A Methodology for Prompt Engineering in Engineering: Framework Development and Performance Evaluation

Valeria Vieira dos Santos [1], Miroslav Malaga [2]

[1] Federal University of São Carlos (UFSCar), São Carlos – SP, Brazil
valeriavieira@estudante.ufscar.br

[2] University of West Bohemia, Pilsen – Czech Republic
malaga@fst.zcu.cz

Annotation: The growing integration of Artificial Intelligence in engineering faces the challenge of response effectiveness and dependence on the quality of user prompts. This article addresses this issue by introducing a systematic framework for prompt engineering, composed of five essential elements: specificity, contextualization, formatting, audience alignment, and constraint definition. The methodology validated the framework through case studies and a comparative evaluation of four LLMs (ChatGPT, DeepSeek, Gemini, MetaAI). The results demonstrate that structured prompts increase output accuracy and reveal significant performance differences among models, with ChatGPT and DeepSeek showing superiority in conceptual tasks. It is concluded that prompt engineering is a fundamental competence, and its maximum effectiveness is achieved through a robust methodology and the careful selection of the AI model, ensuring greater reliability in technical applications.

1 Introduction

The integration of generative Artificial Intelligence (AI) drives transformative advances in engineering, automating tasks ranging from complex design to predictive analyses in mechanical, civil, and aerospace engineering fields. However, this potential is harnessed by a critical interface that determines the quality and relevance of the generated outputs: the prompt.

In this context, prompt engineering emerges as a key competence in this new scenario [1,2,3]. Defined as the practice of structuring instructions to translate human intent into precise computational tasks, its correct application is essential. Poorly defined prompts can lead to critical errors in applications such as structural analysis. In contrast, well-formulated instructions improve model interpretability, reduce debugging cycles, and ensure compliance of the results with engineering workflows [4,5].

Despite its growing relevance, the academic literature lacks a formal structure for developing and evaluating prompts tailored to complex technical tasks. Although helpful, general guidelines on clarity and specificity prove insufficient

https://doi.org/10.24132/PI.2025.13287.169-178

for the rigour and compliance required in engineering domains, which hinders the reliable integration of AI tools into critical processes.

It is therefore essential to further strengthen the theoretical contribution of this field by connecting prompt engineering to established frameworks in human–computer interaction (HCI), where structured interaction design principles have long been shown to reduce ambiguity and cognitive load [4,6,7], to engineering education, where structured scaffolding enhances conceptual retention and learning outcomes [8,9,10], and to engineering standards and regulatory frameworks, such as Eurocode, ISO, and IEEE, which define the practical and safety constraints within which AI must operate [11,12,13,14].

This article addresses this gap by introducing a systematic approach to prompt engineering, specifically designed for engineering disciplines. Our proposal builds on established theories of human–computer interaction (HCI) and task decomposition [4] to create a practical framework. The goal is to align AI-generated results with rigorous academic and industrial standards [6], transforming AI's potential into a reliable tool for technological advancement.

2 Methodology

This study was conducted through a three-stage sequential methodology, designed to develop, apply, and empirically validate a prompt engineering framework for technical contexts.

The first stage consisted of a targeted literature review to identify prompt design principles in adjacent domains, such as Human–Computer Interaction (HCI), cognitive science, and engineering education [6,8]. The objective was to extract consolidated concepts on how humans formulate problems and how computational systems interpret instructions. Based on this theoretical synthesis, a structured framework for prompt engineering was developed, composed of five fundamental elements:

1. Task Specificity: Unambiguous definition of the objective, scope, and expected deliverables.
2. Contextualization: Provision of background information, such as technical standards, design variables, or relevant input data.
3. Output Formatting: Explicit instructions on the desired response format (e.g., table, code, list, report).
4. Audience Alignment: Definition of the level of detail and technical jargon expected in the response, according to the target audience (e.g., a specialist vs. a student).
5. Constraint Definition: Imposition of limits and conditions that the response must respect (e.g., "use only ISO 8601 standard," "do not exceed 200 words").

The framework was applied to case studies across various engineering domains (mechanical, civil, electrical, software, chemical, and aerospace) in the second stage. For each case, pairs of prompts were created: one “ineffective” (generic and ambiguous) and one “effective” (structured according to the five framework elements). The contrast between the obtained results demonstrated the practical applicability of the framework and its usefulness in diagnosing communication failures between the engineer and the AI, illustrating how its adoption improves both the accuracy and the relevance of responses in realistic workflows.

