electronics

MDPI

Article

Energy-Aware Multilingual Evaluation of Large Language Models

I. de Zarzà [1,2], Mauro Liz [3,4], J. de Curtò [2,3,5,*] and Carlos T. Calafate [6]

[1] Human Centered AI, Data & Software, LUXEMBOURG Institute of Science and Technology, 4362 Esch-sur-Alzette, Luxembourg; dezarza@uoc.edu
[2] Estudis d'Informàtica, Multimèdia i Telecomunicació, Universitat Oberta de Catalunya, 08018 Barcelona, Spain
[3] Escuela Técnica Superior de Ingeniería (ICAI), Universidad Pontificia Comillas, 28015 Madrid, Spain; mauroliz@bu.edu
[4] Department of Electrical and Computer Engineering, Boston University, Boston, MA 02215, USA
[5] Department of Computer Applications in Science & Engineering, BARCELONA Supercomputing Center, 08034 Barcelona, Spain
[6] Departamento de Informática de Sistemas y Computadores, Universitat Politècnica de València, 46022 València, Spain; calafate@disca.upv.es
* Correspondence: jdecurto@icai.comillas.edu

Abstract

The rapid deployment of Large Language Models (LLMs) in multilingual, production-scale systems has made inference-time energy consumption a critical yet systematically under-evaluated dimension of model quality. While accuracy-centric benchmarks dominate current evaluation practice, they fail to capture the energy cost of reasoning, particularly across languages and task complexities where consumption profiles diverge substantially. In this work, we present a comprehensive energy–performance evaluation of five instruction-tuned LLMs, spanning Transformer, Grouped-Query Attention, and State Space Model architectures, across thirteen typologically diverse languages and multiple task difficulty levels under controlled GPU-level energy measurement on NVIDIA H200 hardware. Our analysis encompasses 65 model–language configurations totaling over 5100 individual inference runs, supported by rigorous non-parametric statistical testing (Friedman tests, pairwise Wilcoxon signed-rank with Holm correction, and paired Cohen's d effect sizes). We report four principal findings. First, energy consumption varies up to threefold across models under identical workloads ($\chi^2 = 49.42$, $p = 4.78 \times 10^{-10}$, Friedman test), stratifying into three distinct energy regimes driven by architecture and generation dynamics rather than parameter count. Second, energy expenditure and reasoning performance are only weakly coupled, as confirmed by Spearman rank correlation analysis ($r_s = 0.109$, $p = 0.386$). Third, task category and difficulty level introduce substantial and model-dependent variation in both energy demand and performance, with cross-lingual performance variance amplifying at higher difficulty levels. Fourth, language choice acts as a measurable deployment parameter as follows: Romance languages on average achieve lower energy consumption than English across multiple models, while model efficiency rankings shift across languages, yielding language-dependent Pareto-optimal frontiers. We formalize these trade-offs through multi-objective Pareto analysis and introduce a composite AI Energy Score metric that captures reasoning quality per unit of energy. Of the 65 evaluated configurations, only four are Pareto-optimal, three Mistral-7B configurations at the low-energy extreme and one Phi-4-mini-instruct configuration at the high-performance end, while three of the five models are entirely dominated across all language configurations. These findings provide actionable guidelines for energy-aware model selection in multilingual deployments and support the integration of AI Energy Scores as a standard complementary criterion in LLM evaluation frameworks.

Academic Editor: Arkaitz Zubiaga

Received: 12 February 2026
Revised: 20 March 2026
Accepted: 25 March 2026
Published: 27 March 2026

Keywords: large language models; energy efficiency; multilingual evaluation; sustainable AI; GPU energy consumption; AI energy scores

1. Introduction

Large Language Models (LLMs) have become foundational components of modern intelligent systems, underpinning a wide range of applications from conversational agents and decision-support platforms [1] to mathematical discovery [2], clinical decision-making [3], and educational assistance [4]. As these models transition from research prototypes to production-grade infrastructure, their deployment footprint has expanded dramatically as follows: recent estimates indicate that AI-related electricity consumption could reach 4.5% of global electricity generation by 2030 [5], with LLM inference accounting for a substantial and growing share of this demand [6]. This rapid scaling has shifted attention from purely algorithmic performance toward operational considerations that include computational cost, latency, and, critically, energy consumption.

Despite the urgency of this challenge, current LLM evaluation practices remain overwhelmingly accuracy-centric. Established benchmarks such as MMLU [7], BIG-bench [8], and the MATH dataset [9] evaluate models exclusively on task performance, while widely adopted leaderboards including the Open LLM Leaderboard [10] and HELM [11] rank models without any consideration of the energy required to produce their outputs. This evaluation gap has created a systematic blind spot, outlined as follows: model selection in practice is guided almost entirely by accuracy, with energy efficiency treated as a secondary concern or ignored altogether [12]. As regulatory frameworks begin to address the environmental impact of AI, notably the European Union's AI Act, which mandates reporting the resource consumption and energy efficiency for high-risk AI systems [13], the absence of energy-aware evaluation standards becomes increasingly untenable.

