BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **25**, No 2
Sofia • 2025 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2025-0012

Achieving Efficient Prompt Engineering in Large Language Models Using a Hybrid and Multi-Objective Optimization Framework

Sridevi Kottapalli Narayanaswamy, Rajanna Muniswamy

Department of Information Science and Engineering, Vemana Institute of Technology, Bengaluru, Karnataka, India
E-mails: sridevi.kn23@gmail.com (*corresponding author*) *mrajanna.ise@gmail.com*

Abstract**:** *Prompt optimization is crucial for enhancing the performance of large language models. Traditional Bayesian Optimization* (*BO*) *methods face challenges such as local refinement limitations, insufficient parameter tuning, and difficulty handling multi-objectives. This study introduces a hybrid multi-objective optimization framework that integrates BO for global exploration and a Genetic Algorithm for fine-tuning prompt hyperparameters using evolutionary techniques. The Non-dominated Sorting Genetic Algorithm II is employed to identify Pareto-optimal solutions, balancing accuracy, efficiency, and interpretability. The framework is evaluated using the GLUE benchmark dataset with BERT-based tokenization for structured input representation. Experimental results demonstrate that the proposed model achieves 95% accuracy, 85% efficiency, and 79% interpretability across three benchmark datasets, outperforming conventional BO-based methods. The findings confirm that the hybrid approach significantly enhances search efficiency, refinement, and multi-objective optimization, leading to more effective and robust prompt optimization.*

Keywords**:** *Bayesian optimization, BERT, Evolutionary algorithm, Large language model, Prompt engineering.*

1. Introduction

A Large Language Model (LLM) is an Artificial Intelligence (AI) program built on deep learning, especially a neural network called a transformer model, trained on the language collected from the Internet. This is also known as the Generative Language Model (GLM) and uses the Generative Pre-trained Transformer model (GPT). Prompt engineering generates and refines prompts to get desired responses from LLM. This also provides an essential guide to LLM for creating useful and relevant user output [1-2]. Prompt engineering can pre-train the LLM and Vision Language Model (VLM). Fine-tuning is the approach by which pre-trained language models are fine-tuned to downstream tasks, optimizing the model for specific necessities [3]. In

recent times, a wide range of language tasks received robust training through LLMs like GPT-3 and ChatGPT [4]. The prompt engineering develops and improves the accuracy and responsiveness of the designed prompt. It includes roles, context, input, output format, and examples [5]. Still, the user struggles to control the output from the LLM, and the traditional approach to prompt crafting becomes time-consuming and inefficient [6].

To analyze their application and adaptability, LLMs have to be combined with Evolutionary Computation (EC) [7]. Since LLMs, such as the GPT series from OpenAI, can understand and generate human language, they are very important in enhancing the power of EC [8]. Most of the approaches used the Bayesian filtering method to estimate the optimal solution and update the components of the probability distribution [9]. Significant progress was achieved by empowering LLMs to exercise foresight and retrace their steps for holistic decision-making with the recently developed Tree of Thoughts (ToT) [10]. Recent advances in effective prompt engineering for LLMs are encouraging, but significant research gaps remain in terms of balancing accuracy, computational cost, and interpretability. To address these challenges, the proposed model presents a hybrid multi-objective optimization framework for prompt engineering. The framework is designed to improve the efficiency of prompt optimization, allowing LLMs to produce even more accurate and interpretable responses with significantly less computational overhead. The major contributions of the work are summarized as follows:

- The research introduces a hybrid optimization framework that integrates a Bayesian Optimization (BO) for efficient exploration and a Genetic Algorithm (GA) for refining high-performing prompts.
- The proposed model uses Bidirectional Encoder Representations from Transformer (BERT-based) tokenizer for text preprocessing and tokenization to ensure proper input segmentation. BO is used for global exploration to improve search efficiency, and EA fine-tunes prompt configuration, which leads to improved optimization outcomes.
- The proposed model uses the Non-dominated Sorting Genetic Algorithm II (NSGA-II) multi-objective optimization technique to achieve a Pareto-optimal solution, balancing accuracy, efficiency, and interpretability, thus identifying the Pareto-optimal prompt solution.

The rest of the paper is organized as follows: Section 2 explores the existing research. Section 3 introduces the Hybrid and Multi-Objective Optimization Framework for Efficient Prompt Engineering in LLMs. Section 4 illustrates the outcomes and outlines the purpose of the discussion. Section 5 presents the paper's conclusion.

2. Literature survey

This section provides related work on prompt engineering in an LLM. K l y u c h n i k o v et al. [11] developed the Neural Architecture Search Benchmark for LLMs. It addresses the reproducibility issues and the high computational cost of neural architecture. However, the large neural network required high-performance

clusters and a complex search space. Zhao et al. [12] proposed prompt tuning for the Metonymy Resolution (PromptMR) to identify the metonymy expression within the sentences, which highlights the limitation of the time-consuming and resource-intensive use of the fine-tuned model. The model struggled to deliver satisfactory results and was not generalized to other NPL tasks. The authors of [13] demonstrated how to solve the Multi-Armed Bandit (MAB) problem in non-stationary situations using BO with LLMs. An LLM-based technique that permits an adjustable ratio of exploration and exploitation. This study needs efficient and developed algorithms or methods to reduce the computational resources.