The final stage consisted of a quantitative experimental comparison among four large language models (LLMs): ChatGPT (GPT-4), Gemini, DeepSeek, and MetaAI. A set of 20 prompts, covering conceptual tasks (e.g., explaining a metallurgical process) and computational tasks (e.g., generating Python code for numerical integration), was submitted to each model. The responses were evaluated on a 0–5 scale based on five rigorously defined criteria:

- Accuracy: Evaluates the factual correctness and technical precision of the information. For quantitative problems, it refers to the exactness of the numerical result. For conceptual questions, it measures the alignment of the response with scientific principles and established domain knowledge.
- Completeness: Measures the comprehensiveness of the response. A high score indicates that all aspects of the question were addressed, with no omissions of critical information or essential details for a complete understanding of the topic.
- Clarity & Coherence: Analyses the quality of presentation. It evaluates whether the response is well-structured, uses precise and unambiguous language, and whether the reasoning flows logically and is easy to follow.
- Confidence & Justification: Assesses the model’s ability to substantiate its answers. Higher scores were assigned when the model provided the correct answer and explained the reasoning, steps, or principles used to reach it.
- Problem-Solving Ability: A criterion applied mainly to computational tasks. It measures the model’s ability to generate a functional and practical solution, such as an executable code snippet or a step-by-step procedure that effectively solves the proposed problem.

Combining qualitative analysis of the case studies with quantitative validation from the experiments, this mixed approach enabled a robust and multifaceted evaluation of the proposed framework.

3 Discussion

The findings demonstrate that the proposed framework for prompt engineering enhances the reliability of AI outputs in engineering workflows. Case studies show that when prompts specify context, parameters, and constraints, the resulting outputs align more closely with professional standards and reduce

iteration cycles, a finding consistent with other research on optimising AI-driven workflows [5].

The comparative evaluation of four LLMs confirms that model choice also plays a critical role: while ChatGPT and DeepSeek consistently delivered accurate and well-justified responses, Gemini and MetaAI exhibited clarity and contextual precision limitations. This suggests that prompt engineering alone is insufficient; engineers must also critically select and validate the AI system, reinforcing the conclusions of broader evaluations on LLM effectiveness [2].

From a practical perspective, the results indicate that prompt engineering can become an essential skill in engineering education, on par with programming and simulation [9]. By adopting structured prompt design, engineers can reduce ambiguity, accelerate project timelines, and improve compliance with regulatory and safety standards.

However, the study is not without limitations. The evaluation was based on illustrative case studies and controlled computational tasks rather than industrial-scale implementations. Only four LLMs were also tested, and findings may vary as models evolve. Future research should expand validation across different engineering disciplines, larger datasets, and real-world industry applications.

Beyond the controlled case studies presented, the industrial deployment of prompt engineering raises specific challenges. In large-scale engineering projects, prompts that fail to incorporate regulatory requirements (e.g., ISO 9001 for quality management, ISO 26262 for functional safety in automotive systems, or IEEE 830 for software requirements) can lead to outputs that are technically accurate yet non-compliant with established standards [11–13]. This misalignment introduces risks in structural integrity, safety-critical aerospace applications, and chemical process design, where even minor deviations may have significant consequences [16].

Another limitation is the sensitivity of generative AI to domain-specific terminology and multilingual contexts, which are common in global engineering environments. Without rigorous adaptation and validation, models may misinterpret prompts that combine local regulatory frameworks with specialised technical language [16]. In practice, structured prompt engineering must be accompanied by domain validation, standardised documentation, and human oversight to ensure reliable integration into industrial workflows [10].

Finally, while the present study demonstrates the benefits of structured prompts in improving output reliability, it does not evaluate long-term adoption issues, such as user training, organisational change, or the scalability of prompt libraries across different engineering sectors. Therefore, Future research should investigate technical performance and the governance frameworks, audit mechanisms, and compliance protocols required to safely embed prompt engineering into industrial and academic practice [14].

4 Results of the Comparative Evaluation

To systematically evaluate the performance of the models, a test was conducted using a set of 20 prompts covering everyday engineering tasks, including factual knowledge (materials science), procedural understanding (manufacturing processes), and mathematical problem-solving (numerical integration in Python). Each response was assessed on a 0–5 scale based on five criteria: accuracy, completeness, clarity & coherence, confidence & justification, and problem-solving ability.

