan.
```

Figure 4. Linguistic prompt

Weighted Self-Consitency Prompt

This prompt is used in four scenarios: plain-zero-wsc, plain-few-wsc, linguistic-zero-wsc, and linguistic-few-wsc. Unlike standard configurations, the model is explicitly instructed to generate a structured output that consists of both a classification label and a numerical confidence score. This specific mechanism ensures that highly confident predictions exert a proportionally greater influence on the final self-consistency voting results, as depicted in Figure 5.

Weighted Self-Consistency Prompt

```
Klasifikasikan judul berita dibawah ini sebagai clickbait atau non-clickbait

JUDUL:
"{title}"

ATURAN PENTING:
- Jangan bersikap netral.
- Pilih salah satu secara tegas.
- Sertakan tingkat keyakinan antara 0 dan 1.

Output HARUS hanya salah satu format berikut:
CLICKBAIT | 0.85
atau
NOT_CLICKBAIT | 0.85
Tanpa penjelasan tambahan.
```

Figure 5. Weighted Self-consistency prompt

Self Refine Prompt

This prompt is used in four scenarios: plain-zero-self-refine, plain-few-self-refine, linguistic-zero-self-refine, and linguistic-few-self-refine. In this prompt, the model is instructed to refine its output with self-generated feedback. To execute this, the methodology utilizes a dual-prompt structure consisting of an

"Evaluator" prompt to critique the initial draft and provide actionable feedback, followed by a "Refiner" prompt to determine the final classification based on that feedback. By explicitly separating the roles of feedback provider and refiner, this technique thoroughly tests the model's iterative reasoning capabilities, as detailed in the prompt templates in Figure 6.

Evaluator Prompt

```
Kamu adalah evaluator untuk klasifikasi clickbait.

JUDUL:
"{title}"

DRAFT / PREDIKSI SEBELUMNYA:
{draft_text}

Tugasmu:
1. Nilai apakah draft ini sudah tepat.
2. Berikan feedback singkat, spesifik, dan actionable.
3. Tentukan label yang paling tepat.

Output HARUS persis dalam format ini:
FEEDBACK: <satu kalimat singkat>
LABEL: CLICKBAIT atau NOT_CLICKBAIT
```

Refiner Prompt

```
Kamu adalah refiner untuk klasifikasi clickbait.

JUDUL:
"{title}"

DRAFT / PREDIKSI SEBELUMNYA:
{draft_text}

FEEDBACK:
{feedback_text}

Tugasmu:
- Perbaiki draft berdasarkan feedback.
- Output hanya label final.

