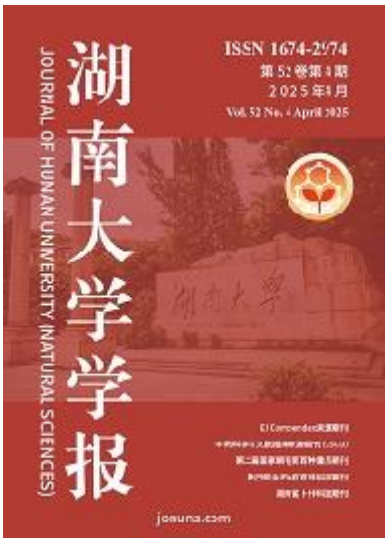

Journal of Hunan University

(Natural Sciences)

Vol. 52 No. 4

April 2025

Available online at

https://jonuns.com

Open Access Article

https://doi.org/10.55463/issn.1674-2974.52.4.2

Tiny Large Language Models in Embedded Nvidia Portable Hardware Comparative Analysis

Marco Antonio Jinete Gómez[1], Robinson Jiménez-Moreno[2], Anny Astrid Espitia-Cubillos[2]*

[1]Research assistant, Engineering Faculty, Universidad Militar Nueva Granada, Bogotá, Colombia

[2]Associated Professors of Engineering Faculty, Universidad Militar Nueva Granada, Bogotá, Colombia

* Corresponding author: anny.espitia@unimilitar.edu.co

Article History:

Received: March 7, 2025
Revised: April 7, 2025
Revised: April 21, 2025
Accepted: April 29, 2025
Published: May 30, 2025

Abstract: This study leverages the NVIDIA platform, a hardware with constrained computational resources, to evaluate the real-time performance of various small language models as local assistants, functioning without Internet access. The research employed a novel four-phase methodology, beginning with the selection of small language models for evaluation, followed by the design of a reproducible test protocol for future studies. This protocol incorporates both quantitative and qualitative assessment criteria, including latency, power consumption, memory usage, accuracy, creativity, narrative coherence, structural adherence, and Spanish performance. In the third phase, the selected models were locally embedded and executed on the hardware to compare their respective performances. Finally, the suitability of each model for real-time applications was analyzed, leading to the development of a comprehensive test protocol. The findings indicate that, of the nine models evaluated, Qwen2.5 with 3 billion parameters emerges as the optimal choice when resources allow, while Qwen2.5 with 0.5 billion parameters provides a viable alternative for scenarios with severe resource limitations.

Keywords: large language models, tiny LLM, Nvidia jetson Xavier hardware, Qwen2.5 model, Phi3.5 model, Smollm2 model, Mistral model.

嵌入式 Nvidia 便携式硬件中的微小大型语言模型对比分析

摘要： 本研究利用 Nvidia Jetson Xavier 平台，该平台具有有限的计算资源，评估了各种小型语言模型在没有互联网连接的情况下作为本地助手的实时性能。本研究采用了一种创新的四阶段方法论，首先选择要评估的小型语言模型，其次设计了一个可重复的测试协议，以便未来的研究使用，该协议包括定量和定性评估元素，主要指标包括延迟、功耗、内存使用、准确性、创造力、叙事连贯性、结构遵循性以及西班牙语表现。在第三阶段，选择的模型被嵌入并在硬件上本地执行，以比较它们的性能。最后，分析了每个模型在实时应用中的适用性，从而制定了全面的测试协议。研究结果表明，在评估的九个模型中，当资源允许时，具有 30 亿参数的 Qwen2.5 是最佳选择，而具有 5 亿参数的 Qwen2.5 则在资源极其有限的情况下提供了可行的替代方案。

关键词：大型语言模型、tiny LLM、Nvidia jetson Xavier **硬件、**Qwen2.5 **模型、**Phi3.5 **模型、**Smollm2 **模型、**Mistral **模型**

1. Introduction

Long language models (LLMs) are in full development and application; therefore, their understanding is necessary for training tasks or use for linguistic inferences [1], from different points of view of security [2], privacy [3], and stability [4] to biased or prejudiced responses [5]. Thus, LLM models, such as ChatGPT, impact user feedback [6] based on their responses and vulnerabilities [7], which leads to the development and application of various LLM models to continue in a growing research boom.

LLM applications are oriented to geoscientific data analysis [8], mathematical reasoning [9], text reordering [10], equipment fault diagnosis [11], radio frequency circuit design [12], medical reasoning [13] and decision making [14] among many others. Although these applications are based on existing models, the development bases continue to generate new models based on encoder–decoder settings [15] and optimization of inference models based on transformer networks [16]. These advances have impacted small models operating as low-parameter LLMs [17], oriented to more specific applications, such as clinical tasks [18], sentiment analysis [19], and even property prediction in solid crystals [20].

Robotic assistive systems are already starting to use language models for their control [21], such that embedded systems based on machine learning algorithms and the Internet of Things [22] can be used as voice assistants in local hardware without Internet connection [23], which has been little studied and whose results can benefit remote populations with specific needs but with connectivity difficulties and limited resources. Some embedded systems such as Nvidia Jetson are used for local applications based on deep learning algorithms [24], such as urban traffic monitoring [25]; for higher robustness requirements, cards such as Nvidia Jetson Xavier [26], [27] are used, but no documented report on the performance of small LLM models on these platforms has been found yet; therefore, in this study, a novel methodology is established to fill this gap.

The rise of small language models (maximum 7 billion parameters) has opened new possibilities in embedded systems with limited computational capacity. In this paper, we report the results of embedding different small language models in an Nvidia Jetson Xavier card and analyze their performance for use as local assistants without internet connection, evaluating key metrics such as latency, power consumption, memory usage, accuracy, creativity, narrative coherence, adherence to the requested structure, and performance in Spanish, validating the functionality of each one for real-time applications. These hardware devices are used in robotics, computer vision, and artificial intelligence (AI) applications. However, as previously discussed, no systematic comparison exists that evaluates the performance of these models in hardware- and energy-constrained environments, both quantitatively and qualitatively.