Ahmed et al. [14] introduced the prompt engineering framework MED-Prompt for medical prediction on clinical notes. Several pre-trained models like BERT, BioBERT, and clinicalBERT were used to guide the process. Furthermore, efficient algorithms and optimization techniques are required to reduce computational requirements and training time. Liu et al. [15] sought to integrate the LLM with the conventional evolutionary search operator to speed up the evolving population's convergence. However, the integrated LLM and EA faced a high computational cost, and future work focused on reducing the time and enhancing the performance accuracy. Sorokin, Safin and Nejati [16] explained the theoretical justification of why Pareto optimization-based testing is insufficient for including failure-inducing regions within a search domain. Henceforth, it uses benchmarking systems to evaluate search-based testing algorithms, which can help to avoid the high cost of system executions. The authors of [17] improved the usability of the Adverse Outcome Pathways Wiki (AOP-Wiki) collection by adjusting its information into a Labelled Property Graph (LPG) schema. This work also used the LLM's generating power to tackle the problem of creating database-specific queries. Henceforward, it enhances the prompt generation methodology to improve accuracy and consistency. The work [18] compared the fine-tuned BERT and LLM to evaluate the applicability and robustness of the intelligent design support systems. It faced issues with handling the short sentences in the dataset. This study needs development in refining the prompting techniques. This study needs efficient and developed algorithms or methods to reduce the computational resources.

Thus, by analyzing the existing works, we observe that the existing works have focused on BO, which results in inefficient search mechanisms. Most of the works were optimized for accuracy by neglecting critical trade-offs like computational efficiency and interpretability. To overcome these limitations, we propose a hybrid multi-objective framework that optimizes global exploration with refined tuning, and the critical optimal trade-offs result in efficient, prompt optimization that balances accuracy, efficiency, and interpretability while reducing computational overhead.

3. Proposed methodology

The proposed methodology introduces a hybrid multi-objective optimization framework for improving prompt engineering in LLMs, as shown in Fig. 1. This structured pipeline begins with collecting the General Language Understanding Evaluation (GLUE), a benchmark dataset comprising six standard NLP datasets. The

model utilizes the MNLI, SST-2, and QQP data from the benchmark. The collected data then undergoes the BERT tokenization, which preprocesses it to structure the raw text into structured input representations. Then, BO performs global search exploration and identifies high-performing prompts, which are fine-tuned by GA using an EA. Finally, a multi-objective optimization algorithm incorporates NSGA-II to optimize the refined prompts and find the Pareto optimal solutions, ensuring an optimal balance between accuracy, efficiency, and interpretability.

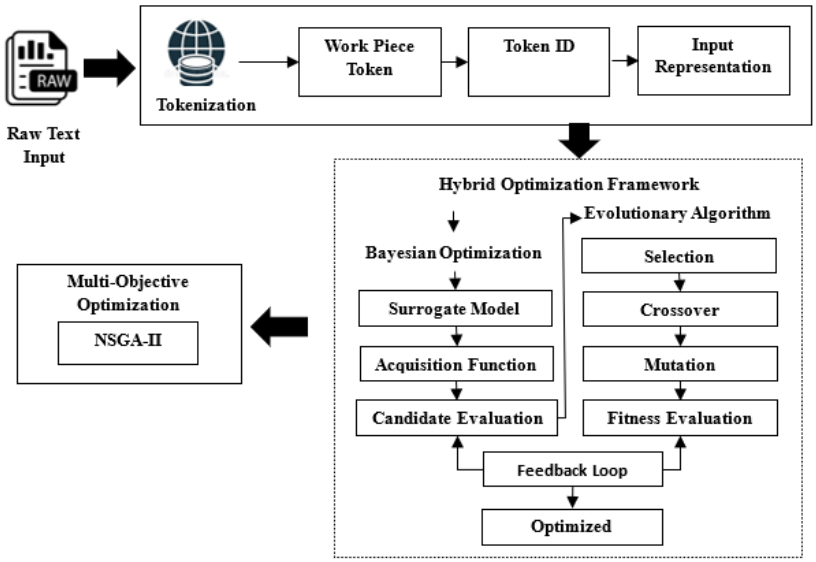

Fig. 1. Workflow of the proposed methodology

3.1. Dataset description

The present study uses a General Language Understanding Evaluation (GLUE) benchmark as a dataset. GLUE [19] is a set of tools for natural understanding system training, improvement, and analysis. It contains six standard datasets related to natural language understanding comprehension, such as textual entailment, question answering, paraphrase identification, and natural language inference. We only employed three of those for our investigation (MNLI, SST-2, and QQP). Here is a quick description of the datasets:

- *Multi-Genre Natural Language Inference* (*MNLI*)

The MNLI is an extensive collection of 433k phrase pairings with textual entailment information annotated, and the dataset enables exploration into a variety of natural language sources from both written and spoken. The task is to determine whether a hypothesis sentence is semantically equivalent (entailment), not semantically equivalent (contradiction), or neither (neutral) based on a given premise.

- *Stanford Sentiment Treebank* (*SST-2*)

This dataset contains 67,000 sentences from movie reviews with sentiment annotations by humans. It only has sentence-level labels and employs the two-way (positive/negative) class split.