The aggregated results are presented in Table 1, while the complete version with all tasks can be found in Appendix A (Table A1).

Table 1: Summary of model performance metrics by evaluation criterion.

Model	Question Type	Accuracy	Completeness	Clarity & Coherence	Confidence & Justification	Problem-Solving	Total Score
ChatGPT	All tasks	5.0	5.0	5.0	5.0	5.0	25.0
DeepSeek	All tasks	5.0	5.0	5.0	5.0	5.0	25.0
Gemini	All tasks	4.0	4.0	4.0	4.0	5.0	21.0
MetaAI	All tasks	3.0	3.0	3.0	3.0	5.0	17.0

It is worth noting that the data presented in the supplementary material (Table A1),available at link[1], show uniformly maximum scores (5.0) across all models for computational tasks such as arithmetic operations, matrix multiplication, factorial calculation, and numerical integration. While this confirms the capability of LLMs to handle well-defined algorithmic problems, it also reduces the discriminative power of the comparative evaluation. In other words, the absence of variation in these tasks limits how differences in model reasoning or architecture can be observed.

This phenomenon can be explained by the fact that computational tasks typically involve deterministic procedures with clear syntactic structures, which align well with the token prediction mechanisms of large language models. Once the underlying algorithm is well represented in the training data, even less robust models can consistently reproduce correct outputs. Contrastingly, conceptual and procedural questions require contextual interpretation, domain

[1] The complete version with all tasks is available online as supplementary material: *Table A1 – Performance Analysis by Specific Task and Model*, at: https://docs.google.com/spreadsheets/d/1doZ14asQtKiOsKKhbT51Ni34ZcJxYTkzfwalhf6OjPw/edit?usp=sharing

knowledge, and abstraction, where architectural differences and training depth become more evident.

Future evaluations should therefore expand the range of computational tasks beyond basic algorithmic exercises, incorporating more complex programming challenges (e.g., multi-step simulations, error handling, optimisation routines). Such tasks would provide a stronger basis for discriminating between models and reflect more realistic engineering workflows.

A performance hierarchy emerged in conceptual and factual tasks. In questions such as “identify the main input in aluminium production” or “explain the purpose of heat treatment in metallurgy”, ChatGPT and DeepSeek consistently achieved maximum scores. Gemini obtained lower results but was still within acceptable average ranges, while MetaAI showed significantly weaker performance. This disparity decreased in computational tasks. All four models achieved the maximum score for programming and mathematics prompts, such as generating a Fibonacci sequence or performing matrix multiplication with NumPy. This suggests that even models with weaker conceptual reasoning can excel in well-defined algorithmic problems, an observation consistent with previous studies highlighting the proficiency of LLMs in code-related tasks with clear syntax [5].

Table 1 confirms that ChatGPT and DeepSeek are the best overall performers, with a perfect average score. Gemini remains a reliable model, although it is less robust in conceptual reasoning. MetaAI positioned itself as the least effective model, with notable clarity and contextual justification deficiencies. A representative example illustrates this failure: for the prompt “What is the output of the following Python code? print(2 + 3 * 2)”, ChatGPT correctly answered 8, explaining operator precedence. In contrast, MetaAI incorrectly answered 10, demonstrating a factual accuracy and reasoning failure.

In summary, the evaluation shows that although prompt engineering is critical for optimising results [1,2,3], model architecture and training depth remain decisive elements for performance [15]. While ChatGPT and DeepSeek demonstrated high reliability across all domains, MetaAI’s performance reinforces the need for rigorous verification of the final answer and the underlying reasoning the model provided.

Components of Good Prompts

Five elements emerged as critical for effective prompt design in technical fields: task specificity, contextual constraints, output formatting, audience alignment, and the definition of practical limitations. These factors establish a framework that transforms abstract design objectives into precise AI instructions. Without them, even advanced models tend to produce generic or incomplete responses, underscoring that prompt quality directly influences engineering outcomes [1,2].

Task Specification. A clear definition of the objective is essential to reduce ambiguity. For example, asking an AI system to “design a circuit” will likely yield a basic schematic. However, the more specific request “design a low-pass filter circuit for EEG signal processing with a cutoff frequency of 50 Hz” produces an output aligned with biomedical standards. Prior research confirms that prompts containing quantifiable information significantly improve the accuracy of thermodynamic simulations and mechanical analyses [4,15].