The first section presents the state of the art and research objectives, and the second section explains the proposed methodology. The third section presents and contrasts the results obtained with each of the nine small

language models from a quantitative and qualitative perspective. Finally, the fourth section presents the conclusions.

2. Methodology

To provide a rigorous methodological framework to evaluate the performance of small-language models executed locally on embedded hardware, four methodological steps were followed. First, different small language models susceptible to being embedded in the NVIDIA Jetson Xavier hardware were selected. Second, a reproducible testing protocol was designed for future research, including both quantitative and qualitative elements with key metrics such as latency, energy consumption, memory usage, accuracy, creativity, narrative coherence, adherence to the requested structure, and performance in Spanish. Third, the designed testing protocol was implemented, starting with the installation of the selected small language models on NVIDIA Jetson Xavier hardware and the configuration of the virtual environments. This allows for the execution of batch tests with standardized inputs and recording of metrics using monitoring tools. Fourth, the results of each model for real-time applications were analyzed, and their suitability was compared to draw conclusions and make recommendations.

This methodology is shown in the flow diagram in Figure 1.

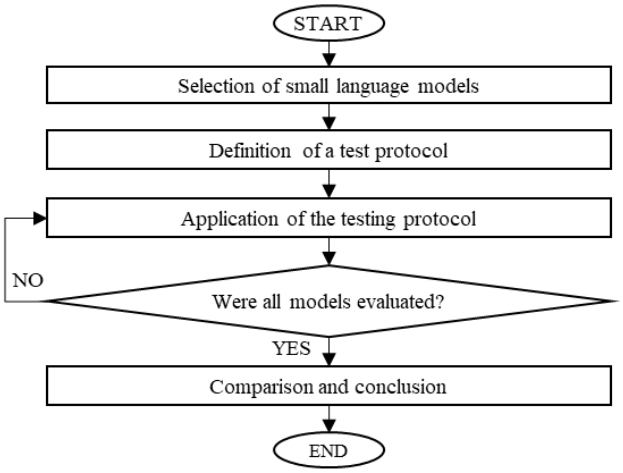

Figure 1. Methodology flowchart (Source: developed by the authors)

The language models were embedded on an NVIDIA Jetson Xavier AGX hardware platform featuring 512 CUDA cores, 64 Tensor Cores, 8-core Carmel ARMv8.2 CPUs and 16GB of RAM operating under Nvidia Jetson Linux 35.2.1, Linux 5.10, and Ubuntu 20.04.

3. Results

The nine language models selected for evaluation correspond to qwen2.5, with 0.5, 1.5, and 3 billion parameters; phi 3, 3.5, and 3.5; smollm2 with 1.7 billion parameters; and mistral with 7 billion parameters and its OpenOrca version that can be embedded in the Nvidia Jetson Xavier board.

The proposed evaluation protocol encompasses both a quantitative and a qualitative part, inference latency (time in milliseconds per token generated), generation variability (standard deviation), length of generated text (number of tokens), narrative coherence (evaluated by semi-automated analysis and expert review) are determined as evaluation metrics to validate the performance of each language model, creativity, RAM and VRAM usage (monitored with tegra_stats and jtop), adherence to the requested structure, model accuracy in text generation and classification tasks and real-time performance (responsiveness in interactive applications), six tests are designed for their application (Table 1).

Table 1. Test description (Source: developed by the authors)

Test	Objective	Prompt	Procedure
A: Creative Text Generation	Evaluate narrative coherence, fluency, creativity and structure in content generation.	"Write a short story about a robot and a human who embark on an adventure in a post-apocalyptic world".	Analyze the length of generated text (tokens), narrative coherence (assessed through semi-automated analysis and expert review), and creativity.
B: Text Summary	To measure the model's ability to synthesize information while maintaining the essence of the content.	A 300-500 word paper on the impact of artificial intelligence on education is provided." It then requests: "Summarize the following text in a paragraph of 100 words maximum."	Compare the generated summary with a reference summary prepared by experts.
C: Answering General Knowledge Questions	To assess the ability to understand and respond accurately to general domain questions.	"What are the main ethical challenges associated with artificial intelligence in medicine?"	Obtain statistics and assess consistency of responses. Compare responses with a set of responses scored by ethics and AI experts.
D: Execution of Complex Instructions	Verify the model's ability to follow specific and structured instructions.	"Create a list of 5 recommendations for improving energy efficiency in homes, justifying each recommendation in two sentences."	Evaluate variability in adherence to requested structure. Check each response contains five recommendations and requested justification.
E: Robustness to Noisy Inputs	Analyze the model's ability to handle input with spelling and	A deliberately altered text is submitted, with omission of	Compare the output of each model with the result obtained

	grammatical errors without losing coherence.	vowels, consonants, and spelling.	from the original text.
F: Long Input Handling and Scalability	Evaluate the model's ability to process long texts without compromising the quality of the output.	A 1000-word paper on the history of computing" is provided and requests: "Generate a 200-word summary of the following document."	Measure consistency in the response and resource use. Evaluate coherence, accuracy, and adherence to instructions. Review expert comments on the quality of the output.

For each test, the corresponding prompt must be executed in each language model to be evaluated, and at least 10 executions must be performed to obtain values that allow calculating the averages and variability of latency and variability in generation.

According to the designed protocol, each test was run ten times with each of the nine models to ensure statistical consistency of results and minimize the impact of random fluctuations in performance. This rigorous methodology allowed us to calculate not only the average values, but also to evaluate the variability in performance under identical conditions. As shown in Table 2, the latency per token (ms) for each model in the C test across the ten runs, the qwen2.5:0.5b models maintained consistently low latency (16-18 ms/token) with minimal standard deviation, while larger models such as mistral:7b showed greater variability (52-59 ms/token).

Table 2. Behavior of latency per token (ms) (Source: developed by the authors)

Model	Average	Standard deviation
Qwen2.5:0.5b	17.24	0.26
Qwen2.5:1.5b	26.91	0.71
Smollm2:1.7b	33.26	1.14
Qwen2.5:3b	39.57	1.06
Phi3.5:3.8b	43.89	2.28
Phi3.5:latest	41.21	1.13
Phi3:latest	55.83	2.16
Mistral:7b	55.57	2.51
Mistral-openorca:7b	55.68	2.63

Performance consistency should also be considered when selecting a model for applications that require predictable responses in resource-constrained environments (see Table 3). where the coefficient of variation represents the standard deviation as a percentage of the mean, providing a normalized measure of dispersion that allows the comparison of stability between models of different latency scales.