Output HARUS hanya salah satu:
CLICKBAIT
atau
NOT_CLICKBAIT
Tanpa penjelasan tambahan.
```

Figure 6 Self-refine prompt

LLM Experimental scenarios

For the primary experimental pipeline, this study utilizes the Llama series, a collection of large language models launched by Meta in February 2023 that are widely used in computer vision and natural language understanding tasks [26]. Specifically, the Llama 4:latest model is employed to execute the various prompting strategies. To ensure a controlled experimental environment, the inference process was conducted on an Ubuntu-based system using Jupyter Notebook, with the LLM running locally through the Ollama platform service.

From the dataset, 300 Indonesian-language news titles will be selected for classification. Experiments will be conducted by comparing the performance of the LLM approach with the fine-tuned IndoBERT approach as a baseline. Single inference and self-refinement strategies will be performed with a temperature of 0. This is because a temperature of 0 tends to be deterministic and therefore best represents the performance of the LLM in this case. For the self-consistency and weighted self-consistency strategies, five reasoning paths are generated using a temperature of 0.7 to ensure diversity in each reasoning path. Each experimental scenario was executed twice to reduce randomness. The experimental scenarios to be conducted in this study are as shown in Table 1 below:

Table 1. List of experimental scenarios

No	Prompt Type	Base Strategies	Inference Strategy	Scenario Description
1	Plain	Zero-shot	Single Inference	Basic prompt without linguistic features, one-time inference
2	Plain	Zero-shot	Self-Consistency	Basic prompt, multiple reasoning path, majority vote
3	Plain	Zero-shot	Weighted Self-Consistency	Basic prompt, weighted aggregation based on confidence
4	Plain	Zero-shot	Self-Refine	Basic prompt with classification and reflection stages
5	Plain	Few-shot	Single Inference	Basic prompt with examples, one time inference
6	Plain	Few-shot	Self-Consistency	Few-shot + majority vote aggregation
7	Plain	Few-shot	Weighted Self-Consistency	Few-shot + weighted aggregation
8	Plain	Few-shot	Self-Refine	Few-shot + self-reflection
9	Linguistic	Zero-shot	Single Inference	Prompts with explicit linguistic features, , one-time inference
10	Linguistic	Zero-shot	Self-Consistency	Linguistic prompt + majority vote
11	Linguistic	Zero-shot	Weighted Self-Consistency	Linguistic prompt + weighted aggregation
12	Linguistic	Zero-shot	Self-Refine	Linguistic prompt + self-reflection
13	Linguistic	Few-shot	Single Inference	Linguistic prompt + examples
14	Linguistic	Few-shot	Self-Consistency	Linguistic + few-shot + majority vote
15	Linguistic	Few-shot	Weighted Self-Consistency	Linguistic + few-shot + weighted aggregation
16	Linguistic	Few-shot	Self-Refine	Linguistic + few-shot + self-reflection

Fine-tuned BERT

IndoBERT is a language model based on the BERT architecture specifically developed for understanding Indonesian text. Because it is pretrained on extensive Indonesian corpora, the model offers a distinct advantage in capturing specific linguistic nuances and cultural contexts that general-purpose multilingual models often struggle to represent [27]. The IndoBERT model used in this study is IndoBERT-p1. The fine-tuning process was carried out on the prepared dataset, in which 300 samples were first separated as an unseen set for final evaluation and comparison with the LLM approach, while the remaining data served as the training dataset. To address class imbalance, undersampling was applied to the majority non-clickbait class, following the approach adopted in [13], and the resulting balanced data was split with a 4:1 ratio into training and validation sets. Each headline was tokenized using the IndoBERT-p1 tokenizer with a maximum padding length of 60 tokens, based on the token length distribution analysis of the dataset.

The model was trained using a configuration adapted from from Sirusstara et al. [13], including a batch size of 64, a learning rate of 9e-6 with the Adam optimizer, and a dropout probability of 0.5 applied to both hidden and attention layers to mitigate overfitting given the relatively small dataset size. Cross-entropy loss was used as the loss function, consistent with the binary classification task of distinguishing clickbait from non-clickbait headlines. Training was conducted for a maximum of 10 epochs with early stopping based on validation F1-score, where the model checkpoint with the highest F1-score was retained as the best model. Training loss, validation loss, accuracy, precision, recall, and F1-score were monitored at each epoch to ensure proper convergence. The best-performing model was subsequently used to generate predictions on the unseen set, which formed the basis for performance evaluation and comparative analysis against the prompt-engineering-based LLM approach.