- *Quora Question Pairs* (*QQP*)

The QQP dataset contains more than 400,000 lines of possible question-duplicate pairs from the question-answering website Quora. The main objective of this dataset is to ascertain if two questions are entailments. It indicates that the same information can be used to answer them. The statistics measure for these datasets is displayed in Table 1.

Table 1. Statistics of the GLUE benchmark dataset

Dataset	Number of sentences	Number of tokens	Vocabulary size	Average tokens per sentence	Out-of-vocabulary rate
MNLI	848,739	17,989,715	41,701	21.20	0.30%
QQP	903,686	13,287,371	69,796	14.70	0.25%
SST-2	68,569	959,762	7,319	14.00	0.10%

3.2. Tokenization

The process of dividing text into smaller units, known as tokens, which can be words or subwords, is known as tokenization. It is a fundamental operation in NLP that converts raw text into a structured format that can be processed effectively by ML models. The model employs BERT for tokenization since it can capture contextual representations of words to understand their relationships, improving prompt effectiveness. The BERT tokenization process is as follows. Initially, raw text data is pre-processed by WordPiece tokenization to break tokens typically into words or subwords. Special tokens are then added to the tokenized text. As such, a classification token ([CLS]) is added at the beginning of the sequence, and a separator token ([SEP]) is added at the end of the token sequence [20]. Finally, each token is converted to its appropriate ID based on the BERT vocabulary. Additionally, an attention mask is formed while generating the tokenized sequence.

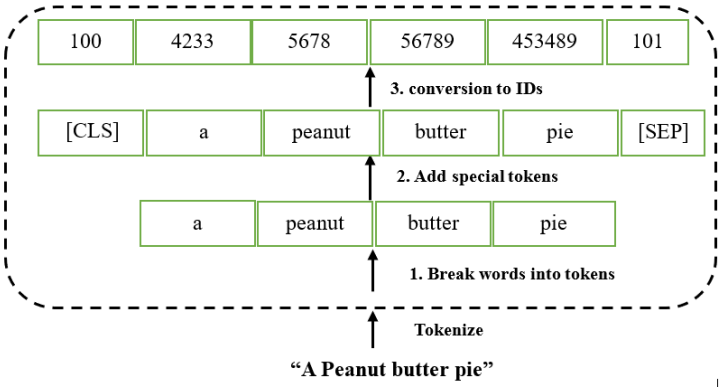

Fig. 2. Process of BERT tokenization

This mask labels the actual tokens to be differentiated from the padding tokens so that only the meaningful input is processed by the model and ignores the padding. Through these processes, BERT tokenization transforms raw text into a structured

and consistent format. The illustration of the tokenization process using BERT is given in Fig. 2.

3.3. Hybrid optimization framework

The hybrid framework combines the BO for global search space exploration and a Genetic Algorithm (GA) for fine-tuning, achieving a balance between exploration and exploitation. The BO identifies high-performance prompts by search space exploration, while the GA refines these solutions with an evolutionary approach to ensure enhanced adaptability and fine-tuned performance [21]. This combination results in enhanced optimization results.

3.3.1. Bayesian optimization

The BO algorithm is an optimization algorithm based on probability that applies to high-dimensional problems with expensive objective functions [22]. It explores the search space efficiently by constructing a surrogate probabilistic model to approximate the objective function. It determines the most informative solution available for the optimizer at the expense of reducing the number of expensive solutions for the objective function. The optimization problem is formulated by the equation

$$b = \text{argmin}_{b \in B} f(b). \tag{1}$$

Here, B represents a search space involved in the optimization process, b is a combination of the hyperparameters in B, and $f(b)$ represents the objective function of the problem. Once the output of each set of new parameters is seen in each iteration, the surrogate model is updated so that it can effectively balance exploration and exploitation.

In this study, BO is used to identify high-performing prompt solutions from a large search space effectively. However, while BO focuses on global exploration, it never performs local refinement. To enhance the optimization of selected prompts, the model employs GA to refine the obtained solutions further. Thus, BO is not only responsible for searching optimal prompts but also for selecting prompts that undergo structured variation through the GA fine-tuned using the crossover and mutation process. This ensures that optimized prompts are computationally efficient and also adaptable across diverse NLP tasks.

3.3.2. Fine-tuning using Genetic Algorithm (GA)

GA is a natural evolutionary process that is referred to as survival of the fittest in evolutionary theory. In this study, GA refines and optimizes prompts by iteratively improving high-performing prompts identified by BO [23]. The search space is represented as a grouping of individuals known as chromosomes. Gene refers to the set of characteristics that identify an individual. To select the most suitable parameters, the fitness of each chromosome is evaluated with the fitness function. To ensure natural selection, the evaluation process uses Mutation and crossover. The best individuals are chosen to progress through crossover, mutation, or selection until a new population is formed. As a result, the optimization problem's solution is determined to be the best member identified.

Here, the EA refines the solution identified by Bayesian Optimization (BO) by examining more localized search areas. After BO has explored the global search space, EA is employed to evolve improved solutions over multiple generations. Bayesian optimization is used for global exploration within the search space, and the EA exploits it by refining solutions through local optimization and evolutionary strategies.