Table 3. Stability Metrics (Source: developed by the authors)

Model	Minimum	Maximum	Range	Variation coefficient (%)
Qwen2.5:0.5b	16.64	17.66	1.02	1.50
Qwen2.5:1.5b	25.84	27.75	1.92	2.65
Smollm2:1.7b	32.26	36.31	4.05	3.43
Qwen2.5:3b	37.92	41.18	3.26	2.68
Phi3.5:3.8b	38.81	46.51	7.70	5.20
Phi3.5:latest	39.23	43.30	4.07	2.74
Phi3:latest	51.12	58.60	7.48	3.87
Mistral:7b	52.48	60.28	7.81	4.52
Mistral-openorca:7b	52.56	61.85	9.30	4.72

The analysis of the evaluated models showed a clear relationship between the size of the model and its performance characteristics. Where Qwen2.5:0.5b stands out as the fastest model with the lowest latency (17.24 ms/token) and the highest throughput (58.53 tokens/second). While the Qwen2.5:1.5b model ranks second in speed with 26.91 ms/token and 38.69 tokens/second. Larger models, such as Mistral:7b and Mistral-openorca:7b, have the highest latencies (approximately 55.6 ms/token) and the lowest throughputs (approximately 18.13 tokens/second).

Regarding the use of hardware resources, the 7 B models (Mistral and Mistral-openorca) consume significantly more memory (46.5% of the total available memory) compared to the smaller models Qwen2.5:0.5b and Qwen2.5:1.5b, which consume 22.69% and 26.58%, respectively. In terms of CPU, smaller models such as Qwen2.5:0.5b use more CPU (10.77%) than larger models such as Phi3:latest (5.94%) and Phi3.5:3.8b (5.93%), suggesting a better CPU/GPU balance in larger models.

The plot of average latency per token for each model and test (Figure 3) reveals important patterns, such as that in Test F showing latency spikes in several models, especially in Phi3, where it reaches 140 ms/token, well above its overall average. This indicates that the models had difficulties with large inputs. In relation to consistency, the Qwen2.5:0.5b and Qwen2.5:1.5b models consistently maintained low latencies across the different tests, whereas the larger models showed greater variability.

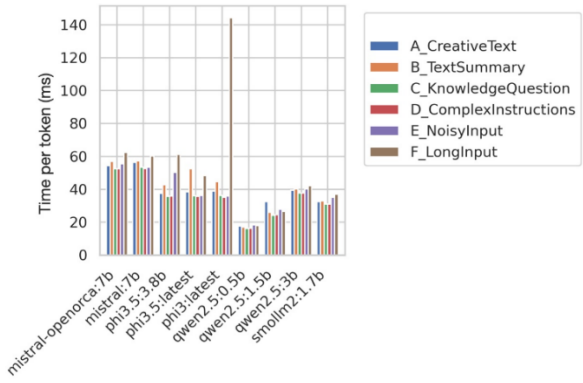

Figure 3. Average latency per token for each model and test (Source: developed by the authors)

In relation to memory size and performance, a positive correlation is evident, that is, there is a clear pattern where higher memory usage is associated with higher latency. The models were grouped into three main categories: high efficiency (Qwen2.5:0.5b and Qwen2.5:1.5b with low memory usage and low latency), medium efficiency (Smollm2:1.7b, Qwen2.5:3b, Phi3.5:latest and phi3.5:3.8b), and high capacity/low efficiency (Mistral:7b, Mistral-openorca:7b, and Phi3:latest).

Figure 4 shows that the efficiency decreases exponentially, but not linearly, with increasing model size. Tests C and D show better performance in tokens/s than tests F and E. In small models (0.5B-1.5B), performance on different tasks is significantly higher (50-60 tokens/second) compared to 7 B models (15-20 tokens/second).

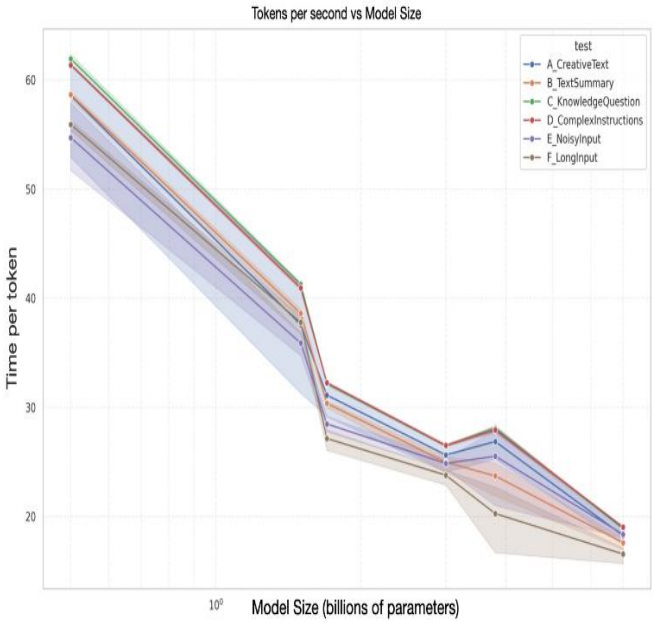

Figure 4. Tokens per second vs. Model size (Source: developed by the authors)

The results clearly show that architecture matters more than size; Qwen2.5 models consistently outperform other models of similar sizes, indicating better design and optimization. There is an inflection point around the 1.5-3B parameters, where the trade-off between capabilities and performance is maximized for embedded systems. It is also noted that efficiency is relative, as smaller new generation models (such as Qwen2.5) show efficiencies that rival or exceed older models of considerably larger size. In general, the performance does not degrade linearly with model size, suggesting that architectural factors may mitigate the impact of size.