Several initiatives have begun to address this gap. Strubell et al. [14] provided an early and influential quantification of the energy costs of training large NLP models, demonstrating that a unique training run can emit as much carbon as five automobiles over their lifetimes. Subsequent work by Lacoste et al. [15] introduced the Machine Learning CO_2 Impact calculator, while the CodeCarbon framework [16] enabled researchers to track emissions during both training and inference. More recently, the AI Energy Score initiative [17] has proposed standardized energy efficiency ratings for AI models based on watt-hours per 1000 queries under controlled conditions, providing a comparable metric across models and task types. However, these tools and frameworks have primarily been demonstrated on relatively simple tasks (e.g., text generation or classification) and rarely account for multilingual deployment scenarios, where energy consumption profiles may diverge substantially from English-only evaluations.

The multilingual dimension is particularly important given the global reach of contemporary LLM deployments. While benchmarks have gradually expanded beyond English [18,19], systematic evaluation of how language choice affects both performance and energy consumption remains scarce. Existing multilingual assessments typically focus on accuracy metrics alone, overlooking the possibility that tokenization strategies, vocabulary coverage, and morphological complexity may introduce language-dependent variation in inference cost [20]. In practice, LLMs are expected to serve diverse linguistic populations under identical infrastructure constraints, yet the assumption that energy consumption scales uniformly across languages has not been empirically validated. This omission is consequential as follows: if certain languages consistently incur higher energy costs for

https://doi.org/10.3390/electronics15071395

comparable performance, deployment decisions informed solely by English-language benchmarks may be systematically suboptimal.

Recent advances in model architectures have further complicated the energy–performance landscape. The field has witnessed a proliferation of design strategies aimed at improving efficiency without sacrificing capability. Grouped-Query Attention (GQA) reduces memory bandwidth by sharing key-value heads [21], while State Space Models (SSMs) such as Mamba replace the quadratic attention mechanism with recurrent structures that scale linearly with sequence length [22]. Compact, optimization-oriented models in the 3–4 billion parameter range now aim to match or exceed the reasoning capabilities of much larger counterparts [23,24]. Meanwhile, larger and more instruction-tuned models continue to expand in scale, often incurring substantially higher energy demands without consistently delivering proportional improvements in task quality [14,25]. Understanding how these architectural choices translate into performance-per-energy outcomes across languages and task complexities is essential for informed model selection, yet systematic, statistically grounded analyses that jointly consider these dimensions remain limited.

The question of reliability adds a further layer of complexity. Zhou et al. [25] demonstrated that larger and more instructable language models do not always exhibit improved reliability, and they may in fact become less consistent as scale increases. Complementary findings on output stability [26] and sensitivity to prompt phrasing [27,28] indicate that consistency across semantically equivalent inputs remains a significant challenge. From a deployment perspective, a model that achieves high peak accuracy but exhibits erratic behavior across languages or task variants may be less suitable than a more consistent alternative, even if the latter's average performance is lower. These observations motivate evaluation frameworks that consider not only peak performance but also operational characteristics such as energy efficiency, behavioral stability, and robustness across linguistic contexts.

Multi-objective optimization provides a natural framework for reasoning about these trade-offs. In settings where accuracy, energy consumption, and language coverage represent competing objectives, no model can be expected to dominate across all criteria simultaneously. Pareto frontier analysis [29] offers a principled approach to identifying the set of non-dominated configurations (those for which no alternative achieves strictly better performance on all objectives) and has been applied in related contexts including neural architecture search and hardware-aware model design [30]. Extending this perspective to the joint energy–performance–language evaluation space enables practitioners to make informed deployment decisions that explicitly balance sustainability constraints against accuracy requirements, rather than treating energy as an afterthought.

In this work, we address the gap between accuracy-centric benchmarking and the operational realities of multilingual LLM deployment by presenting a comprehensive energy–performance evaluation of five instruction-tuned Large Language Models across thirteen typologically diverse languages. The evaluated models span three distinct architectural families (standard Transformer, Transformer with Grouped-Query Attention, and State Space Model) and range from 3.8 to 7.3 billion parameters, covering a representative spectrum of current open-weight LLM designs. Using a fixed set of 79 reasoning-oriented problems and a controlled hardware configuration based on NVIDIA H200 GPUs, we measure GPU-level energy consumption alongside multiple performance indicators including overall task score, step accuracy, and consistency. This paired experimental design, encompassing 65 model–language configurations and over 5100 individual inference runs, allows us to isolate model- and language-dependent effects while ensuring strict comparability across all experimental conditions.

Our analysis extends substantially beyond aggregate energy and performance summaries. We decompose results by task category and difficulty level to reveal how energy–

performance trade-offs vary across problem types, quantify the statistical significance of observed differences using non-parametric tests with appropriate corrections for multiple comparisons, and characterize the strength of energy–performance coupling through correlation analysis. Furthermore, we examine language-dependent deviations from English baselines across all models and metrics, revealing that language choice acts as a measurable and potentially tunable deployment parameter with respect to energy efficiency.

The contributions of this paper are as follows:

1. We provide a systematic multilingual benchmark that jointly evaluates reasoning performance and GPU-level energy consumption of LLMs under controlled inference conditions, covering five architectural variants across thirteen languages and multiple task difficulties.
2. We quantify the extent to which energy efficiency and reasoning performance diverge across models, languages, and task categories, demonstrating that higher energy expenditure does not reliably translate into superior task outcomes.
3. We introduce and apply a composite AI Energy Score metric that captures reasoning quality per unit of energy, and we use Pareto frontier analysis to identify language-dependent optimal configurations in the multi-objective energy–performance space.
4. We provide actionable deployment guidelines based on empirical evidence, demonstrating that only two of the five evaluated models, Mistral-7B-Instruct-v0.1 and Phi-4-mini-instruct, contribute Pareto-optimal configurations and that language-aware model selection can yield substantial energy savings without performance degradation.