Performance Evaluation

To evaluate the performance of the classification models, this study utilizes a confusion matrix. A confusion matrix is an evaluation method commonly used in classification tasks that compares the model's predictions with the actual ground truth labels, providing a detailed analysis of model performance [28]. In the context of this clickbait detection research, the confusion matrix consists of four main components:True Positive (TP): A clickbait label correctly predicted as clickbait, True Negative (TN): A non-clickbait label correctly predicted as non-clickbait, False Positive (FP): A non-clickbait label incorrectly predicted as clickbait, False Negative (FN): A clickbait label incorrectly predicted as non-clickbait.

Based on these confusion matrix components, the model's performance is measured using the following four metrics:

Accuracy

Accuracy measures the overall percentage of correct predictions made by the model across all classes, equation (1) is the formula for calculating accuracy.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

Precision

Precision measures the quality of the positive predictions, calculating the proportion of instances predicted as clickbait that are actually clickbait, equation (2) is the formula for calculating precision.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

Recall

Recall measures the model's ability to correctly identify all actual positive instances, calculating the proportion of actual clickbait labels that were successfully predicted, equation (3) is the formula for calculating recall.

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

F1-Score

The F1-score is an evaluation metric that combines precision and recall. By utilizing the harmonic mean rather than a simple arithmetic average, the F1-score heavily penalizes extreme values, ensuring that the model only achieves a high score if both precision and recall are reasonably well-balanced, equation (4) is the formula for calculating F1-Score.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

Comparative Analysis

The classification output of each method is evaluated against the ground truth labels from the dataset using accuracy, precision, recall, and F1-score. Following this evaluation, an analysis is conducted to compare the performance of each approach. The analysis specifically focuses on a prompting strategy comparison to determine the effect of different prompt types and to identify the most effective inference strategy for clickbait detection. Finally, the results from the LLM approaches are compared with the fine-tuned IndoBERT model to evaluate the performance differences between an inference-based strategy and a fine-tuning approach.

3. RESULTS AND DISCUSSIONS

All experiments were run twice to reduce the risk of coincidence and ensure consistency of the results. Several metrics were used to evaluate the results of each experiment, namely Accuracy (Acc), Macro F1 (Mac F1), Micro F1 (Mic F1), Weighted F1 (W-F1), as well as precision, recall, and F1-score for each class, namely non-clickbait (NC) and clickbait (CB). The evaluation results for each experiment are presented in Table 2 below.

Table 2. LLM experiments result

Metode	Acc	Mac F1	Mic F1	W-F1	NC P	NC R	NC F1	CB P	CB R	CB F1
PLAIN_ZERO_SINGLE	0.82	0.78	0.81	0.80	0.78	0.97	0.86	0.92	0.57	0.71
PLAIN_ZERO_SC	0.82	0.79	0.82	0.81	0.79	0.97	0.87	0.92	0.59	0.72
PLAIN_ZERO_WSC	0.82	0.81	0.82	0.82	0.81	0.92	0.86	0.84	0.68	0.75
PLAIN_ZERO_SELF_REFINE	0.72	0.71	0.72	0.72	0.81	0.70	0.75	0.62	0.75	0.68
PLAIN_FEW_SINGLE	0.83	0.81	0.83	0.83	0.81	0.95	0.87	0.89	0.66	0.76
PLAIN_FEW_SC	0.84	0.83	0.84	0.84	0.84	0.90	0.87	0.82	0.75	0.78
PLAIN_FEW_WSC	0.83	0.82	0.83	0.83	0.85	0.86	0.86	0.78	0.77	0.78
PLAIN_FEW_SELF_REFINE	0.71	0.70	0.70	0.71	0.81	0.67	0.73	0.60	0.76	0.67
LINGUISTIC_ZERO_SINGLE	0.85	0.82	0.85	0.84	0.80	0.99	0.89	0.97	0.63	0.76
LINGUISTIC_ZERO_SC	0.85	0.83	0.85	0.84	0.81	0.99	0.89	0.97	0.64	0.77
LINGUISTIC_ZERO_WSC	0.85	0.83	0.85	0.84	0.81	0.99	0.89	0.97	0.64	0.77
LINGUISTIC_ZERO_SELF_REFINE	0.71	0.70	0.71	0.71	0.80	0.70	0.75	0.61	0.73	0.67
LINGUISTIC_FEW_SINGLE	0.89	0.88	0.89	0.89	0.88	0.96	0.92	0.93	0.79	0.85
LINGUISTIC_FEW_SC	0.90	0.89	0.90	0.90	0.88	0.96	0.92	0.93	0.81	0.86
LINGUISTIC_FEW_WSC	0.89	0.88	0.89	0.89	0.90	0.93	0.91	0.89	0.83	0.85
LINGUISTIC_FEW_SELF_REFINE	0.72	0.71	0.71	0.72	0.82	0.68	0.74	0.61	0.77	0.68

Baseline Evaluation