Table 4 illustrates the ranking obtained by each of the nine models compared according to their technical feasibility with weightings proposed by the authors for each of the quantitative criteria considered. As latency is considered critical for real-time applications, throughput is important to achieve efficient processing, memory usage is a limited resource in embedded systems, CPU usage has an impact on energy consumption, and the model size assesses efficiency per parameter.

This ranking should be interpreted by considering the specific requirements of each application, where factors such as output quality, task-specific accuracy, or special inference needs could alter the ranking for particular use cases.

Table 4. Comparative ranking of models according to their technical feasibility (Source: developed by the authors)

Model	Latency 35%	Throughput 25%	Memory 20%	CPU 10%	Size 10%	Total (0-10)
Qwen2.5:0.5b	10.0	10.0	10.0	5.2	9.5	9.2
Qwen2.5:1.5b	8.5	8.3	8.7	6.0	9.0	8.5
Smollm2:1.7b	7.7	6.5	7.9	7.8	8.7	7.8
Qwen2.5:3b	6.9	5.4	6.8	6.0	7.5	7.3
Phi3.5:3.8b	6.3	5.4	6.2	8.4	7.2	7.1
Phi3.5:latest	6.5	5.4	6.5	8.2	7.2	7.0
Phi3:latest	4.9	5.4	6.4	8.3	7.2	6.0
Mistral:7b	4.9	3.9	5.2	8.1	5.5	5.2
Mistral-openorca:7b	4.9	3.9	5.3	7.6	5.5	5.0

For the ranking, a normalization of Metrics was established for Latency inverse normalized, where Qwen2.5:0.5b (17.24ms) = 10.0, and Mistral:7b (55.68ms) ≈ 4.9. The throughput was also normalized where Qwen2.5:0.5b (58.53 tokens/s) = 10.0, and Mistral:7b (18.13 tokens/s) ≈ 3.9. Memory was inversely normalized when Qwen2.5:0.5b (22.69%) = 10.0, and Mistral:7b (46.71%) ≈ 5.2. The CPU was normalized considering CPU/GPU balance, where Phi3.5:3.8b (5.93%) ≈ 8.4, and size/efficiency, which considers relative efficiency per parameter, favoring models that achieve more with less.

Table 5 illustrates the interpretation of the ranking scores proposed by the authors, according to their technical feasibility and practical usefulness. The results show that Qwen2.5 models dominate the ranking owing to their excellent optimization.

Table 5. Interpretation of ranking score (Source: developed by the authors)

Total score	Interpretation
9.0-10.0	Exceptional performance for embedded systems
8.0-8.9	Excellent balance of performance and resources
7.0-7.9	Very good performance with some limitations
6.0-6.9	Acceptable performance with notable compromises
5.0-5.9	Significant limitations for resource-constrained systems
<5.0	Not recommended for most NVIDIA Jetson applications

There is a clear relationship between size and total score, with notable exceptions in the Phi3.5 family. Efficient CPU usage mainly benefits Phi models, and the score drops significantly for models above 3 B

parameters. For specific applications, models with lower total scores may be preferable if a particular metric (such as output quality, not measured here) is critical.

Based on established benchmarks and the ranking of small language models on NVIDIA Jetson embedded systems, according to their technical feasibility, it is concluded that the Qwen2.5-3B model exhibits the best overall performance with an excellent balance between quality and size; the Qwen2.5-1.5B model excelled in practical tasks with great parameter efficiency; the Mistral-OpenOrca-7B model presents good overall performance but is larger in size; the Mistral-7B model has a solid performance but with some specific errors; the Qwen2.5-0.5B model has surprising performance for its ultra-compact size; the SmoLLM2-1.7B model has significant issues that limit its applicability; and the Phi-3/Phi-3.5 models are not recommended in their current state.

Additionally, this analysis addresses a critical but often underestimated aspect of implementing language models: their qualitative performance in languages other than English, specifically, Spanish. While quantitative metrics such as latency, power consumption, and memory utilization are fundamental for assessing technical feasibility, linguistic quality ultimately determines the practical usefulness of these models in real-world applications.

Tables 6 to 11 present the results of the qualitative criteria of tests A to F, which are evaluated using a scale of 0 to 10, where an excellent rating is 10, very high is 9, high is 8, very good is 7, good is 6, average is 5, average-low is 4, low is 3, poor is 2, null is 1, and not evaluable is equivalent to a score of 0. Table 6 presents the results for the elements Narrative Quality, Coherence and Spanish to assess Creative Text Generation in test A.

Table 6. Creative Text Generation Assessment (Source: developed by the authors)

Model	Creative Text Generation		
	Narrative Quality	Coherence	Spanish
Qwen2.5:0.5b	5	5	6
Qwen2.5:1.5b	0	0	6
Smollm2:1.7b	3	3	2
Qwen2.5:3b	8	8	7
Phi3.5:3.8b	5	4	2
Phi3.5:latest	5	4	2
Phi3:latest	5	4	2
Mistral:7b	8	8	6
Mistral-openorca:7b	8	8	7

Table 7 presents the assessments of the elements Synthesis Capacity, Coherence and Spanish to qualify the Text Summary capacity in test B.

Table 7. Text Summary Assessment (Source: developed by the authors)

Model	Text Summary		
	Synthesis Capacity	Coherence	Spanish
Qwen2.5:0.5b	8	8	7
Qwen2.5:1.5b	8	8	10
Smollm2:1.7b	5	5	2
Qwen2.5:3b	9	8	9
Phi3.5:3.8b	1	1	0
Phi3.5:latest	1	1	0
Phi3:latest	3	3	2
Mistral:7b	8	8	7
Mistral-openorca:7b	8	8	7

Table 8 shows the perception regarding the answers to the Knowledge Questions, considering precision, structure, and Spanish in test C.

Table 8. Answering Knowledge Questions Assessment (Source: developed by the authors)

Model	Answering Knowledge Questions		
	Precision	Structure	Spanish
Qwen2.5:0.5b	8	8	6
Qwen2.5:1.5b	8	8	7
Smollm2:1.7b	3	5	2
Qwen2.5:3b	9	8	7
Phi3.5:3.8b	1	1	0
Phi3.5:latest	1	1	0
Phi3:latest	1	1	0
Mistral:7b	8	8	6
Mistral-openorca:7b	8	5	6

Table 9 assesses the Execution of Complex Instructions with the elements' adherence, coherence, and Spanish in Test D.