Together, these contributions support a shift from accuracy-only benchmarks toward multi-objective evaluation practices that account for the energy cost of intelligence, the linguistic diversity of real-world deployments, and the growing regulatory and societal demand for sustainable AI systems.

The remainder of this paper is organized as follows. Section 2 reviews related work on energy measurement in machine learning, multilingual LLM evaluation, multi-objective model selection, and AI Energy Score frameworks. Section 3 describes the experimental setup, including the evaluated models, the multilingual dataset, hardware configuration, performance and energy metrics, statistical analysis methods, and Pareto frontier construction. Section 4 presents the main findings, covering energy consumption profiles across models, task performance outcomes, category- and difficulty-level decompositions, language sensitivity analysis, correlation structure, and Pareto-optimal configurations. Section 5 interprets the results in the context of sustainable LLM deployment. Finally, Section 6 summarizes the key contributions and outlines directions for future work.

2. Related Work

This section reviews the four bodies of literature most relevant to the present study, as follows: energy measurement in machine learning systems, multilingual evaluation of large language models, multi-objective model selection and Pareto-based methods, and the emerging AI Energy Score framework. While each of these areas has seen significant recent progress, their intersection, joint energy–performance evaluation under multilingual, multi-objective conditions, remains largely unexplored.

Energy Measurement in Machine Learning Systems

The environmental cost of training and deploying machine learning models first received broad attention through the work of Strubell et al. [14], who estimated the carbon emissions of training large NLP models and found that one training run of a Transformer-based architecture could rival the lifetime emissions of an automobile. This work catalyzed a line of research focused on making energy consumption visible as a first-class metric in

the machine learning workflow. Schwartz et al. [12] subsequently introduced the distinction between "Red AI", which pursues accuracy gains regardless of computational cost, and "Green AI", which explicitly considers efficiency alongside performance. Their call for reporting floating-point operations, runtime, and energy as standard experimental practice has been widely cited but only partially adopted.

On the measurement side, several tools and methodologies have been proposed. Henderson et al. [31] provided recommendations for the systematic reporting of energy and carbon footprints, identifying key sources of variability including hardware heterogeneity, software framework overhead, and measurement granularity. Lacoste et al. [15] introduced the Machine Learning CO_2 Impact calculator, which estimates carbon emissions from self-reported training details and regional electricity carbon intensity. The CodeCarbon framework [16] (https://codecarbon.io, accessed on 1 February 2026) enables real-time tracking of GPU, CPU, and memory power draw during both training and inference, integrating with standard Python workflows (https://www.python.org, accessed on 1 February 2026). More recently, Luccioni et al. [32] conducted a systematic comparison of the energy consumption and carbon emissions of multiple LLM architectures across diverse inference tasks, demonstrating that multi-purpose generative models consume substantially more energy than task-specific alternatives and that model size alone is a poor predictor of energy efficiency.

Despite these advances, the focus of energy measurement has predominantly been on training-time costs, with inference-time energy consumption receiving comparatively less attention [33,34]. This imbalance is significant because, for deployed LLM-based systems, inference accounts for the dominant share of cumulative energy expenditure over the model's operational lifetime [6]. Furthermore, most existing studies report aggregate energy values without decomposing them by task type, difficulty, or language, limiting the actionability of their findings for practitioners who must select models for specific deployment scenarios.

García-Martín et al. [35] provided an early survey of methods for estimating energy consumption in machine learning, covering hardware counters, software profiling, and model-based estimation techniques. Their taxonomy remains useful for contextualizing the measurement choices made in empirical studies, including the GPU-level NVML-based monitoring approach adopted in the present work. More recently, Bannour et al. [36] evaluated the reliability of software-based energy measurement tools for NLP workloads, finding that GPU power sampling at sub-second granularity provides robust and reproducible estimates when coupled with controlled warm-up and cooldown protocols, a finding that informs our experimental design.

The evaluation of language models has historically been dominated by English-centric benchmarks. GLUE [18] and SuperGLUE [19] established the paradigm of aggregate leaderboard-based evaluation for natural language understanding, but were limited to English. Subsequent efforts extended this paradigm to the multilingual setting: XTREME [37] and XGLUE [38] introduced cross-lingual benchmarks covering up to 40 languages across tasks such as classification, sequence labeling, question answering, and sentence retrieval. These benchmarks revealed substantial performance gaps between English and low-resource languages, motivating research on cross-lingual transfer and multilingual model pretraining [39].

More recently, evaluation has expanded to cover reasoning capabilities in multilingual settings. Ahuja et al. [40] evaluated multiple LLMs across a broad set of languages on the MEGA benchmark, finding that performance degradation in non-English languages is not uniform but depends on the interaction between model architecture, pretraining data composition, and task type. Lai et al. [41] assessed ChatGPT (GPT-3.5) across a diverse set

of NLP tasks and languages, reporting that, while frontier models achieve near-parity with English on high-resource languages, substantial gaps persist for low-resource and typologically distant languages. The MMLU benchmark [7] has been translated into multiple languages [41], but evaluations on translated benchmarks typically report only accuracy metrics without considering the computational or energy cost of multilingual inference.