The plain-zero-single experiment serves as the baseline performance because the model is only given basic instructions to classify titles. Overall, it achieves an accuracy of 0.82 and a Macro-F1 of 0.78. However, the detailed metrics reveal that the model exhibits a conservative bias, meaning it tends to default to the not-clickbait label. Because of this cautious behavior, when the model actually predicts a title as clickbait, it is highly accurate, achieving a high clickbait precision of 0.92. Unfortunately, this also causes the model to miss nearly half of actual clickbait titles, leading to a low clickbait recall of 0.57. Similarly, while it successfully identifies almost all non-clickbait data resulting in not-clickbait recall of 0.97, the substantial number of real clickbait titles mistakenly labeled as not-clickbait lowers the not-clickbait precision to 0.78. The confusion matrices in Figure 7 clearly visualize this behavior, showing a large number of actual clickbait titles incorrectly grouped into the not-clickbait category.

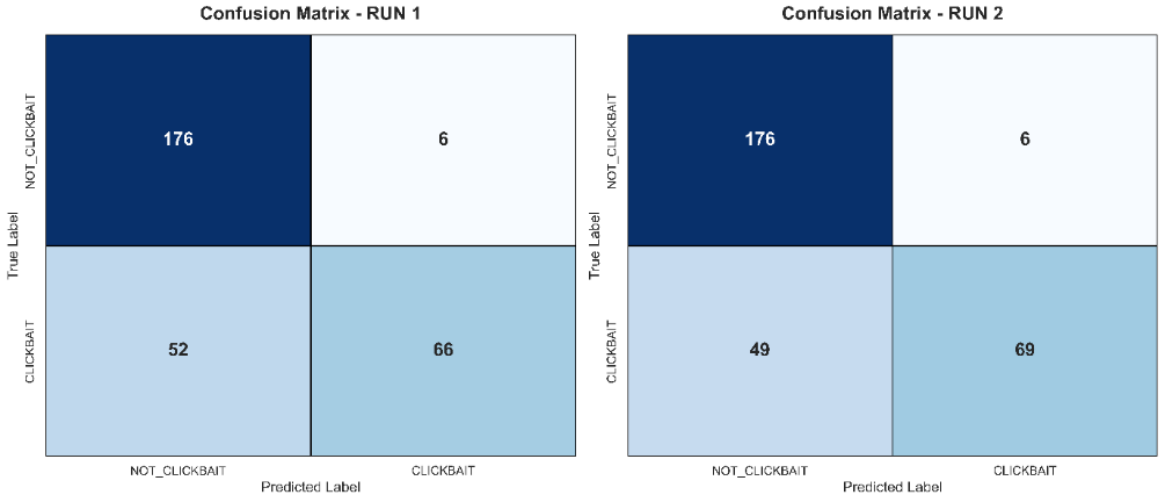


Figure 7. Plain-zero-single confusion matrix

Comparison of Linguistic and Plain Prompts

The linguistic prompt contains classification instructions and linguistic characteristics of clickbait as additional information for the model. The effect of the linguistic prompt on the model's performance in classifying clickbait headlines can be seen by comparing the results of the plain-zero-single experiment with the linguistic-zero-single experiment. This comparison was chosen because both experiments were unaffected by the basic prompting strategy and the inference refinement strategy. A comparison of the accuracy and Macro-F1 of these two experiments is presented in Figure 8 below.

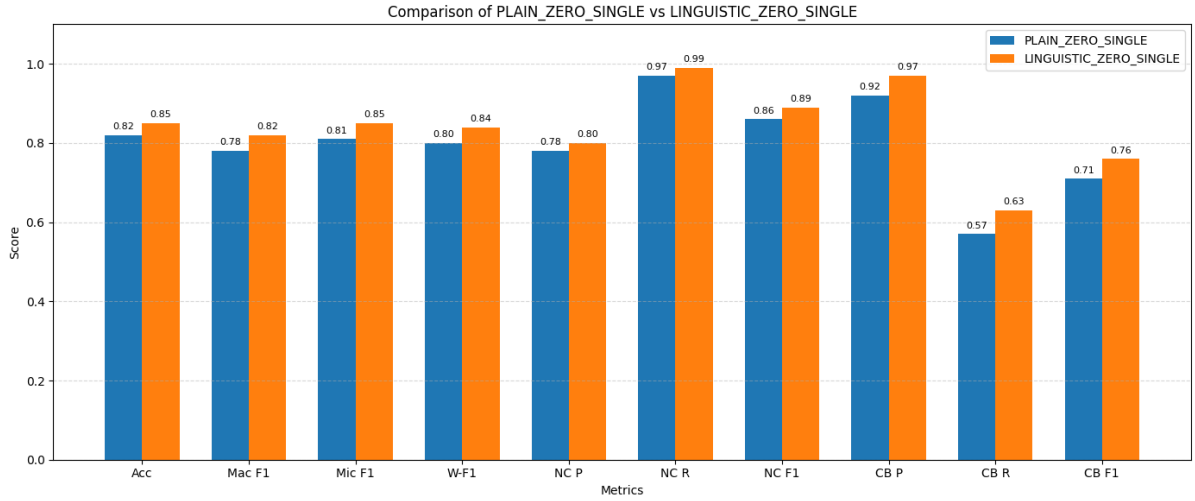


Figure 8. Comparison of plain-zero-single vs linguistic-zero-single

In this experiment, the not-clickbait recall reached 0.99, and the clickbait recall increased to 0.63. Theoretically, providing explicit language rules gives the model clear indicators to look for. This supports recent findings that style-based methods are very effective for short texts like headlines, because the text is too short for the model to verify facts or deep meanings [29]. However, a clickbait recall of 0.63 means a large portion of clickbait remains undetected. Empirically, this shows the limitation of relying only on linguistic rules. These rules are great at catching obvious clickbait, such as titles with exaggerated punctuation. However, they fail to catch subtle clickbait that uses normal sentences but manipulative meanings. Because short-text environments possess limited lexical diversity and high ambiguity, models relying purely on stylistic markers often struggle to capture deeper semantic manipulations or contextual framing [29].

Comparison of zero-shot and few-shot approaches

A comparison of the zero-shot and few-shot approaches was conducted to determine the effect of providing examples on the model's performance in classifying clickbait headlines. The plain-zero-single experiment was compared with the plain-few-single experiment, and then the linguistic-zero-single experiment was compared with the linguistic-few-single experiment. This comparison was conducted to determine how

the provision of example data and the linguistic characteristics of clickbait provide additional information for the model.

As presented in Figure 9, when applying plain prompts, the few-shot approach outperforms the zero-shot approach in accuracy and Macro-F1, with results of 0.83 compared to 0.82 and 0.81 compared to 0.78, respectively. Although the increase in accuracy is not significant, it indicates that providing examples as additional information improves the model's performance in classifying clickbait headlines. The increase in Macro-F1 also indicates that the model can understand the classification patterns of clickbait headlines more balanced. In the clickbait class, the recall value increased significantly from 0.57 to 0.66. The F1-score for the clickbait class also increased from 0.71 to 0.76. This indicates that providing examples can reduce the number of false negatives in the clickbait class.

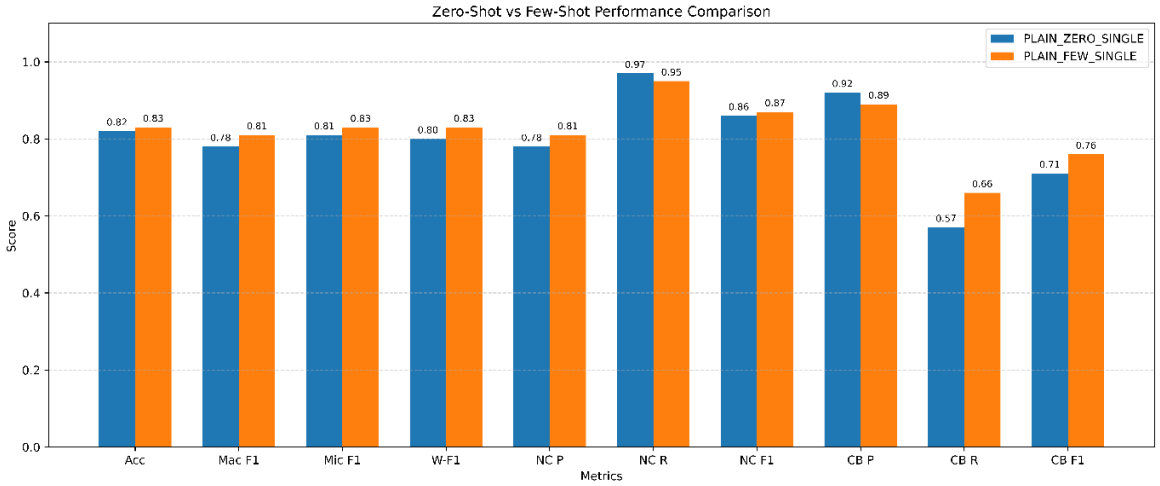


Figure 9. Zero-shot vs few-shot performance comparison

Similarly, the results of the linguistic prompts in Figure 9 show that the few-shot approach also outperformed the zero-shot approach in accuracy and Macro-F1, with results of 0.89 compared to 0.85 and 0.88 compared to 0.82. The recall value in the clickbait class jumped sharply from 0.63 to 0.79, proving that the model has become more reliable in recognizing clickbait data. Theoretically, this indicates that the model successfully utilizes in-context learning by receiving labeled data examples alongside linguistic characteristics, the model maps the specific task pattern without needing to update its underlying weights [24]. Empirically, providing these examples acts as a practical guide that helps the model accurately identify the subtle boundaries between clickbait and non-clickbait. This mechanism effectively reduces the model's uncertainty and lowers the false negative rate, a behavior consistent with recent findings that few-shot demonstrations strongly improve an LLM's classification stability and effectiveness in detecting deceptive texts [30].