Table 9. Execution of Complex Instructions Assessment (Source: developed by the authors)

Model	Execution of Complex Instructions		
	Adherence	Coherence	Spanish
Qwen2.5:0.5b	5	5	2
Qwen2.5:1.5b	8	8	7
Smollm2:1.7b	3	3	2
Qwen2.5:3b	8	8	6
Phi3.5:3.8b	1	1	0
Phi3.5:latest	1	1	0
Phi3:latest	1	1	0
Mistral:7b	5	5	2
Mistral-openorca:7b	8	8	6

Table 10 presents the results for the elements Effectiveness, Correctness, and Spanish to assess the robustness against noisy inputs in test E.

Table 10. Execution of Complex Instructions Assessment (Source: developed by the authors)

Model	Execution of Complex Instructions		
	Effectivity	Correction	Spanish
Qwen2.5:0.5b	9	8	10
Qwen2.5:1.5b	9	8	10
Smollm2:1.7b	1	1	0
Qwen2.5:3b	9	8	10
Phi3.5:3.8b	1	1	0
Phi3.5:latest	1	1	0
Phi3:latest	1	1	0
Mistral:7b	8	8	7
Mistral-openorca:7b	8	8	7

Finally, Table 11 presents the assessments of the elements (Synthesis Capacity, Coherence and Spanish) to qualify for the handling of extensive entries in the F test.

In general, the F test (long input processing) is the most demanding for all the models compared, both quantitatively and qualitatively.

Table 11. Handling Long Entries Assessment (Source: developed by the authors)

Model	Handling long entries		
	Synthesis	Coherence	Spanish
Qwen2.5:0.5b	8	8	6
Qwen2.5:1.5b	8	8	7
Smollm2:1.7b	3	5	2
Qwen2.5:3b	8	8	7
Phi3.5:3.8b	1	1	0
Phi3.5:latest	1	1	0
Phi3:latest	1	1	0
Mistral:7b	8	8	7
Mistral-openorca:7b	5	8	7

After this analysis, a ranking is made with respect to the practical usefulness of these models, which is presented in Table 12, where the numerical rating (0-10) reflects both the linguistic performance and the efficiency of parameters for implementation in embedded systems on NVIDIA Jetson hardware.

Table 12. Comparative ranking of models according to their practical usefulness (Source: developed by the authors)

Model	Size	Strengths	Limitations	Total (0-10)
Qwen 2.5-3B	3B	Excellent narrative generation, exceptional synthesis, in-depth responses, robustness in the face of errors	Occasional minor errors, formatting inconsistencies	9,4
Qwen 2.5-1.5B	1.5B	Exceptional synthesis and correction, clear structuring, high-quality Spanish	Excessive caution in creative generation	8,7
Mistral - OpenOrca-7B	7B	Structured narrative, good synthesis, complete, informative responses	Larger size, some translation errors	8,3
Mistral -7B	7B	Excellent narrative ability, good synthesis, effective error correction	Occasional grammatical errors, Anglicisms	8,0
Qwen 2.5-0.5B	0.5B	Surprising performance for its size, good synthesis and correction	Logical inconsistencies, errors in complex tasks	7,2
SmoLLM2-1.7B	1.7B	Recognizable structure, identification of main themes	Serious grammatical problems, language switching	4,5
Phi-3.5 (3.8B)	3.8B	Partially coherent generation of creative text	Severe inconsistency in most tasks, language mixing	2,8
Phi-3.5 Latest	3.8B	Like the previous position with slight improvements	Severe inconsistency, chaotic multilingualism	2,6
Phi-3 Latest	-	Narrative beginnings with some coherence	Truncations with numerical codes, general inconsistency	2,1

For its interpretation, Table 5 is used, pointing out that the Qwen 2.5-3B model presents an exceptional performance for embedded systems, the Qwen 2.5-1.5B, Mistral-OpenOrca-7B and Mistral-7B models exhibit an excellent balance of performance and resources, the Qwen 2.5-0.5B model has a very good performance with some limitations, while the SmoLLM2-1.7B, Phi-3.5 (3.8B), Phi-3.5 Latest and Phi-3 Latest models are not recommended for NVIDIA Jetson applications.

4. Discussion

It is acknowledged that natural language processing has had differential advances that depend largely on the language; the greatest progress corresponds to the English language, with which most models have been trained. However, in [17], the authors presented the development of two compact models for text generation in Brazilian Portuguese. In this study, the linguistic performance of nine Tiny LLMs in Spanish embedded in Nvidia Portable Hardware was analyzed both quantitatively and qualitatively.

In [9], the performances of different LLMs in Catalan were compared considering the accuracy of the answers to mathematical problems, together with parameter sizes. In [11], the LLM comparison criteria were parameters, single inference time (50 tokens), memory usage, accuracy, precision, and F1 score, all of which were quantitative. In addition, in [15], open- and closed-source LLM models were evaluated in STEM tasks using measures of mean average precision, perplexity, Spearman correlation, and Kendall correlation. In [18], the Macro-averaged Receiver Operating Characteristic area under the curve, F1 scores, training time, sizes, inference time, and total cost for the triage task were used. In addition, in [17], several of the quantitative criteria proposed in this article were used; however, the qualitative criteria are novel at this level.

In this study, an evaluation protocol was established that contains both qualitative and quantitative elements to systematically compare the performance of language models. which If rigorously applied, it guarantees the statistical consistency of the results when obtained under identical conditions and minimizes the impact of

random variations in performance, calculating average, values, and variability in performance.

The weighting of quantitative and qualitative criteria can be modified for future research, considering the type of application for which the model will be used. Normalizing the data before calculating the total score ensured consistency of the evaluation scale. The authors also proposed a guide that facilitates and standardizes the interpretation of total score values in ranking according to their technical feasibility and practical utility.