A critical but often overlooked factor in multilingual evaluation is the role of tokenization. Rust et al. [42] demonstrated that tokenizer quality, measured by fertility (the average number of tokens per word) and the proportion of unknown tokens, varies dramatically across languages for the same model, directly affecting both sequence length and computational cost. Languages with higher fertility require longer input sequences, leading to proportionally higher memory usage, latency, and energy consumption during inference. Petrov et al. [43] further quantified this "language modeling tax", showing that speakers of low-resource languages face systematically higher costs per unit of semantic content. These findings motivate our analysis of language-dependent energy consumption, which can be partially attributed to tokenization efficiency, but it also reflects deeper interactions between the model's learned representations and the linguistic structure of the input.

Despite this growing body of work on multilingual accuracy evaluation, no prior study has systematically analyzed how language choice affects inference-time energy consumption, nor has any benchmark jointly reported energy and performance metrics across a typologically diverse set of languages. The present work addresses this gap directly.

The AI Energy Score initiative [17], led by a consortium including Hugging Face, proposes a standardized framework for rating the energy efficiency of AI models. The methodology measures GPU energy consumption in watt-hours per 1000 queries under controlled conditions, using fixed hardware configurations and standardized task workloads to ensure comparability across models. The resulting scores provide an efficiency rating analogous to energy labels for household appliances, enabling stakeholders to incorporate energy considerations into model selection decisions.

The AI Energy Score builds on several antecedent efforts to standardize sustainability reporting in machine learning. The ML CO_2 Impact calculator [15] established the principle of estimating emissions from computational metadata, while CodeCarbon [16] provided tooling for real-time measurement. Henderson et al. [31] articulated a set of best practices for reporting energy consumption, emphasizing the need for controlled experimental conditions, hardware specification, and statistical rigor. Luccioni et al. [32] extended this line of work by systematically comparing the inference-time energy consumption of 88 models across ten tasks, producing one of the most comprehensive energy profiles of the generative AI landscape to date, and finding that energy consumption varies by up to two orders of magnitude across model–task combinations.

Regulatory developments have added urgency to these standardization efforts. The European Union's AI Act [13] includes provisions for reporting the resource consumption and energy efficiency of AI systems, particularly those classified as high-risk. While the implementing regulations are still being finalized, the direction is clear, as follows: energy efficiency is transitioning from a voluntary best practice to a regulatory requirement, at least within the European Union. This regulatory context strengthens the case for evaluation frameworks that integrate energy metrics alongside traditional performance measures.

However, existing applications of the AI Energy Score and related frameworks have several limitations that the present study aims to address. First, most evaluations have been conducted on relatively simple tasks such as text generation, summarization, or classification, rather than on complex, multi-step reasoning tasks where energy consumption profiles may differ substantially. Second, multilingual evaluation has been largely absent as follows: energy scores are typically reported for English-language tasks only, despite the

global deployment context of most LLM-based systems. Third, the relationship between energy efficiency and task performance, whether models that consume more energy also produce better outputs, has not been formally characterized through statistical analysis or multi-objective optimization. Our work addresses all three limitations by applying the AI Energy Score methodology to a multilingual reasoning benchmark, formally testing the energy–performance coupling, and situating the results within a Pareto-optimal framework that enables principled model selection under competing constraints.

3. Materials and Methods

This section describes the experimental design used to evaluate the energy–performance characteristics of Large Language Models across multiple languages. The methodology ensures strict comparability across models and languages by controlling for hardware, workload, evaluation protocol, and statistical analysis, following best practices for the empirical evaluation of AI systems [12,31,36].

3.1. Evaluated Models

We evaluate five instruction-tuned Large Language Models that represent three distinct architectural families and span a range of parameter counts, context window lengths, and optimization strategies. Table 1 summarizes the key specifications. The models were selected to cover the architectural diversity encountered in contemporary open-weight LLM deployments while remaining within the 3–8 billion parameter range that is most relevant for cost-sensitive and edge-adjacent inference scenarios [21].

Table 1. Model specifications and architectural characteristics of the five evaluated LLMs.

Model	Params (B)	Architecture	Context (K)	FP16 (GB)	Vocab Size
Falcon-Mamba-7B	7.27	SSM (Mamba)	8 †	∼14	65,024
Mistral-7B-Instruct-v0.1	7.24	Transformer+GQA	8	∼14	32,000
Phi-3-mini-128k-instruct	3.82	Transformer	128	∼8	32,064
Phi-4-mini-instruct	3.84	Transformer+GQA	128	∼8	32,064
Qwen2-7B-Instruct	7.07	Transformer+GQA	32	∼14	151,936

† Training sequence length used for positional encoding; Mamba-based models use a fixed-size recurrent state and do not use an attention window at inference. GQA = Grouped-Query Attention; SSM = State Space Model. FP16 denotes half-precision (16-bit) floating-point model weights. Reported sizes exclude KV cache, activations, optimizer state, and runtime overhead.