Contextual Constraints. Effective prompts incorporate technical boundaries such as material properties, environmental conditions, or regulatory standards. For instance, instructing an AI to “optimise a bridge design” is insufficient without mentioning seismic requirements or material specifications. When Eurocode 8 constraints were included in civil engineering tasks, the AI-generated outputs adhered to safety parameters absent in unconstrained versions [11]. Embedding constraints thus ensures that the AI solution remains technically sound and legally and environmentally compliant.

Output Formatting. Explicitly requesting structured outputs increases reproducibility and usability. In electrical engineering, a prompt that specifies “generate a SPICE netlist for a bandpass filter (20–2000 Hz) with simulation results in tabular format” leads to ready-to-use files and visualisations. A 2022 survey of engineering professionals found that 78 % reported improved productivity when AI-generated outputs followed predefined formatting standards [5,7]. This aligns with broader findings in human–computer interaction, which emphasise the role of clear formatting in reducing cognitive load [6].

Audience Alignment. Prompts must consider the expertise of the intended reader. For example, an explanation of catalyst behaviour in hydrogen fuel cells for senior engineers should include reaction kinetics and mathematical modelling. In contrast, the duplicate content for first-year students should use simplified terminology and diagrams. Studies in engineering education show that prompts tailored to learners’ backgrounds enhance retention and conceptual understanding by nearly 30 % [8, 10].

Constraint Definition. Setting boundaries on word count, safety norms, or material restrictions prevents overgeneralization. In aerospace engineering, specifying “optimise a convergent-divergent rocket nozzle for Mach 3 using Inconel 718 within 300 words” compels the AI to balance technical accuracy with conciseness. Such constraints have been identified as key to minimising AI hallucinations and ensuring outputs remain practical for industrial adoption [4].

Taken together, these components function synergistically. Task specification and contextualization reduce ambiguity, formatting enhances reproducibility, audience alignment improves accessibility, and constraints ensure practical feasibility. These elements can transform generic AI outputs into reliable

engineering tools when integrated. This confirms previous findings that structured prompt engineering can improve simulation accuracy by up to 40 % compared to open-ended queries [1, 5].

5 Conclusion

The results of this study reinforce the central thesis that the quality of prompts alone does not guarantee the excellence of AI-generated responses. Model architecture, training data, and reasoning capabilities are determining factors. As demonstrated, even well-designed prompts can lead to inaccurate responses when the model lacks a robust understanding of the engineering domain. ChatGPT and DeepSeek stood out among the evaluated systems for their ability to deliver more accurate, complete, and well-justified responses, highlighting that selecting the AI tool is as important as prompt engineering.

To address this challenge, this article proposed a systematic structure based on five key elements: (i) specificity, (ii) contextualization, (iii) formatting, (iv) audience alignment, and (v) constraint definition, to convert abstract engineering problems into structured directives for AI. Applying these principles is not merely theoretical; studies show that prompts specifying material properties or analysis parameters can improve the accuracy of simulations by more than 47 % [15], confirming their practical relevance.

Concerning future directions, prompt engineering may become an essential competence for modern engineers, on par with programming and computational modelling. Future innovations may include adaptive interfaces, prompt libraries tailored to each domain, and methodologies informed by cognitive science. Such advances will be crucial to align prompt structures with emerging technologies, such as quantum computing and generative design.

Nevertheless, ethical considerations remain paramount. Engineers must continue to verify AI-generated outputs, especially in safety-critical domains such as structural integrity and aerospace. Tools must adhere to standards, such as those established by the IEEE [13], and serve to enhance, rather than replace, human judgment. Academic institutions and industry leaders should integrate prompt engineering into curricula, training programs, and certification pathways to prepare professionals for this transformation.

Moreover, this study aligns with recent trends in engineering education and professional practice, emphasising the need to critically prepare future engineers to engage with AI-based systems. Systematic reviews show that teaching strategies incorporating AI tools and active learning significantly enhance performance and conceptual retention in STEM fields [8]. This suggests that integrating prompt engineering into curricula and workplace environments is desirable and indispensable to ensure that engineers can responsibly and effectively harness the potential of emerging technologies.