5. Conclusion

The rankings obtained reflect not only the absolute linguistic quality but also the parameter efficiency, which is a crucial factor for resource-constrained embedded systems. Finally, we conclude that the Qwen2.5 family stands out significantly, demonstrating exceptional optimization for resource-constrained Spanish processing. By contrast, Phi models exhibit fundamental issues that compromise their practical usefulness in this specific context. For implementations on NVIDIA Jetson hardware, Qwen2.5-3B currently represents the best choice when resources permit, whereas Qwen2.5-0.5B offers a surprisingly capable alternative for extremely resource-constrained systems. The results of this study contribute to a deeper understanding of the inherent trade-offs between model capacity, performance, and energy efficiency by exploring their distinctive features, linguistic capabilities, and application potential. This allows for benchmarking, which guides model selection for specific applications in embedded systems. This knowledge is critical for driving the next generation of smart devices capable of processing and generating high-quality natural language without requiring a cloud infrastructure.

Declarations

Author Contributions

Conceptualization, formal analysis, and writing—review and editing Jiménez-Moreno R. and Espitia-Cubillos A.; methodology, validation, investigation supervision, project administration, funding acquisition, and data curation Jiménez-Moreno R.; writing—original draft preparation and visualization Jinéte Gómez M. All authors read and agreed to the published version of the manuscript.

Funding

Product derived from the research project titled “Diseño de un modelo de interacción humano robot mediante algoritmos de aprendizaje profundo” INV-ING-3971 financed by the vice-rector for research of the Universidad Militar Nueva Granada, year 2024.

Acknowledgements

The authors thank the Universidad Militar Nueva Granada, where they are associate professors, for the time and resources available for the development of this article.

Institutional Review Board Statement

The study was conducted in accordance with the Declaration of Helsinki and was approved by the Ethics Committee of the Universidad Militar Nueva Granada (project INV-ING-3971, date of approval: 18/01/2024).

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this manuscript. In addition, ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancies, have been completely observed by the authors.

References

[1] LIU Y., HE H., HAN T., ZHANG X., LIU M., TIAN J., ZHANG Y., WANG J., GAO X., ZHONG T., PAN Y., XU S., WU Z., LIU Z., ZHANG X., ZHANG S., HU X., ZHANG T., QIANG N., LIU T., and GE B. Understanding LLMs: A comprehensive overview from training to inference. *Neurocomputing*, 2025, 620: 129190, https://doi.org/10.1016/j.neucom.2024.129190

[2] YAO Y., DUAN J., XU K., CAI Y., SUN Z., and ZHANG Y. A survey on large language model (LLM) security and privacy: The Good, The Bad, and The Ugly. *High-Confidence Computing*, 2024, 4(2): 100211, https://doi.org/10.1016/j.hcc.2024.100211

[3] B. YAN, K. LI, M. XU, Y. DONG, Y. ZHANG, Z. REN, and X. CHENG. On protecting the data privacy of Large Language Models (LLMs) and LLM agents: A literature review. *High-Confidence Computing*, 2025, 100300, https://doi.org/10.1016/j.hcc.2025.100300

[4] P. Y. P. CHAN, J. KEUNG, and Z. YANG. Effectiveness of symmetric metamorphic relations on validating the stability of code generation LLM. *Journal of Systems and Software*, 2025, 222: 112330, https://doi.org/10.1016/j.jss.2024.112330

[5] Z. W. PETZEL, and L. SOWERBY, Prejudiced interactions with large language models (LLMs) reduce trustworthiness and behavioral intentions among members of stigmatized groups, Computers in Human Behavior, 2025, 165: 108563, https://doi.org/10.1016/j.chb.2025.108563

[6] RAPP A., DI LODOVICO C., and DI CARO L. How do people react to ChatGPT's unpredictable behavior? Anthropomorphism, uncanniness, and fear of AI: A qualitative study on individuals’ perceptions and understandings of LLMs’ nonsensical hallucinations. *International Journal of Human-Computer Studies*, 2025, 198: 103471, https://doi.org/10.1016/j.ijhcs.2025.103471

[7] M. A. FERRAG, F. ALWAHEDI, A. BATTAH, B. CHERIF, A. MECHRI, N. TIHANYI, T. BISZTRAY, and M. DEBBAH. Generative AI in Cybersecurity: A Comprehensive Review of LLM Applications and