3.4. Fitness function

Each optimized prompt is evaluated based on the texts generated by the corresponding Text LLM. The evaluation process involves comparing the prompt's intended objective with independent NLP classifier predictions. The probability scores for the correct class are employed as the objective value in both optimization and final evaluation. These probability scores are obtained from multiple independent classifiers trained on different datasets. During evaluation, we ensure that redundant or semantically similar generated responses are filtered out to maintain the diversity and effectiveness of the optimized prompts.

Accuracy is measured based on performance metrics on benchmark datasets, as in the equation

$$A(P) = \frac{1}{N}\sum_{i=1}^{N} P_{\text{correct}}(i), \tag{2}$$

where $P_{\text{correct}}(i)$ represents the probability score for the corrected class.

Efficiency is measured as a normalized metric, which includes inference time, FLOating Point operation (FLOPs), and memory consumption, as expressed in the equation

$$E(P) = 1 - \frac{\text{Inference time }(P)}{\max(\text{Inference time})}, \tag{3}$$

where $\text{Inference time }(P)$ represents *the time* taken per query, and $\max(\text{Inference time})$ is the normalization factor ensuring consistency across different configurations. While this provides a relative measure of efficiency, the normalization factors for FLOPs and memory are embedded in the overall efficiency calculation. They are not explicitly separated, making direct mapping to real hardware cost challenging.

Interpretability measures how clear and c*on*cise the prompt depending on the prompt length and readability score, as shown in the equation

$$I(P) = 1 - \frac{\text{Prompt Length }(P)}{\max(\text{Prompt Length})}, \tag{4}$$

where $\text{Prompt Length }(P)$ represents the number of words in the prompt and $\max(\text{Prompt Length})$ ensures normalization.

3.5. Multi-optimization technique

Multi-objective optimization is essential in prompt engineering, where trade-offs exist between accuracy, computational efficiency, and interpretability [24]. In this study, the proposed model employs the NSGA-II to identify Pareto-optimal solutions, ensuring a balanced trade-off between these conflicting objectives. The main aim is to achieve a balance between accuracy, efficiency, and interpretability, guaranteeing an optimal trade-off among these competing factors. NSGA-II preserves these

solutions within the Pareto front so they can be selected according to trade-off preferences. NSGA-II operates based on the concept of non-domination and Pareto optimality.

Pareto selection. The model utilizes the NSGA-II [25] algorithm to rank prompts from the evaluation set F_{eval}, forming the Pareto front, which represents the optimal trade-offs between multiple conflicting objectives. The ideal scenario in prompt engineering would be to identify a single optimal prompt that optimizes all objectives, but that is rarely feasible in practice. Instead, the Pareto optimal selection method provides a set of solutions that are the best possible trade-offs among competing goals.

The NSGA-II uses non-dominated sorting to rank prompts based on their performance across the objective front. A prompt c is non-dominated if no other prompt d exists, as expressed in the equation

$$f_i(d) \geq f_i(c) \text{ and } \exists k, f_k(d) > f_k(c), \tag{5}$$

where f_i represents the objective functions. This method selects solutions that are not necessarily optimal for all objectives but are optimal for the inherent trade-offs.

In addition to selecting the top-n solutions ranked by NSGA-II, we also include the top-n-performing solutions from each objective that were excluded from the Pareto ranking. This inclusion is based on the assumption that highly specific objective solutions can contribute useful features in the next generation during genetic operations like combination. By combining these solutions, we ensure that different prompt structures contribute to the evolutionary process, resulting in more efficient and adaptive prompts.

4. Results and discussion

This section presents the performance of the hybrid and multi-objective optimization for efficient, prompt engineering in LLMs. The model has been implemented on the Python 3.12 platform with Windows 10, Intel(R) Xeon(R) CPU E5-1650 v3 @ 3.50GHz, memory 32.0 GB, Graphics = NVIDIA Quadro M2000, and Visual Studio Code – v1.86.

4.1. Performance evaluation of the multi-objective optimization

Fig. 3 depicts the performance of the multi-objective optimization in terms of three criteria such as accuracy, interpretability, and efficiency. The NSGA-II algorithm obtains an effective accuracy of 95% in finding near-optimal solutions to the multi-objective problem. The proposed model demonstrates the effective use of computational resources, providing quality solutions with an efficiency rate of 85% and an interpretability of 79%.