Architectural Families

The evaluated models belong to three architectural categories with fundamentally different computational profiles as follows:

- Standard Transformer. Phi-3-mini-128k-instruct employs the canonical multi-head self-attention mechanism [44], with full key-value (KV) heads at every layer. This architecture provides maximum representational flexibility but incurs quadratic memory and compute scaling with sequence length due to the attention matrix.
- Transformer with Grouped-Query Attention (GQA). Mistral-7B-Instruct-v0.1, Phi-4-mini-instruct, and Qwen2-7B-Instruct adopt GQA [21,45], in which multiple query heads share a smaller number of key-value heads. This reduces the KV cache memory footprint and the memory bandwidth required during autoregressive decoding, yielding lower latency and potentially lower energy consumption per generated token without substantial degradation in representational capacity.
- State Space Model (SSM). Falcon-Mamba-7B is based on the Mamba architecture [22], which replaces the attention mechanism entirely with a selective state space model. Mamba processes sequences in linear time with respect to sequence length and maintains a fixed-size recurrent state, eliminating the KV cache altogether. This architectural

choice is expected to yield different energy consumption patterns, particularly for long sequences, compared to attention-based alternatives.

An important and often overlooked factor influencing both performance and energy consumption in multilingual settings is the tokenizer vocabulary. As shown in Table 1, vocabulary sizes vary substantially across models, from 32,000 tokens (Mistral) to 151,936 tokens (Qwen2). This variation has direct consequences for multilingual efficiency.

Models with smaller vocabularies tend to exhibit higher *fertility*, the average number of tokens required to encode a given word or morpheme [42], for languages that are underrepresented in their training data. Higher fertility translates to longer input sequences for the same semantic content, increasing the number of forward-pass steps during autoregressive generation and, consequently, the total energy consumed per query. Conversely, models with larger and more multilingual vocabularies (such as Qwen2, which was designed with extensive CJK coverage) may achieve lower fertility on typologically diverse languages, potentially reducing per-query energy costs for those languages at the expense of a larger embedding matrix [43].

This interaction between vocabulary design and multilingual energy consumption motivates our language-level analysis of energy metrics (Section 4), where we examine whether observed differences in energy consumption across languages can be partially attributed to tokenization efficiency.

All models were executed using the same inference framework and configuration, without any task-specific fine-tuning, to isolate architectural and implementation-level differences. Model inference was performed via a vLLM 0.5.0 HTTP endpoint [46] with PyTorch 2.3.0 [47] as the backend, using low-temperature sampling (temperature $=$ 0.2, max_tokens $=$ 300, batch size = 1 per API call). Each of the 79 problems was queried three times independently (`runs = 3`); the mean score across the three runs constitutes the per-problem score used in all performance analyses, while the standard deviation across runs directly operationalizes the consistency metric C (Section 3.4), providing an empirical measure of output stability under repeated inference. Prior to the energy measurement window, a single warmup query was issued on the first problem followed by a 2-s GPU stabilization interval; neither is included in the energy accounting [36]. This low-temperature, multi-run design approximates near-deterministic generation while retaining sufficient stochasticity to measure output stability, and is representative of production configurations for reasoning-oriented inference tasks. This evaluation focuses exclusively on inference-time behavior, which represents the dominant cost factor in most production deployments [6,14].

3.2. Multilingual Reasoning Dataset

The evaluation dataset consists of a fixed set of 79 reasoning-oriented problems, originally introduced in [48] for the assessment of semantic invariance in LLMs. Each problem requires multi-step reasoning and produces a structured output that enables fine-grained evaluation at both the overall and intermediate-step levels.

3.2.1. Languages

The problem set was translated into the following thirteen languages: Arabic (ar), Catalan (ca), German (de), English (en), Spanish (es), Basque (eu), French (fr), Italian (it), Luxembourgish (lb), Portuguese (pt), Russian (ru), Simplified Chinese (zh-cn), and Traditional Chinese (zh-tw). This selection spans three language families (Indo-European, Afro-Asiatic, and Sino-Tibetan) and one language isolate (Basque), three writing systems (Latin, Arabic, and Chinese), and a range of resource levels from high-resource (English, German, and French) to low-resource (Luxembourgish and Basque). Translations were pro-

duced to preserve the semantic structure and difficulty of the original problems, enabling paired comparisons across languages.

By using an identical problem set across all models and languages, the experimental design controls for task variability and allows language-specific effects on both performance and energy consumption to be analyzed directly. This multilingual setup reflects realistic deployment scenarios in which LLMs are expected to operate consistently across diverse linguistic contexts [20] and addresses concerns about the predominance of English-centric evaluation [11,40].

3.2.2. Task Categories

The 79 problems are organized into multiple task categories that reflect different reasoning modalities, including arithmetic and numerical reasoning, logical deduction, pattern recognition, commonsense inference, and multi-step word problems. This categorical structure enables decomposition of both performance and energy metrics by problem type, revealing whether certain reasoning modalities are systematically more energy-intensive or exhibit greater cross-lingual variation.

3.2.3. Difficulty Levels

Each problem is additionally annotated with a difficulty level, enabling analysis of how energy consumption and performance scale with task complexity. This dimension is particularly informative for understanding whether energy–performance coupling strengthens or weakens as problems become more demanding, and whether models exhibit different difficulty-sensitivity profiles.

The combination of categorical and difficulty annotations, applied uniformly across all thirteen languages, produces a rich factorial structure that supports the multi-level analyses reported in Section 4.

3.3. Hardware and Measurement Protocol

All experiments were conducted on an NVIDIA (Santa Clara, CA, USA) H200 GPU (141 GB HBM3e, 4.8 TB/s memory bandwidth, 989 TFLOPS FP16 Tensor Core peak throughput) to eliminate hardware-induced variability across models and languages. One GPU was used for all inference runs, with no model parallelism or tensor sharding, ensuring that energy measurements reflect the full computational cost of each model on identical hardware.