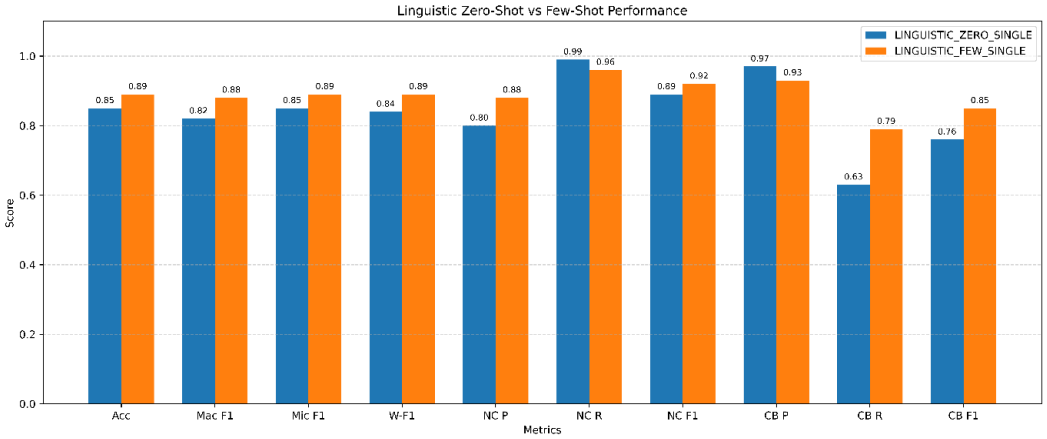


Figure 10. Linguistic zero-shot vs few-shot performance

Comparison of Self-Consistency and Weighted Self-Consistency Approaches

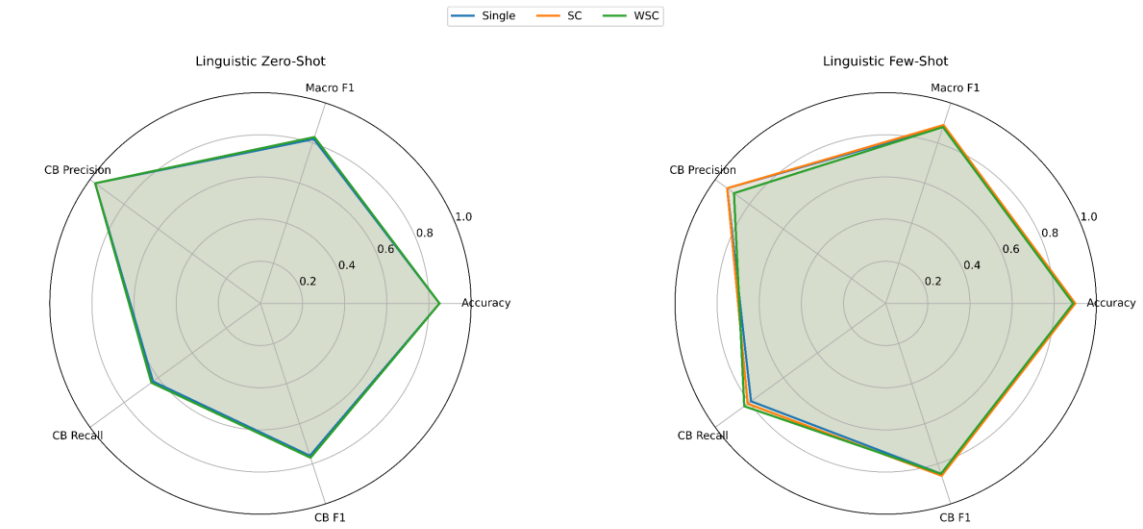
The analysis of the experimental results of the self-consistency and weighted self-consistency approaches will be conducted by comparing several metrics most relevant to the model's ability to classify titles: accuracy, Macro-F1, clickbait precision, clickbait recall, and clickbait f1. To determine whether there is an increase in performance, the self-consistency and weighted self-consistency approaches will be compared with the single inference approach. The results of the plain prompt approach are presented in Figure 11 below.

	Accuracy	Macro F1	CB Precision	CB Recall	CB F1
Single	**0.82**	0.78	**0.92**	0.57	0.71
SC	**0.82**	0.79	**0.92**	0.59	0.72
WSC	**0.82**	**0.81**	0.84	**0.68**	**0.75**

	Accuracy	Macro F1	CB Precision	CB Recall	CB F1
Single	0.83	0.81	**0.89**	0.66	0.76
SC	**0.84**	**0.83**	0.82	0.75	**0.78**
WSC	0.83	0.82	0.78	**0.77**	**0.78**

Figure 11. Performance comparison of single, SC, and WSC for plain prompting

As presented in Figure 11, applying SC and WSC under plain prompting yields accuracy and Macro-F1 scores that are relatively similar to the single inference approach. The most noticeable variation occurs in the WSC approach, which achieves the highest clickbait recall. This indicates that applying weighted voting makes the model more aggressive in recognizing potential clickbait data. However, this comes at the cost of decreased clickbait precision, meaning the WSC strategy increases the false positive rate by more frequently misclassifying legitimate news as clickbait. This results indicate that eventhough SC and WSC are designed to smooth out hallucinations by aggregating diverse reasoning paths, in basic classification tasks lacking explicit linguistic constraints, the aggregated paths may inadvertently amplify the model's bias toward the positive class.



	Accuracy	Macro F1	CB Precision	CB Recall	CB F1
Single	**0.85**	0.82	**0.97**	0.63	0.76
SC	**0.85**	**0.83**	**0.97**	**0.64**	**0.77**
WSC	**0.85**	**0.83**	**0.97**	**0.64**	**0.77**

	Accuracy	Macro F1	CB Precision	CB Recall	CB F1
Single	0.89	0.88	**0.93**	0.79	0.85
SC	**0.9**	**0.89**	**0.93**	0.81	**0.86**
WSC	0.89	0.88	0.89	**0.83**	0.85

Figure 12. Performance comparison of single, SC, and WSC for linguistic prompting

The results for the linguistic prompting approach, shown in Figure 12, show a similar trend. The SC and WSC strategies do not yield a significant performance increase over single inference. In the zero-shot linguistic setting, clickbait recall values remain largely stagnant across all three strategies. Meanwhile, in the few-shot linguistic setting, SC and WSC provide only a marginal increase in clickbait recall. This suggests that when the model is already provided with strong linguistic rules and concrete examples, its reasoning paths become highly deterministic. Consequently, sampling multiple paths yields highly similar outputs, rendering the voting mechanism redundant. This occurs because demonstration labels or explicit rules act as strong anchors that centralize the model's information flow, leaving very little room for diverse reasoning paths in standard classification.

Overall, it can be concluded that self-consistency and weighted self-consistency have a relatively small impact on the model's ability to classify clickbait headlines. A possible explanation is that clickbait detection is fundamentally a text classification task that relies more on recognizing linguistic patterns and persuasive cues than on solving complex multi-step reasoning problems. Self-consistency was originally developed for tasks with a single correct answer and multiple valid reasoning paths, such as arithmetic reasoning, commonsense reasoning, and symbolic reasoning, where aggregating diverse reasoning trajectories can effectively improve prediction accuracy [20]. In contrast, clickbait detection may not benefit substantially from generating multiple reasoning paths because the classification decision is often driven by surface-level semantic and stylistic features rather than complex reasoning chains. This study used five reasoning paths for the self-consistency and weighted self-consistency experiments. Consequently, the resource requirements are at least five times greater than those for a single inference strategy. In other words, the use of self-consistency and weighted self-consistency is suboptimal, as the performance improvements are not proportional to the significantly higher inference costs.