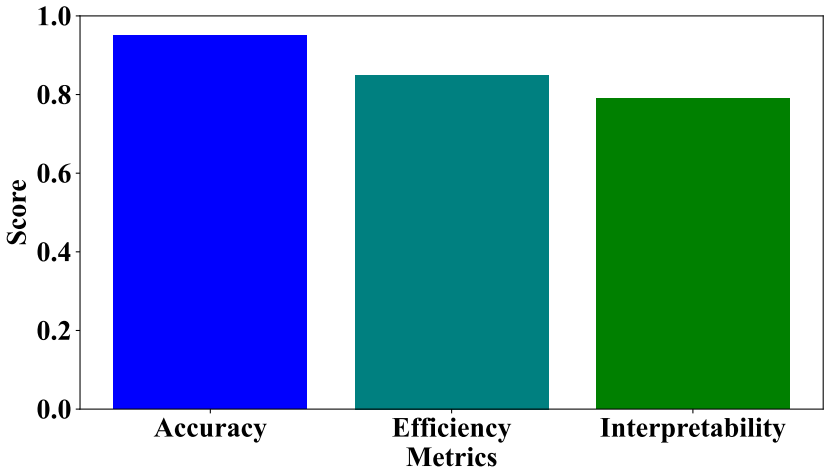

Fig. 3. Performance metrics of the NSGA-II Algorithm

4.1.1. Trade-Off Visualization of NSGA-II Algorithm

Table 2 provides the performance evaluation of the NSGA-II Algorithm across three tasks (i.e., SST-2, MNLI, and QQP) and clusters (Cluster 1, 2, and 3). Since NSGA-II is a multi-objective optimization algorithm, it does not produce a single best solution; instead, it generates a set of Pareto-optimal solutions, which are grouped into clusters, where each cluster represents an optimized prompt configuration with a unique balance of accuracy, efficiency, and interpretability. Notably, QQP Cluster 2 has the best interpretability (0.89) and the fastest processing time (95 ms/query), in addition to its excellent accuracy of 98.8%. In comparison to other clusters, SST-2 Cluster 3 and QQP Cluster 3 have the best accuracy (99% and 99.5%, respectively), good interpretability scores, and somewhat slower processing times. MNLI clusters typically perform somewhat worse on all parameters; MNLI Cluster 3 has the slowest efficiency and the lowest accuracy.

Table 2. Performance analysis of the NSGA-II algorithm in the dataset

Task	Cluster-ID	Accuracy (%)	Efficiency (%)	Interpretability (%)
SST-2	Cluster 1	98.5	110	88
	Cluster 2	97.2	105	85
	Cluster 3	99	115	86
MNLI	Cluster 1	96.8	120	82
	Cluster 2	97.5	110	8
	Cluster 3	95.9	125	83
QQP	Cluster 1	99.2	100	87
	Cluster 2	98.8	95	89
	Cluster 3	99.5	110	88

Fig. 4 shows the 3D visualization of the multi-objective optimization. It shows the effect of dynamic refinement on performance.

Fig. 4. 3D visualization of the multi-objective optimization

4.1.2. Multi-objective trades off with Pareto front

The Pareto front is obtained from the multi-objective optimization process. Fig. 5 shows the trade-off among multiple conflicting objectives. In this graph, objective 1 and objective 2 are the optimal solutions, and the Pareto front represents the red line with a circular marker that contains the optimal solution. These solutions are non-dominated, which means improved objective results. The green and blue points in the graph are the optimal solutions obtained by the NSGA-II Algorithm. It highlights the boundary of achievable trade-offs and makes an efficient decision-making tool in multi-objective optimization problems.

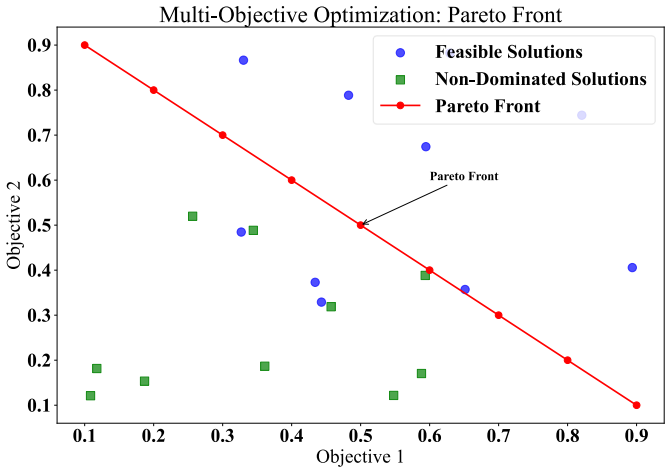

Fig. 5. Pareto graph obtained from the NSGA-II Algorithm

4.1.3. Convergence analysis of hybrid framework

Fig. 6 presents the convergence of loss during the optimization iteration for BO and GA. It highlighted the efficiency of the hybrid framework in reaching the optimal solution.

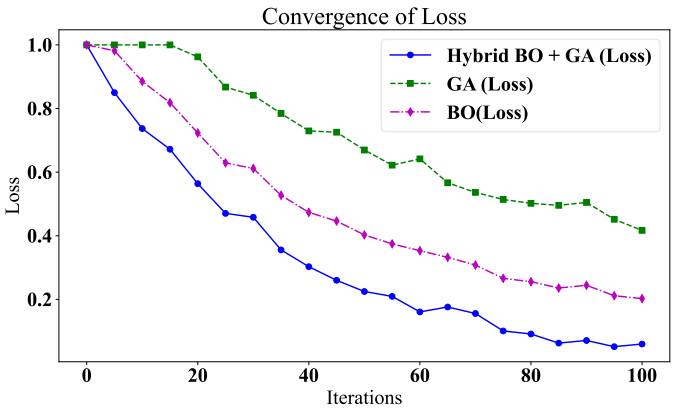

Fig. 6. Convergences of loss

4.2. Result of the proposed model's performance across different datasets

To assess the generalizability of the model, we validate our proposed model with another external dataset, which involves more complex reasoning tasks and longer prompts that were not part of the training dataset. We use the GSM8K dataset from Kaggle as an external validation dataset. The GSM8K Linguistically Diverse Training & Test Set consists of 8000 questions and answers that have been created to simulate real-world scenarios in grade school mathematics. Each question is paired with one answer based on a comprehensive test set. The questions cover topics such as algebra, arithmetic, probability, and more. The proposed model evaluates its performance metrics on the GLUE dataset and the external dataset. We compare the performance metrics of our proposed model datasets with this external dataset to obtain the model's generalizability.