3.3.1. Energy Measurement

GPU power consumption was monitored throughout execution using the NVIDIA Management Library (NVML), which provides access to the GPU's onboard power sensor. Power readings were sampled at sub-second intervals (approximately 100 ms granularity) and recorded as time-stamped traces for each experimental run. Total GPU energy consumption in watt-hours was computed by numerical integration of the power trace over the wall-clock execution time as follows:

$$E_{\text{GPU}} = \frac{1}{3600} \int_{t_0}^{t_1} P_{\text{GPU}}(t)\, dt \approx \frac{1}{3600} \sum_{o} P_o \cdot \Delta t_o \tag{1}$$

where P_o is the instantaneous GPU power draw at the o-th sample and Δt_o is the inter-sample interval. This trapezoidal integration approach follows recommendations for reliable GPU energy estimation in machine learning workloads [31,35,36].

3.3.2. Execution Parameters

The same sampling temperature (temperature = 0.2), maximum generation length (max_tokens = 300), number of runs per problem (`runs = 3`), and batch size (1 per API call) were applied uniformly across all model–language configurations. These controls ensure that observed differences in energy consumption are attributable to model architecture and language characteristics rather than to incidental variation in execution parameters. The max_tokens = 300 ceiling additionally bounds the maximum number of autoregressive steps per problem, partially controlling for output-length confounding across models.

3.3.3. Energy Metrics

The following energy-related quantities are reported for each model–language configuration:

- E_{total}: Total GPU energy consumption (Wh) for the complete set of 79 problems.
- E_{1000}: Watt-hours per 1000 queries, computed as $E_{1000} = E_{\text{total}} \times (1000/N_{\text{prompts}})$, where $N_{\text{prompts}} = 79$.
- $\bar{P}$: Average GPU power draw (W) during inference.
- $P_{\max}$: Peak GPU power draw (W) observed during inference.
- T_{wall}: Wall-clock execution time (s) for the complete run.
- E_{prompt}: Energy per prompt, $E_{\text{prompt}} = E_{\text{total}}/N_{\text{prompts}}$.

The primary energy metric used throughout this study is E_{1000} (watt-hours per 1000 queries), following the AI Energy Score convention [17]. This normalization enables direct comparison of energy efficiency across models regardless of the number of evaluation prompts.

3.4. Performance Metrics

Model performance was assessed using three complementary metrics that capture different dimensions of reasoning quality, following evaluation practices established for chain-of-thought reasoning [23] and the behavioral testing of language models [26,49].

3.4.1. Overall Task Score

The overall task score $S_{\text{overall}} \in [0, 1]$ reflects the end-to-end problem-solving success rate, computed as the fraction of problems for which the model produces a fully correct final answer. This metric captures the aggregate reasoning capability of the model on the evaluation set and serves as the primary performance indicator in our energy–performance analyses.

3.4.2. Step Accuracy

Step accuracy $S_{\text{step}} \in [0, 1]$ measures the correctness of intermediate reasoning steps, independent of whether the final answer is correct. For multi-step problems, each intermediate computation or logical deduction is evaluated against the reference solution. This metric captures the quality of the reasoning process itself and is particularly informative for distinguishing models that arrive at correct answers through sound reasoning from those that produce correct final answers despite intermediate errors [50].

3.4.3. Consistency

Consistency $C \in [0, 1]$ is the mean standard deviation of per-problem scores across the three independent inference runs as follows:

$$C = \frac{1}{N} \sum_{o=1}^{N} \sigma_o, \quad \sigma_o = \text{std}\left(s_o^{(1)}, s_o^{(2)}, s_o^{(3)}\right) \tag{2}$$

where $s_o^{(k)}$ is the score of the k-th run on the o-th problem and $N = 79$. Lower values indicate that the model produces similar outputs across repeated queries at temperature $= 0.2$, while higher values indicate greater sensitivity to the stochastic variation introduced by sampling. This metric is an empirically measured quantity derived directly from the three-run experimental design [24,26], and it is relevant to deployment reliability since inconsistent models impose higher verification costs on downstream applications.

All three metrics were computed uniformly across models and languages using the same evaluation scripts and scoring criteria, ensuring fairness and reproducibility. Recent work has shown that accuracy-metric evaluations can obscure important behavioral differences across models [8,11]; the use of complementary metrics mitigates this risk.

The absolute values of C reported in this study are small (ranging from 0.046 to 0.065 across models, see Section 4.2), which reflects the near-deterministic character of low-temperature sampling at temperature $= 0.2$; for the majority of problems, the model produces near-identical outputs across the three runs, resulting in per-problem standard deviations σ_o close to zero whose mean is accordingly small. Higher values of C therefore indicate models that are comparatively more sensitive to the residual stochasticity at this temperature, rather than models that are inconsistent in an absolute sense; the metric is most informative when comparing models against each other rather than against an external absolute scale.

3.5. AI Energy Score Definition

To capture the joint efficiency of reasoning quality and energy expenditure in one comparable quantity, we define the AI Energy Score as the ratio of overall task performance to energy consumption per 1000 queries, scaled for readability, as follows:

$$\text{AI Energy Score} = \frac{S_{\text{overall}}}{E_{1000}} \times 10^3 \tag{3}$$

where S_{overall} is the overall task score (Section 3.4) and E_{1000} is the energy consumption in watt-hours per 1000 queries (Section 3.3). The 10^3 scaling factor is applied to yield values in a convenient numerical range. Higher AI Energy Scores indicate more favorable energy–performance trade-offs, i.e., greater reasoning quality per unit of energy expenditure.