Evaluation of the Self-Refine Method

The analysis of the self-refine approach was conducted to evaluate the model's ability to generate self-feedback and assess its impact on clickbait headline classification performance. The primary evaluation metrics were accuracy, Macro-F1, and clickbait F1, and the performance of all self-refine scenarios was compared against the single inference approach. As shown in Figure 13, the self-refine method consistently reduced the model's performance in classifying clickbait headlines, as indicated by decreases in accuracy, Macro-F1, and

clickbait F1 across all scenario combinations. A possible explanation is that Self-Refine was originally developed to improve outputs through iterative feedback and revision, particularly for generation and reasoning tasks such as dialogue generation, text generation, and mathematical reasoning [21]. In contrast, clickbait detection is a short-text binary classification task with limited contextual information. The model may generate feedback that is redundant or inaccurate, causing subsequent refinements to deviate from an already correct initial prediction. As a result, the self-refinement process may introduce additional noise rather than improving classification performance.

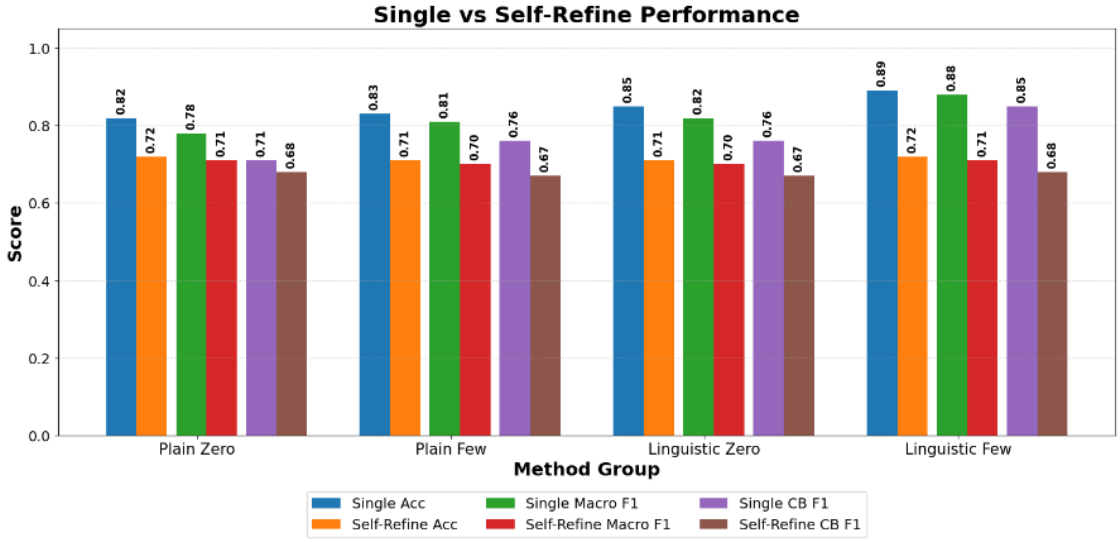


Figure 13. Single inference vs self-refine performance comparison

Evaluation of Fine-Tuned IndoBERT

IndoBERT-p1 model training result

In this study, the IndoBERT-p1 model was trained for ten epochs. The training considered three main metrics which are loss, validation accuracy, and validation F1-score. As can be seen in Figure 14, the training and validation loss graph, both curves experience a consistent decrease from the first to the last epoch. The training loss starts at 0.68 and the validation loss at 0.55 in the first epoch. After that, there is a steady decrease until the training loss reaches 0.27 and the validation loss reaches 0.29 in the last epoch. The parallel decrease in both curves indicates that the model is able to learn from the training data while maintaining its generalization ability to unseen data, thus preventing significant overfitting. Meanwhile, the graph shows an increase in the validation accuracy, F1-score, precision, and recall curves. Validation accuracy starts at 0.75 and then rises steadily to around 0.88-0.89. The F1-score, precision, and recall also converge at around 0.88-0.89. This indicates that the model has the ability to detect both classes equally.

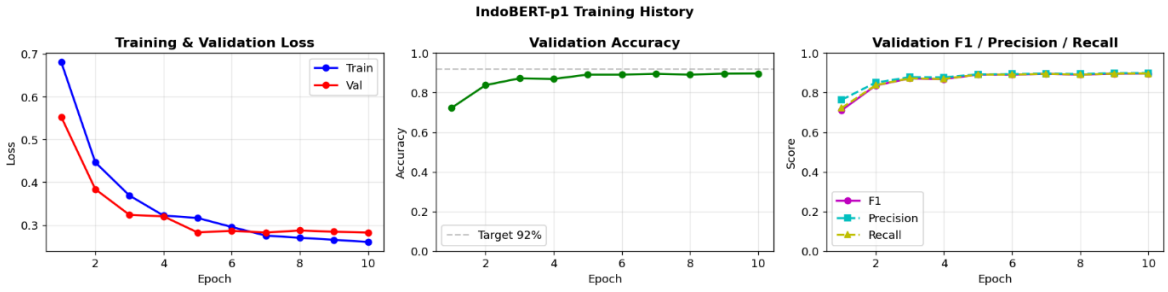


Figure 14. IndoBERT-p1 train result

Evaluation results on unseen set

The final evaluation was conducted using an unseen set of 300 data points (182 non-clickbait and 118 clickbait), utilizing the same samples as the LLM evaluation. As shown in Table 3, the IndoBERT-p1 model achieved an Accuracy and Macro-F1 score of 0.92. This high Macro-F1 score demonstrates the model's ability to equally recognize both classes despite the imbalanced data distribution. For the non-clickbait class, the recall score reached 0.97 (successfully identifying 177 out of 182 samples). Meanwhile, the clickbait class recorded

a high precision of 0.95 but a lower recall of 0.85, with 18 clickbait headlines misclassified as non-clickbait (false negatives). The detailed confusion matrix is presented in Figure 15.

Table 3. Unseen set test result on IndoBERT-p1

Model	Acc	Mac F1	Mic F1	W-F1	NC P	NC R	NC F1	CB P	CB R	CB F1
IndoBERT-p1	0.92	0.92	0.92	0.92	0.91	0.97	0.94	0.95	0.85	0.90

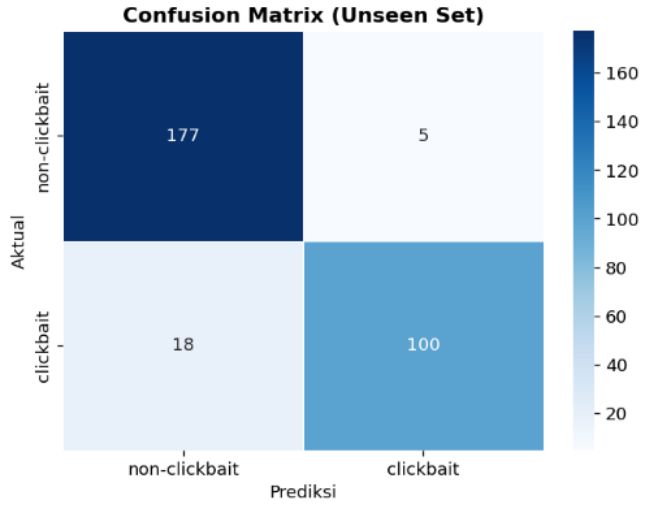


Figure 15. Unseen set confusion matrix

Comparison of LLM and Fine-Tuned IndoBERT Results

To determine how an LLM such as Llama4:latest performs compared to a fine-tuned model, the evaluation results of IndoBERT were compared with the best-performing Llama4:latest scenario, namely linguistic-few-sc. The comparison results for all evaluation metrics are presented in Figure 16. Based on the figure, IndoBERT-p1 outperforms Llama4:latest across all evaluation metrics. IndoBERT-p1 achieved 0.92 for both accuracy and Macro-F1, whereas linguistic-few-sc achieved 0.90 and 0.89, respectively. A notable difference is also observed in clickbait recall, where IndoBERT-p1 achieved 0.85 compared to 0.81 for linguistic-few-sc, indicating that the latter failed to identify a greater number of clickbait headlines. This performance gap may be attributed to the fine-tuning process, which enables IndoBERT-p1 to learn task-specific clickbait patterns from thousands of labeled training examples. Meanwhile, linguistic-few-sc only learns patterns from ten labeled data sets and the linguistic characteristics of clickbait provided in the prompt as additional information.