Acknowledgment

The authors would like to thank the UNIGOU Academic Internships Program for providing the opportunity that fostered the collaboration between Brazil and the Czech Republic.

This contribution was prepared with the support of the SGS-2025-026 project entitled “Artificial intelligence algorithms in sustainable manufacturing”, financed by the Ministry of Education, Youth and Sports of the Czech Republic.

References

[1] GGALIWANGO, M., NAKAYIZA, H. R., JJINGO, D., NAKATUMBA-NABENDE, J. Prompt Engineering in Large Language Models. In: *Data Intelligence and Cognitive Informatics*. Singapore: Springer, 2024. DOI: 10.1007/978-981-99-7962-2_30.

[2] KOSTEN, C., NOORALAHZADEH, F., STOCKINGER, K. Evaluating the Effectiveness of Prompt Engineering for Knowledge Graph Question Answering. *Frontiers in Artificial Intelligence*. 2025, 7. DOI: 10.3389/frai.2024.1454258.

[3] ZAMFIRESCU-PEREIRA, J., KUMAR, V., CHAN, J., et al. Why Johnny Can’t Prompt: How Non-AI Experts Try (and Fail) to Design LLM Prompts. In: *CHI Conference on Human Factors in Computing Systems*. New York: ACM, 2023.

[4] CHEN, X., LI, Y., ZHAO, H. Task Decomposition in AI-Enhanced Engineering: A Framework for Prompt Optimisation. *Journal of Computational Design and Engineering*. 2021, 8(3), 245–263.

[5] GUPTA, R., LEE, S. Optimizing AI-Driven Engineering Workflows through Prompt Structuring. *International Journal of Artificial Intelligence Applications in Engineering*. 2022, 15(4), 177–192.

[6] SWELLER, J. Cognitive Load Theory: Recent Theoretical Advances. In: PLASS, J., MORENO, R., BRÜNKEN, R. (eds.). *Cognitive Load Theory*. Cambridge: Cambridge University Press, 2011. p. 29–47.

[7] GILL, S. S.; TAYLOR, S. J. Artificial Intelligence in Industry 4.0: A Survey on Compliance, Standards, and Governance. Future Generation Computer Systems. Elsevier, 2022, 128, 91–106. DOI: 10.1016/j.future.2021.10.012.

[8] FREEMAN, S., EDDY, S. L., McDONOUGH, M., SMITH, M. K., OKOROAFOR, N., JORDT, H., WENDEROTH, M. P. Active learning increases student performance in science, engineering, and mathematics. *Proceedings of the National Academy of Sciences*. 2014, 111(23), 8410–8415.

[9] ZHANG, J., LEE, J., KIM, H. Large Language Models in STEM Education: Opportunities and Challenges. *Computers & Education: Artificial Intelligence*. 2023, 4, 100152. DOI: 10.1016/j.caeai.2023.100152.

[10] FLORIDI, L.; COWLS, J. A Unified Framework of Five Principles for AI in Society. Harvard Data Science Review. 2019, 1(1). DOI: 10.1162/99608f92.8cd550d1.

[11] EUROPEAN COMMISSION JOINT RESEARCH CENTRE (JRC). *Worked Examples for Seismic Design of Buildings According to Eurocode 8*. Luxembourg: Publications Office of the European Union, 2022. Disponível em: https://eurocodes.jrc.ec.europa.eu.

[12] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. ISO 9001:2015 – Quality management systems – Requirements. Geneva: ISO, 2015.

[13] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. ISO 26262:2018 – Road vehicles – Functional safety. Geneva: ISO, 2018.

[14] EUROPEAN COMMISSION. Ethics Guidelines for Trustworthy AI. High-Level Expert Group on Artificial Intelligence. Brussels: European Union, 2019. Disponível em: https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai.

[15] SILVA, J., SCHMIDT, M. The Impact of Structured Prompting on the Fidelity of Finite Element Analysis Simulations. *Journal of Computational Engineering*. 2023, 12(3), 451–468.

[16] FISCHER, T.; VÖLKER, J.; MAIER, A. Challenges of Multilingual Natural Language Processing in Global Engineering Design. Artificial Intelligence for Engineering Design, Analysis and Manufacturing. Cambridge University Press, 2021, 35(4), 457–472. DOI: 10.1017/S0890060421000309.