Vulnerabilities. *Internet of Things and Cyber-Physical Systems*, 2025, https://doi.org/10.1016/j.iotcps.2025.01.001
[8] ZHANG J., CLAIRMONT C., QUE X., LI W., CHEN W., LI C., and MA X. Streamlining geoscience data analysis with an LLM-driven workflow. *Applied Computing and Geosciences*, 2025, 25: 100218, https://doi.org/10.1016/j.acags.2024.100218
[9] RHOMRASI L., AHSINI Y., IGUALDE-SÁEZ A., VINUESA R., HOYAS S., GARCÍA-SABATER J. P., FULLANA-I-ALFONSO M. J., and CONEJERO J. A. LLM performance on mathematical reasoning in Catalan language. *Results in Engineering*, 2025, 25: 104366, https://doi.org/10.1016/j.rineng.2025.104366
[10] SUN J., and LV Z., Zero-shot detection of LLM-generated text via text reorder. *Neurocomputing*, 2025, 631: 129829, https://doi.org/10.1016/j.neucom.2025.129829
[11] LIN L., ZHANG S., FU S., and LIU Y. FD-LLM: Large language model for fault diagnosis of complex equipment. *Advanced Engineering Informatics*, 2025, 65(Part A): 103208, https://doi.org/10.1016/j.aei.2025.103208
[12] JIN H., WANG J., SHENG J., WU Y., CHEN J., WANG Y., and LIU J. WiseEDA: LLMs in RF Circuit Design. *Microelectronics Journal*, 2025, 158: 106607, https://doi.org/10.1016/j.mejo.2025.106607
[13] LIÉVIN V., HOTHER C. E., MOTZFELDT A. G., and WINTHER O., Can large language models reason about medical questions? *Patterns*, 2024, 5(3): 100943, https://doi.org/10.1016/j.patter.2024.100943
[14] XU Z., SONG T., and LEE Y. Confronting verbalized uncertainty: Understanding how LLM's verbalized uncertainty influences users in AI-assisted decision-making. *International Journal of Human-Computer Studies*, 2025, 197: 103455, https://doi.org/10.1016/j.ijhcs.2025.103455
[15] SOLIMAN G., ZAKI H., and KILANY M. A comparative analysis of encoder only and decoder only models for challenging LLM-generated STEM MCQs using a self-evaluation approach. *Natural Language Processing Journal*, 2025, 10: 100131, https://doi.org/10.1016/j.nlp.2025.100131
[16] CHITTY-VENKATA K. T., MITTAL S., EMANI M., VISHWANATH V., and SOMANI A. K. A survey of techniques for optimizing transformer inference. *Journal of Systems Architecture*, 2023, 144: 102990, https://doi.org/10.1016/j.sysarc.2023.102990
[17] CORRÊA N. K., FALK S., FATIMAH S., SEN A., and DE OLIVEIRA N. TeenyTinyLlama: Open-source tiny language models trained in Brazilian Portuguese. *Machine Learning with Applications*, 2024, 16: 100558, https://doi.org/10.1016/j.mlwa.2024.100558
[18] TAYLOR N., GHOSE U., ROHANIAN O., NOURIBORJI M., KORMILITZIN A., CLIFTON D. A., and NEVADO-HOLGADO A. Efficiency at scale: Investigating the performance of diminutive language models in clinical task. *Artificial Intelligence in Medicine*, 2024, 157: 103002, https://doi.org/10.1016/j.artmed.2024.103002
[19] CHIU I. C., and HUNG M. Finance-specific large language models: Advancing sentiment analysis and return prediction with LLaMA 2. *Pacific-Basin Finance Journal*, 2025, 90: 102632, https://doi.org/10.1016/j.pacfin.2024.102632
[20] ZHU J., REN Y., ZHOU W., XU J., NIU Z., ZHAN S., and MA W. LLM-mambaformer: Integrating mamba and transformer for crystalline solids properties prediction. *Materials Today Communications*, 2025, 44: 112029, https://doi.org/10.1016/j.mtcomm.2025.112029
[21] ZAHEDIFAR R., BAGHSHAH M. S., and TAHERI A. LLM-controller: Dynamic robot control adaptation using large language models. *Robotics and Autonomous Systems*, 2025, 186: 104913, https://doi.org/10.1016/j.robot.2024.104913
[22] OLIVEIRA F., COSTA D. G., ASSIS F., and SILVA I. Internet of Intelligent Things: A convergence of embedded systems, edge computing and machine learning. *Internet of Things*, 2024, 26: 101153, https://doi.org/10.1016/j.iot.2024.101153
[23] LAZZARONI L., BELLOTTI F., and BERTA R. An embedded end-to-end voice assistant. *Engineering Applications of Artificial Intelligence*, 2024, 136(Part B): 108998, https://doi.org/10.1016/j.engappai.2024.108998
[24] MITTAL S. A Survey on optimized implementation of deep learning models on the NVIDIA Jetson platform. *Journal of Systems Architecture*, 2019, 97: 428-442, https://doi.org/10.1016/j.sysarc.2019.01.011
[25] NOCUA F., PÉREZ-HOLGUÍN W. J., and PARDO-BEAINY C. Urban traffic monitoring based on deep learning on an embedded GPU. *Expert Systems with Applications*, 2025, 273: 126847, https://doi.org/10.1016/j.eswa.2025.126847
[26] CHEN Q., and JIANG X. A portable real-time concrete bridge damage detection system. *Measurement*, 2025, 240: 115536, https://doi.org/10.1016/j.measurement.2024.115536
[27] KORTLI Y., GABSI S., VOON L. F. L. Y., JRIDI M., MERZOUGUI M., and ATRI, M. Deep embedded hybrid CNN–LSTM network for lane detection on NVIDIA Jetson Xavier NX. *Knowledge-Based Systems*, 2022, 240: 107941, https://doi.org/10.1016/j.knosys.2021.107941

参考文:

[1] LIU Y., HE H., HAN T., ZHANG X., LIU M., TIAN J., ZHANG Y., WANG J., GAO X., ZHONG T., PAN Y., XU S., WU Z., LIU Z., ZHANG X., ZHANG S., HU X., ZHANG T., QIANG N., LIU T., and GE B. 了解 LLM：从训练到推理的全面概述. *神经计算*、, 2025, 620: 129190, https://doi.org/10.1016/j.neucom.2024.129190
[2] YAO Y., DUAN J., XU K., CAI Y., SUN Z., and ZHANG Y. 大型语言模型（LLM）安全性和隐私性调查：好、坏、丑。*高可信度计算*、 2024, 4(2): 100211, https://doi.org/10.1016/j.hcc.2024.100211
[3] B. YAN, K. LI, M. XU, Y. DONG, Y. ZHANG, Z. REN, and X. CHENG. 关于保护大型语言模型（LLM）和 LLM 代理的数据隐私： 文献综述。*高可信度计算*、 2025, 100300, https://doi.org/10.1016/j.hcc.2025.100300
[4] P. Y. P. CHAN, J. KEUNG, and Z. YANG. 对称变形关系对验证代码生成 LLM 稳定性的有效性。*系统与软件期刊》*、 2025, 222: 112330, https://doi.org/10.1016/j.jss.2024.112330
[5] Z. W. PETZEL, and L. SOWERBY, 与大型语言模型（LLMs）的偏见互动会降低受鄙视群体成员的可信度和行为意向、 *计算机在人类行为中的应用*, 2025, 165: 108563, https://doi.org/10.1016/j.chb.2025.108563
[6] RAPP A., DI LODOVICO C., and DI CARO L. 人们如何应对 ChatGPT 不可预测的行为？拟人化、不可思议和