Table 3. Generalization performance of the proposed model: Metrics

Metrics	GLUE Dataset	GSM8K Dataset
Accuracy (%)	95	94
Efficiency (%)	85	81
Interpretability (%)	79	77

Table 3 demonstrates that the model's performance on the GSM8K dataset [26] was not significantly different from the GLUE datasets [19]. It shows that our proposed model is better generalized across various datasets. The successful validation of our model using the GSM8K dataset (external dataset) highlights its potential as a reliable tool for prompt optimization.

4.3. Comparison of prompt length

The proposed model inserts the sentences from the datasets SST-2, MNLI, and QQP into the initial prompt and optimized prompt. It showed how the prompts were adapted to different data, with the optimized prompts providing a clearer and more effective way to phrase the task for the mode. If the optimized prompts were either shorter or more concise, it would efficiently reduce the complexity and computational cost of LLM.

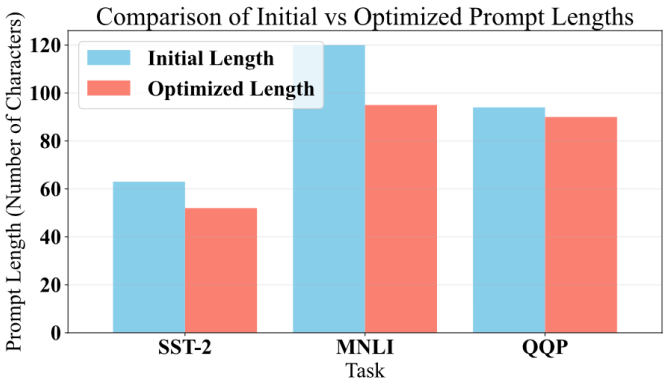

Fig. 7. Comparison of initial prompt Vs optimized prompt

Fig. 7 illustrates the comparison of the initial prompt vs the optimized prompt with three tasks. In this graph, the optimized prompt is shorter than the initial prompt. So, it is proven that our model efficiently reduces the complexity and computational cost.

4.4. Comparison with Baseline approaches

This section compares the performance of the proposed hybrid approach with standalone baseline models.

Table 4. Comparison analysis of the proposed model with baseline approaches

Techniques	Accuracy (%)	Efficiency (%)	Interpretability (%)	Convergence speed (s)
Bayesian Optimization for Prompt Engineering (BOPE)	87	80	75	500
Prompt bench	88	78	72	520
Optimization by PROmpting (OPRO)	89	79	74	480
Evolutionary Prompt Search (EPS)	90	81	77	470
Evolutionary optimizer	93	82	76	460
RL-based proximal policy Optimization (PPO)	92	83	78	450
BO+GA+NSGA-II (Proposed)	95	85	79	410

Table 4 illustrates the performance of different prompt optimization methods across four key metrics, including accuracy, efficiency, interpretability, and convergence rate. Traditional techniques such as BOPE and Prompt Bench provide good baselines with moderate results in all measures, while OPRO utilizes a learned model to improve slightly over prompt generation. Evolutionary approaches such as Evolutionary Prompt Search and LLM as an Evolutionary Optimizer show consistent improvements across all domains, especially interpretability and convergence. PPO-based reinforcement learning achieves a robust equilibrium with quicker convergence and improved overall scores. Specifically, the proposed hybrid approach combining BO, GA, and NSGA-II outperforms others by offering the highest accuracy (95%), efficiency (85%), and interpretability (79%) with the fastest convergence rate (410 s). These outcomes demonstrate the effectiveness of combining global search,

local tuning, and multi-objective optimization in generating high-quality, well-balanced prompts effectively.

4.5. Ablation study

The ablation study evaluates the model's performance with and without the hybrid optimization algorithms on three metrics: accuracy, efficiency, and interpretability. Table 5 shows the difference between the three metrics with and without hybrid optimization. With a hybrid optimization algorithm: The method with all components (BO, GA, and NSGA-II) achieves a very high accuracy of 95% with 85% efficiency and 79% interpretability. Without BO, there is a slight drop in the model's three metrics. Achieves 90% accuracy with 75% efficiency and 72% interpretability. Without GA: Removing the GA leads to a significant drop in accuracy (from 95% to 85%). The efficiency improves slightly (from 75% to 80%). The interpretability score drops to 70%. Without NSGA-II: without multi-objective optimization, NSGA-II leads to a decrease in accuracy of 88%, with 78% efficiency and 74% interpretability. Fig. 8 shows the bar graph representation of the ablation study.