This formulation aligns with the AI Energy Score initiative's emphasis on watt-hours per 1000 queries as the standard energy unit [17], and it extends it by incorporating a task performance numerator, enabling the direct comparison of models that differ in both energy consumption and reasoning capability.

We additionally define the *performance efficiency* as follows:

$$\eta = \frac{S_{\text{overall}}}{E_{1000}} \tag{4}$$

which is the unscaled ratio used in internal analyses and Pareto frontier construction. Where language-specific values are required, both S_{overall} and E_{1000} are evaluated on the corresponding language variant of the evaluation set, yielding language-specific AI Energy Scores as follows:

$$\text{AI Energy Score}(\ell) = \frac{S_{\text{overall}}(\ell)}{E_{1000}(\ell)} \times 10^3, \quad \ell \in \{\text{ar}, \text{ca}, \ldots, \text{zh-tw}\}. \tag{5}$$

This per-language formulation enables the analysis of language-dependent efficiency frontiers reported in Section 4.

Design Rationale and Relationship to the AI Energy Score Initiative

The formulation in Equation (3) is grounded in, and intentionally aligned with, the AI Energy Score initiative [17], which standardizes GPU energy measurement as watt-hours per 1000 queries under controlled hardware conditions. That initiative reports a *pure energy* figure (Wh/1000 queries) without a performance numerator, thereby enabling cross-model energy comparisons independently of task. The present work extends this framework to a *composite* efficiency metric by introducing S_{overall} as the numerator, so that models can be ranked on the joint energy–performance trade-off rather than energy alone.

Three design choices in this formulation merit explicit justification. First, S_{overall} was selected as the sole performance numerator rather than a weighted combination of the three reported metrics (S_{overall}, S_{step}, C) for the following two reasons: (1) the overall task score is the aggregate end-to-end success rate and is the primary performance indicator most directly comparable to accuracy figures on existing LLM leaderboards; and (2) the three metrics exhibit substantial collinearity in our data (models with higher overall scores tend to also exhibit higher step accuracy and consistency, as shown in Section 4.2, so a weighted combination would introduce redundancy and an arbitrary weighting scheme without materially changing the efficiency ranking. We did consider a composite numerator of the form $\tilde{S} = w_1 S_{\text{overall}} + w_2 S_{\text{step}} + w_3 C$, but we rejected it because any choice of weights (w_o) is domain-specific and would hinder cross-study comparability.

Second, the scaling factor 10^3 is applied purely for readability as follows: the unscaled ratio $\eta = S_{\text{overall}}/E_{1000}$ lies in the range $[7 \times 10^{-4}, 5 \times 10^{-3}]$ across our 65 configurations, while the scaled AI Energy Score ($\eta \times 10^3$) maps these into the range $[0.7, 5.0]$. All Pareto frontier construction and statistical comparisons use the unscaled η; the 10^3-scaled value appears only in the summary table and efficiency ranking figure presented in Sections 4.2 and 4.7 for presentation purposes and does not affect any ranking.

Third, because the denominator E_{1000} follows the exact Wh-per-1000-queries convention of the AI Energy Score initiative, the values reported here are directly comparable to entries on the Hugging Face AI Energy Score leaderboard once a task-specific performance normalization is agreed upon by the broader community. This alignment is a deliberate design choice; it positions the composite metric as a natural extension of an emerging community standard rather than a bespoke measure, facilitating future adoption in multilingual evaluation frameworks.

3.6. Statistical Framework

To assess the statistical significance and practical magnitude of differences across models and languages, we employ a suite of non-parametric tests suited to the paired, repeated-measures design of the experiment. All analyses were performed using SciPy [51] and statsmodels (https://www.statsmodels.org, accessed on 1 February 2026).

3.6.1. Friedman Test

For each metric (energy and performance), we apply the Friedman test [52] to assess whether there are significant differences across the five models when treating the thirteen languages as paired blocks. The Friedman test is the non-parametric counterpart of repeated-measures ANOVA, being appropriate when the assumption of normality cannot be guaranteed. The null hypothesis states that all models have identical distributions for the metric in question; rejection indicates that at least one model differs systematically.

3.6.2. Pairwise Wilcoxon Signed-Rank Tests with Holm Correction

Where the Friedman test indicates a significant overall effect, we conduct pairwise comparisons between all $\binom{5}{2} = 10$ model pairs using the Wilcoxon signed-rank test [53],

which evaluates whether the distribution of paired differences between two models is symmetric around zero. To control the family-wise error rate across the ten comparisons, p-values are adjusted using the Holm step-down procedure [54], which provides a uniformly more powerful correction than the classical Bonferroni method while maintaining strong control of Type I error.

3.6.3. Paired Cohen's d

To quantify the practical magnitude of observed differences beyond statistical significance, we compute paired Cohen's d [55] for each model pair and metric as follows:

$$d = \frac{\bar{D}}{s_D} \tag{6}$$

where $\bar{D}$ is the mean of the paired differences across languages and s_D is their standard deviation. Following standard conventions, effect sizes are interpreted as small ($|d| \approx 0.2$), medium ($|d| \approx 0.5$), or large ($|d| \geq 0.8$).