对人工智能的恐惧：关于个人对 LLMs 无厘头幻觉的看法和理解的定性研究。*国际人机研究杂志》*、2025, 198: 103471, https://doi.org/10.1016/j.ijhcs.2025.103471
[7] M. A. FERRAG, F. ALWAHEDI, A. BATTAH, B. CHERIF, A. MECHRI, N. TIHANYI, T. BISZTRAY, and M. DEBBAH. 网络安全中的生成式人工智能：对 LLM 应用和漏洞的全面回顾。*物联网与网络物理系统》*、2025, https://doi.org/10.1016/j.iotcps.2025.01.001
[8] ZHANG J., CLAIRMONT C., QUE X., LI W., CHEN W., LI C., and MA X. 用 LLM 驱动的工作流程简化地球科学数据分析。*应用计算与地球科学*、2025, 25: 100218, https://doi.org/10.1016/j.acags.2024.100218
[9] RHOMRASI L., AHSINI Y., IGUALDE-SÁEZ A., VINUESA R., HOYAS S., GARCÍA-SABATER J. P., FULLANA-I-ALFONSO M. J., and CONEJERO J. A. 用加泰罗尼亚语进行数学推理的 LLM 成绩。*工程学成果*、2025, 25: 104366, https://doi.org/10.1016/j.rineng.2025.104366
[10] SUN J., and LV Z., 通过文本重排实现 LLM 生成文本的零镜头检测 *神经计算*, 2025, 631: 129829, https://doi.org/10.1016/j.neucom.2025.129829
[11] LIN L., ZHANG S., FU S., and LIU Y. FD-LLM：用于复杂设备故障诊断的大型语言模型。*高级工程信息学*, 2025, 65(A 部分): 103208, https://doi.org/10.1016/j.aei.2025.103208
[12] JIN H., WANG J., SHENG J., WU Y., CHEN J., WANG Y., and LIU J. WiseEDA：射频电路设计的 LLMs。*微电子学杂志*, 2025, 158: 106607, https://doi.org/10.1016/j.mejo.2025.106607
[13] LIÉVIN V., HOTHER C. E., MOTZFELDT A. G., and WINTHER O., 大型语言模型能推理医学问题吗？*模式*、2024, 5(3): 100943, https://doi.org/10.1016/j.patter.2024.100943
[14] XU Z., SONG T., and LEE Y. 面对语言化的不确定性：了解 LLM 的口头不确定性如何影响人工智能辅助决策中的用户。*国际人机研究期刊》*、2025, 197: 103455, https://doi.org/10.1016/j.ijhcs.2025.103455
[15] SOLIMAN G., ZAKI H., and KILANY M. 使用自我评价方法对仅编码器模型和仅解码器模型进行比较分析，以应对具有挑战性的 LLM 生成的 STEM MCQ。*自然语言必理期刊》*、2025, 10: 100131, https://doi.org/10.1016/j.nlp.2025.100131
[16] CHITTY-VENKATA K. T., MITTAL S., EMANI M., VISHWANATH V., and SOMANI A. K. 变压器推理优化技术概览。*系统结构期刊》*、2023, 144: 102990, https://doi.org/10.1016/j.sysarc.2023.102990
[17] CORRÊA N. K., FALK S., FATIMAH S., SEN A., and DE OLIVEIRA N. TeenyTinyLlama：以巴西葡萄牙语训练的开源微小语言模型。*机器学习与应用*、2024, 16: 100558, https://doi.org/10.1016/j.mlwa.2024.100558
[18] TAYLOR N., GHOSE U., ROHANIAN O., NOURIBORJI M., KORMILITZIN A., CLIFTON D. A., and NEVADO-HOLGADO A 规模效率：研究微型语言模型在临床任务中的表现。*人工智能在医学中的应用*、2024, 157: 103002, https://doi.org/10.1016/j.artmed.2024.103002
[19] CHIU I. C., and HUNG M. 金融专用大型语言模型：利用 LLaMA 2 推进情感分析和回报预测。*太平洋盆地金融期刊》*、2025, 90: 102632, https://doi.org/10.1016/j.pacfin.2024.102632
[20] ZHU J., REN Y., ZHOU W., XU J., NIU Z., ZHAN S., and MA W. 金融专用大型语言模型：利用 LLaMA 2 推进情感分析和回报预测。*太平洋盆地金融期刊》*、2025, 44: 112029, https://doi.org/10.1016/j.mtcomm.2025.112029
[21] ZAHEDIFAR R., BAGHSHAH M. S., and TAHERI A. LLM-controller：使用大型语言模型的动态机器人控制适应。*机器人与自主系统*、2025, 186: 104913, https://doi.org/10.1016/j.robot.2024.104913
[22] OLIVEIRA F., COSTA D. G., ASSIS F., and SILVA I. 智能物联网：嵌入式系统、边缘计算和机器学习的融合。*物联网*、2024, 26: 101153, https://doi.org/10.1016/j.iot.2024.101153
[23] LAZZARONI L., BELLOTTI F., and BERTA R. 嵌入式端到端语音助手 *人工智能的工程应用*、2024, 136(Part B): 108998, https://doi.org/10.1016/j.engappai.2024.108998
[24] MITTAL S. 英伟达™（NVIDIA®）Jetson 平台上深度学习模型优化实施调查。*系统架构期刊* 2019, 97: 428-442, https://doi.org/10.1016/j.sysarc.2019.01.011
[25] NOCUA F., PÉREZ-HOLGUÍN W. J., and PARDO-BEAINY C. 基于嵌入式 GPU 深度学习的城市交通监控。*专家系统与应用》*、2025, 273: 126847, https://doi.org/10.1016/j.eswa.2025.126847
[26] CHEN Q., and JIANG X. 便携式混凝土桥梁损伤实时检测系统。*测量*、2025, 240: 115536, https://doi.org/10.1016/j.measurement.2024.115536
[27] KORTLI Y., GABSI S., VOON L. F. L. Y., JRIDI M., MERZOUGUI M., and ATRI, M. NVIDIA Jetson Xavier NX 上用于车道检测的深度嵌入式混合 CNN-LSTM 网络。*基于知识的系统*、2022, 240: 107941, https://doi.org/10.1016/j.knosys.2021.107941

Word count: 5800 words, excluding references.

Peer review information:
Whether the manuscript was fast tracked? - No
Number of reviewer report submitted in first round: 3 reports
Number of revision rounds: 2 rounds
Final revised version submitted: April 26, 2025