Furthermore, changes in the patterns of clickbait news headlines over time can cause data drift. IndoBERT-p1 is more vulnerable to these pattern shifts because the model relies on the specific patterns learned during the fine-tuning process. In contrast, some LLMs, particularly those integrated with web access and retrieval systems are more easily able to adapt to changing information and the latest trends compared to models that depend solely on static training data. Nevertheless, LLM models like Llama4:latest still produce competitive performance with 90% accuracy, making them a relevant alternative when labeled data availability and computational resources are limited.

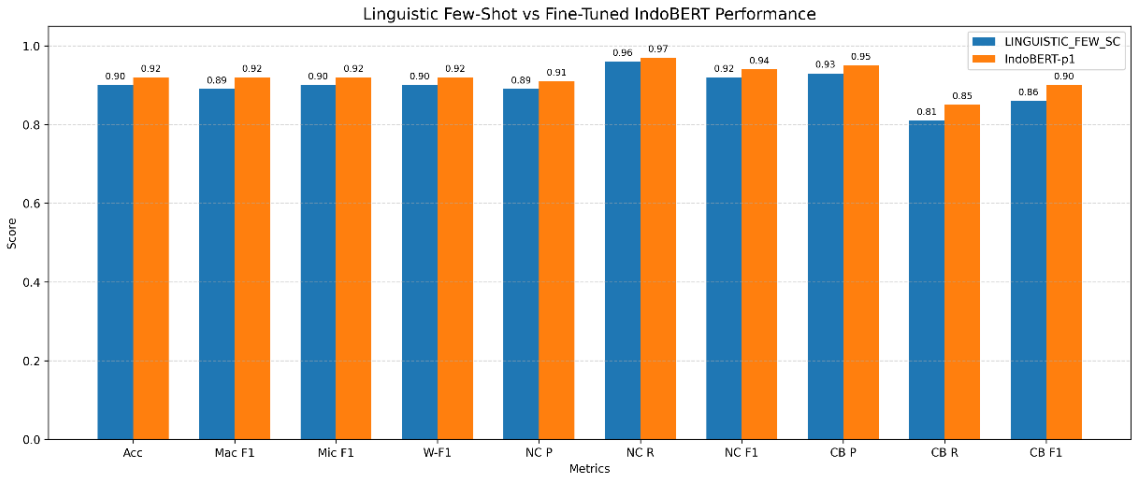


Figure 16. Best LLM scenarios vs IndoBERT performance comparison

4. CONCLUSION

Based on the research results, the prompting approach in the Large Language Model (LLM) can be used for the task of classifying Indonesian clickbait with quite good performance, consistent with the objective of this study to compare prompting strategies for this task. The use of linguistic prompting and few-shot has been proven to provide improved performance compared to plain prompting and zero-shot, with the best combination obtained in the linguistic-few-sc scenario (accuracy of 0.90 and Macro F1-score of 0.89). A key contribution of this study is showing that the benefit of additional reasoning techniques is not always proportional to their cost. The self-consistency and weighted self-consistency methods provide improvements in several metrics, especially clickbait class recall, but the improvements are relatively small compared to the additional inference costs required, while the Self-Refine method was unable to improve overall model performance. The comparison results also show that fine-tuned IndoBERT still produces higher performance than the prompting approach in the LLM, indicating that prompting alone is not yet sufficient to replace task-specific fine-tuning for this task. This study is limited by its evaluation on a relatively small sample of 300 news titles randomly selected from the larger dataset, this sample size was chosen to ensure experimental efficiency given that several prompting strategies required iterative inference on the LLM, as well as due to limited computing resources and time constraints. The study is further limited by its use of a single LLM and a single dataset.

Future research is recommended to address these limitations. A larger-scale evaluation using a larger test dataset and a greater number of samples is needed, drawing from multiple Indonesian clickbait datasets from different sources, supported by more efficient prompting strategies or greater computing resources. Prompt optimization strategies should also be explored along with other LLM architectures that have stronger reasoning capabilities. In addition, robustness analysis of the proposed approach should be conducted across different clickbait datasets. Furthermore, using a hybrid approach between LLM and supervised models could be an alternative for achieving optimal classification performance on the task of detecting Indonesian-language clickbait.

REFERENCES

[1] T. N. Ahmed, N. N. Mustafa, R. K. Ahmed, M. S. Saeed, A. Q. Ali, and K. A. Qadir, "The Impact of Digital Technologies on Journalistic Integrity: An Analysis of Clickbait, Algorithmic Influence and Societal Consequences," *Asian Journal of Education and Social Studies*, vol. 51, no. 6, pp. 566–580, May 2025, doi: 10.9734/ajess/2025/v51i62018.

[2] R. Nurisma and H. Syahrul, "PENGARUH CLICKBAIT JOURNALISM TERHADAP MINAT BACA GENERASI Z NURISMA RAHMATIKA, SYAHRUL HIDAYANTO," 2020.

[3] D. Jácobo-Morales and M. Marino-Jiménez, "Clickbait: Research, challenges and opportunities – A systematic literature review," Oct. 01, 2024, *Bastas*. doi: 10.30935/ojcmt/15267.

[4] S. Mohammada, Mayasari, and T. W. Budhiharti, "Dampak Penggunaan Clickbait pada Judul Berita di Tribunnews.com terhadap Minat Baca Mahasiswa Ilmu Komunikasi Universitas Singaperbangsa Karawang Angkatan 2020," 2024.

[5] K. Scott, "You won't believe what's in this paper! Clickbait, relevance and the curiosity gap," *J. Pragmat.*, vol. 175, pp. 53–66, Apr. 2021, doi: 10.1016/j.pragma.2020.12.023.

[6] D. A. Sayyidina and G. A. Ahmad, "KRIMINALISASI TERHADAP KREATOR YANG MEMBUAT DAN MENGGUNAKAN JUDUL KONTEN UMPAN KLIK (CLICKBAIT) DI MEDIA INTERNET DITINJAU DARI PERSPEKTIF HUKUM PIDANA," 2024.

[7] A. Krivičić, "Clickbait Titles Detection with Machine and Deep Learning Methods," 2023. [Online]. Available: https://urn.nsk.hr/urn:nbn:hr:195:568022