Table 5. Comparison of metrics with and without proposed optimization algorithms

Components	Accuracy	Efficiency	Interpretability
BO+GA+NSGA-II (Proposed)	95%	85%	79%
Without BO	90%	75%	72%
Without GA	85%	80%	70%
Without NSGA-II	88%	78%	74%

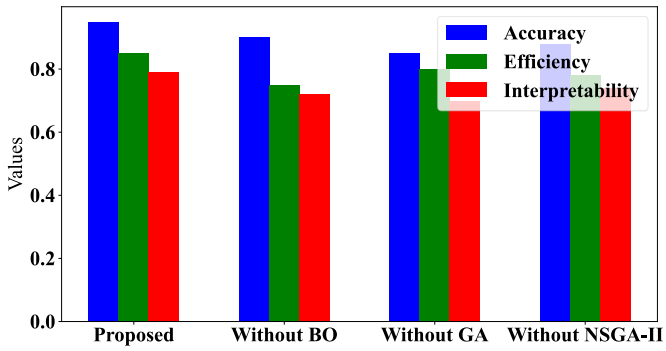

Fig. 8. Performance metrics of the ablation study evaluation

4.6. Statistical test

The proposed model utilizes the Wilcoxon signed-rank test to strictly examine the statistical significance of the observed difference between the proposed hybrid method and its ablated variants. The obtained P-value serves as a key assessment measure of the importance of the model performance disparities among the compared models. Moreover, Table 6 presents the results of the Wilcoxon signed rank test, thus reinforcing the differences observed.

Table 6. Wilcoxon signed-rank test results

Comparison	W-statistic	P-value	Sample size	Significant level (α)
Proposed vs Without BO	1	0.043	5	0.05
Proposed vs Without GA	1	0.031	5	0.05
Proposed vs Without NSGA-II	2	0.078	5	0.05

4.7. Discussion

In this study, we evaluated the performance of the NSGA-II Algorithm in optimizing multi-objective optimization problems with prompt optimization in LLMs. The proposed hybrid optimization framework integrating BO, GA, and NSGA-II attains 95% accuracy, 85% efficiency, and 79% interpretability over benchmark datasets. The Pareto front analysis (Fig. 5) illustrates that the proposed approach effectively balances accuracy, efficiency, and interpretability, resulting in an optimal solution. The combination of BO and GA improves the search process and hyperparameter optimization, resulting in an efficient prompt optimization process. Prompt length optimization (Fig. 7) significantly minimizes the consumption of resources by producing shorter and more efficient prompts without any loss of performance. Additionally, the ablation study (Fig. 8) identifies the importance of each component in the proposed approach. Finally, the pie chart analysis in Fig. 8 visually represents the relative contribution of each optimization component (BO, GA, and NSGA-II) to the final performance improvements.

5. Conclusion

The study introduced a hybrid multi-objective optimization framework to enhance prompt engineering in LLMs through the combination of BO, GA, and NSGA-II. Compared to traditional methods that are based on BO, the proposed approach enhances the search process, improves hyperparameter tuning, and optimizes trade-offs. It balances accuracy, efficiency, and interpretability effectively. In various experimental validations on the GLUE benchmark dataset, our proposed hybrid framework achieved an accuracy of 95%, an efficiency score of 85%, and an interpretability score of 79%, better than the performance of other approaches. In addition, the Pareto front analysis and trade-off visualizations provide a valuable decision-making tool for decision-makers to understand the attainable trade-offs between conflicting objectives. In future work, we will plan to incorporate user-centric evaluations to validate the interpretability of optimized prompts through human studies, ensuring that improvements align with actual human preferences and perceptions.

Statements and Declarations

***Author Contribution*:** All authors contributed to the conception of the problem setting and overall design of the work. T.S. and B.S. built the conceptualization and methodology. T.S. and G.S. implemented the work. Validation was performed by B.S. and G.S., and writing was done by T.S., B.S., and G.S. This version was revised and improved by all authors, who also read and approved the final manuscript.
***Funding*:** No funding was received for conducting this study.
***Conflict of interest*:** The authors declare that they have no conflict of interest.

Availability of data and materials: The dataset is available in the Kaggle repository
***Ethical approval*:** The research is original, and all the figures and tables are created by the authors of this manuscript.
***Consent for publication*:** All authors agree with the submission of the manuscript to this journal and possible publication afterwards.

References

1. Pornprasit, Chanathip, C. Tantithamthavorn. Fine-Tuning and Prompt Engineering for Large Language Models-Based Code Review Automation. – Information and Software Technology, Vol. **175**, 2024, 107523.
2. Heston, T. F., C. Khun. Prompt Engineering in Medical Education. – International Medical Education, Vol. **2**, 2023, No 3, pp. 198-205.
3. He, X., S. Zannettou, Y. Shen, Y. Zhang. You only Prompt Once: On the Capabilities of Prompt Learning on Large Language Models to Tackle Toxic Content. – In: 2024 IEEE Symposium on Security and Privacy (SP'24), 2024, pp. 770-787.
4. Sabbatella, A., A. Ponti, I. Giordani, A. Candelieri, F. Archetti. Prompt Optimization in Large Language Models. – Mathematics, Vol. **12**, 2024, No 6, p. 929.
5. Song, Y. F., Y. Q. He, X. F. Zhao, H. L. Gu, D. Jiang, H. J. Yang, L. X. Fan. A Communication Theory Perspective on Prompting Engineering Methods for Large Language Models. – Journal of Computer Science and Technology, Vol. **39**, 2024, No 4, pp. 984-1004.
6. Knoth, N., A. Tolzin, A. Janson, J. M. Leimeister. AI Literacy and Its Implications for Prompt Engineering Strategies. – Computers and Education: Artificial Intelligence, Vol. **6**, 2024, 100225.
7. Liu, S., X. Chen, Qu, K. Tang, Y. S. Ong. Large Language Models as Evolutionary Optimizers. – In: 2024 IEEE Congress on Evolutionary Computation (CEC'24), June 2024, pp. 1-8.
8. He, C., Y. Tian, Z. Lu. Artificial Evolutionary Intelligence (AEI): Evolutionary Computation Evolves with Large Language Models. – Journal of Membrane Computing, 2024, pp. 1-18.
9. Patania, S., E. Masiero, L. Brini, V. Piskovskyi, D. Ognibene, G. Donabauer, U. Kruschwitz. Large Language Models as an Active Bayesian Filter: Information Acquisition and Integration. – In: Proc. of 28th Workshop on the Semantics and Pragmatics of Dialogue, September 2024.
10. Chen, S., W. Wang, X. Chen, M. Zhang, P. Lu, X. Li, Y. Du. Enhancing Chinese Comprehension and Reasoning for Large Language Models: An Efficient LoRA Fine-Tuning and Tree of Thoughts Framework. – Journal of Supercomputing, Vol. **81**, 2025, No 1, p. 50.
11. Klyuchnikov, N., I. Trofimov, E. Artemova, M. Salnikov, M. Fedorov, A. Filippov, E. Burnaev. Nas-Bench-Nlp: Neural Architecture Search Benchmark for Natural Language Processing. – IEEE Access, Vol. **10,** 2022, pp. 45736-45747.
12. Zhao, B., W. Jin, Y. Zhang, S. Huang, G. Yang. Prompt Learning for Metonymy Resolution: Enhancing Performance with Internal Prior Knowledge of Pre-Trained Language Models. – Knowledge-Based Systems, Vol. **279**, 2023, 110928.
13. De Curtò, J., I. de Zarzà, G. Roig, J. C. Cano, P. Manzoni, C. T. Calafate. Llm-Informed Multi-Armed Bandit Strategies for Non-Stationary Environments. – Electronics, Vol. **12**, 2023, No 13, 2814.
14. Ahmed, A., X. Zeng, R. Xi, M. Hou, S. A. Shah. MED-Prompt: A Novel Prompt Engineering Framework for Medicine Prediction on Free-Text Clinical Notes. – Journal of King Saud University-Computer and Information Sciences, Vol. **36**, 2024, No 2, 101933.
15. Liu, S., C. Chen, X. Qu, K. Tang, Y. S. Ong. Large Language Models as Evolutionary Optimizers. – In: Proc. of IEEE Congress on Evolutionary Computation (CEC'24), June 2024, pp. 1-8.
16. Sorokin, L., D. Safin, Sh. Nejati. Can Search-Based Testing with Pareto Optimization Effectively Cover Failure-Revealing Test Inputs? – Empirical Software Engineering, Vol. **30**, 2025, No 1, pp. 1-39.

17. K u m a r, S., D. D e e p i k a, K. S l a t e r, V. K u m a r. AOPWIKI-EXPLORER: An Interactive Graph-Based Query Engine Leveraging Large Language Models. – Computational Toxicology, Vol. **30**, 2024, 100308.
18. Q i u, Y., Y. J i n. ChatGPT and Finetuned BERT: A Comparative Study for Developing Intelligent Design Support Systems. – Intelligent Systems with Applications, Vol. **21**, 2024, 200308.
19. GLUE Dataset. **https://www.kaggle.com/datasets/thedevastator/nli-dataset-for-sentence-understanding**
20. S o n i, U., D. A. G. G o r d h a n J e t h a v a, A. G a n a t r a. Latest Advancements in Credit Risk Assessment with Machine Learning and Deep Learning Techniques. – Cybernetics and Information Technologies, Vol. **24**, 2024, No 4, pp. 22-44.
21. N g o, V. B., V. H. V u. Multi-Level Machine Learning Model to Improve the Effectiveness of Predicting Customer Churn in Banks. – Cybernetics and Information Technologies, Vol. **24**, 2024, No 3, pp. 3-20.
22. V i n c e n t, A. M., P. J i d e s h. An Improved Hyperparameter Optimization Framework for AutoML Systems Using Evolutionary Algorithms. – Scientific Reports, Vol. **13**, 2023, No 1, 4737.
23. B a k ı r, H., Ö. C e v i z. Empirical Enhancement of Intrusion Detection Systems: A Comprehensive Approach with Genetic Algorithm-Based Hyperparameter Tuning and Hybrid Feature Selection. – Arabian Journal for Science and Engineering, Vol. **49**, 2024, No 9, pp. 13025-13043.
24. A l S a b a, M. T., N. A. H a k a m i, K. S. A l J e b r e e n, M. A. A b i d o. Multi-Objective Distributionally Robust Approach for Optimal Location of Renewable Energy Sources. – Alexandria Engineering Journal, Vol. **77**, 2023, pp. 75-94.
25. H a r a n e, P. P., D. R. U n u n e, R. A h m e d, S. W o j c i e c h o w s k i. Multi-Objective Optimization for Electric Discharge Drilling of Waspaloy: A Comparative Analysis of NSGA-II, MOGA, MOGWO, and MOPSO. – Alexandria Engineering Journal, Vol. **99**, 2024, pp. 1-16.
26. GSM8K Dataset Link. **https://www.kaggle.com/datasets/thedevastator/grade-school-math-8k-q-a**

Received: 11.03.2025, Revised version: 17.04.2025, Accepted: 04.05.2025