[8] A. Chowanda, Nadia, and L. M. M. Kolbe, "Identifying clickbait in online news using deep learning," *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 3, pp. 1755–1761, Jun. 2023, doi: 10.11591/eei.v12i3.4444.

[9] F. Wei and U. T. Nguyen, "An Attention-Based Neural Network Using Human Semantic Knowledge and Its Application to Clickbait Detection," *IEEE Open Journal of the Computer Society*, vol. 3, pp. 217–232, 2022, doi: 10.1109/OJCS.2022.3213791.
[10] T. Liu, K. Yu, L. Wang, X. Zhang, H. Zhou, and X. Wu, "Clickbait detection on WeChat: A deep model integrating semantic and syntactic information," *Knowl. Based. Syst.*, vol. 245, Jun. 2022, doi: 10.1016/j.knosys.2022.108605.
[11] S. Kurniawan, A. S. Pramayoga, and Y. F. Ashari, "An Ensemble-Based Approach for Detecting Clickbait in Indonesian Online Media," *Jurnal Masyarakat Informatika*, vol. 16, no. 1, pp. 104–118, May 2025, doi: 10.14710/jmasif.16.1.73115.
[12] G. Y. Satriawan and B. Prasetiyo, "Hyperparameter Tuning of Long Short-Term Memory Model for Clickbait Classification in News Headlines," *Recursive Journal of Informatics*, vol. 2, no. 1, pp. 28–36, Mar. 2024, doi: 10.15294/rji.v2i1.71831.
[13] J. Sirusstara, N. Alexander, A. Alfarisy, S. Achmad, and R. Sutoyo, *Clickbait Headline Detection in Indonesian News Sites using Robustly Optimized BERT Pre-training Approach (RoBERTa)*. IEEE, 2022.
[14] M. N. Fakhruzzaman, S. Z. Jannah, R. A. Ningrum, and I. Fahmiyah, "Clickbait Headline Detection in Indonesian News Sites using Multilingual Bidirectional Encoder Representations from Transformers (M-BERT)," Feb. 2021, [Online]. Available: http://arxiv.org/abs/2102.01497
[15] M. E. Syahputra, A. P. Kemala, F. A. Tjan, and R. Susanto, "Clickbait Detection in Indonesia Headline News Using BERT Ensemble Models," in *6th International Seminar on Research of Information Technology and Intelligent Systems, ISRITI 2023 - Proceeding*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 475–479. doi: 10.1109/ISRITI60336.2023.10467417.
[16] G. Ramesh *et al.*, "A review on NLP zero-shot and few-shot learning: methods and applications," Sep. 01, 2025, *Springer Nature*. doi: 10.1007/s42452-025-07225-5.
[17] L. Qin *et al.*, "Large Language Models Meet NLP: A Survey," *Front. Comput. Sci.*, Aug. 2025, doi: 10.1007/s11704-025-50472-3.
[18] Y. Wang, Y. Zhu, Y. Li, L. Wei, Y. Yuan, and J. Qiang, "Multi-modal soft prompt-tuning for Chinese Clickbait Detection," *Neurocomputing*, vol. 614, Jan. 2025, doi: 10.1016/j.neucom.2024.128829.
[19] H. Wang, Y. Zhu, Y. Wang, Y. Li, Y. Yuan, and J. Qiang, "Clickbait Detection via Large Language Models," May 2025, [Online]. Available: http://arxiv.org/abs/2306.09597
[20] X. Wang *et al.*, "Self-Consistency Improves Chain of Thought Reasoning in Language Models," Mar. 2023, [Online]. Available: http://arxiv.org/abs/2203.11171
[21] A. Madaan *et al.*, "SELF-REFINE: Iterative Refinement with Self-Feedback," 2023. [Online]. Available: https://selfrefine.info/
[22] A. William and Y. Sari, "CLICK-ID: A novel dataset for Indonesian clickbait headlines," 2020, doi: 10.17632/k42j7.
[23] T. B. Brown *et al.*, "Language Models are Few-Shot Learners," Jul. 2020, [Online]. Available: http://arxiv.org/abs/2005.14165
[24] Q. Dong *et al.*, "A Survey on In-context Learning," in *EMNLP 2024 - 2024 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, Association for Computational Linguistics (ACL), 2024, pp. 1107–1128. doi: 10.18653/v1/2024.emnlp-main.64.
[25] G. Dugac and T. Altwicker, "Classifying legal interpretations using large language models," *Artif. Intell. Law (Dordr).*, 2025, doi: 10.1007/s10506-025-09447-9.
[26] W. Huang *et al.*, "An empirical study of LLaMA3 quantization: from LLMs to MLLMs," *Visual Intelligence*, vol. 2, no. 1, Dec. 2024, doi: 10.1007/s44267-024-00070-x.
[27] S. Kurniawan, A. S. Pramayoga, Y. F. Ashari, and M. A. Amrustian, "Development and Evaluation of an IndoBERT-Based NLP Model for Automated Clickbait Detection," *Advance Sustainable Science, Engineering and Technology*, vol. 8, no. 1, Nov. 2026, doi: 10.26877/asset.v8i1.2637.
[28] G. Zeng, "Invariance Properties and Evaluation Metrics Derived from the Confusion Matrix in Multiclass Classification," *Mathematics*, vol. 13, no. 16, Aug. 2025, doi: 10.3390/math13162609.
[29] A. Sittar, M. Smiljanic, A. Guček, and M. Grobelnik, "Fake News Detection Through LLM-Driven Text Augmentation Across Media and Languages," *Mach. Learn. Knowl. Extr.*, vol. 8, no. 4, Apr. 2026, doi: 10.3390/make8040103.
[30] F. Trad and A. Chehab, "Prompt Engineering or Fine-Tuning? A Case Study on Phishing Detection with Large Language Models," *Mach. Learn. Knowl. Extr.*, vol. 6, no. 1, pp. 367–384, Mar. 2024, doi: 10.3390/make6010018.

.