3.6.4. Spearman Rank Correlation

To characterize the relationship between energy consumption and task performance across the full set of model–language configurations, we compute Spearman's rank correlation coefficient as follows [56]:

$$r_s = 1 - \frac{6 \sum_{o=1}^{n} d_o^2}{n(n^2 - 1)} \tag{7}$$

where d_o is the difference in ranks of the o-th observation on the two variables and n is the total number of model–language data points. This non-parametric correlation measure is robust to outliers and does not assume linearity, making it well-suited for detecting monotonic but potentially non-linear relationships between energy and performance. We report both pooled correlations (across all 65 configurations) and per-model correlations to distinguish global trends from model-specific patterns.

3.7. Pareto Frontier Analysis

To capture the multi-objective nature of model selection, we perform Pareto frontier analysis over the joint energy–performance space. Each model–language configuration $c = (m, \ell)$ is characterized by a two-dimensional objective vector as follows:

$$\mathbf{f}(c) = \left(E_{1000}(c),\ S_{\text{overall}}(c) \right) \tag{8}$$

where the goal is to *minimize* E_{1000} (energy consumption) and *maximize* S_{overall} (task performance).

3.7.1. Dominance Criterion

A configuration c_1 is said to *dominate* another configuration c_2, written $c_1 \succ c_2$, if and only if,

$$E_{1000}(c_1) \leq E_{1000}(c_2) \ \text{ and } \ S_{\text{overall}}(c_1) \geq S_{\text{overall}}(c_2) \tag{9}$$

with at least one strict inequality. A configuration is *Pareto-optimal* (or *non-dominated*) if no other configuration in the evaluation set dominates it [29].

3.7.2. Frontier Construction

The Pareto frontier is the set of all non-dominated configurations, as follows:

$$\mathcal{F} = \left\{ c \in \mathcal{C} \mid \nexists\, c' \in \mathcal{C} : c' \succ c \right\} \tag{10}$$

where $\mathcal{C}$ is the set of all 65 model–language configurations (5 models $\times$ 13 languages). We identify $\mathcal{F}$ using a simple non-dominated sorting procedure as follows: configurations are sorted by ascending E_{1000}, and a configuration is added to the frontier if and only if its S_{overall} exceeds that of all previously added frontier members.

The Pareto frontier is computed both globally (across all models and languages jointly) and per-model (across languages for a fixed model), yielding, respectively, a global efficiency frontier and model-specific language frontiers. Pareto optimality is defined exclusively with respect to E_{1000} and S_{overall}; step accuracy and consistency are reported for completeness but are not included in the dominance criterion, as the overall task score captures aggregate reasoning performance.

This multi-objective perspective enables practitioners to navigate energy–performance trade-offs explicitly rather than collapsing them into a scalar metric, and to identify which models and languages occupy favorable regions of the efficiency space under different priority weightings.

4. Results

This section presents the results of the multilingual energy–performance evaluation. We first analyze energy consumption profiles across models (Section 4.1), followed by task performance outcomes (Section 4.2). We then decompose results by task category (Section 4.3) and difficulty level (Section 4.4), examine language sensitivity and baseline deviations (Section 4.5), characterize the energy–performance correlation structure (Section 4.6), report AI Energy Scores and efficiency rankings (Section 4.7), and finally present the Pareto frontier analysis (Section 4.8). All results are based on paired comparisons across the thirteen languages to ensure statistical validity.

4.1. Energy Consumption Profiles

Substantial and statistically significant differences in energy consumption were observed across the five evaluated models, despite identical hardware, workloads, and execution parameters. Figure 1 summarizes the distribution of watt-hours per 1000 queries across languages for each model, revealing a clear stratification into three distinct energy regimes.

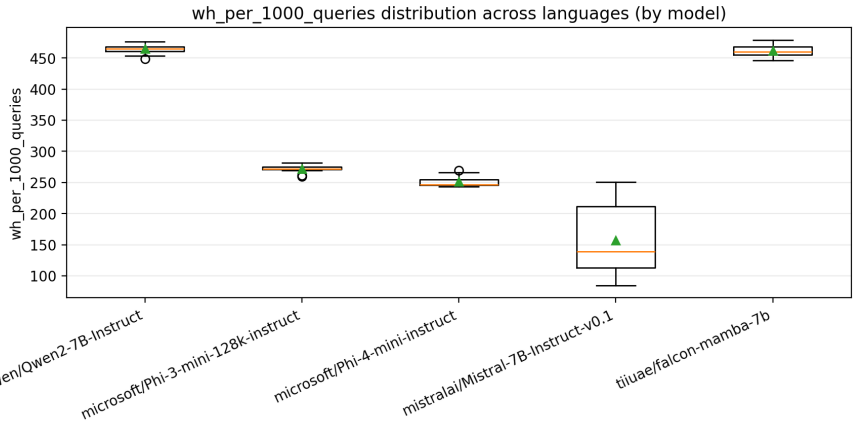

Figure 1. Distribution of energy consumption (Wh per 1000 queries) across languages for each evaluated model. Boxes indicate interquartile ranges; diamonds indicate means. Models are clearly stratified into low-energy (Mistral), mid-energy (Phi-3, Phi-4), and high-energy (Falcon-Mamba, Qwen2) regimes.

Mistral-7B-Instruct-v0.1 consistently exhibits the lowest energy consumption, with a mean of 157.3 $\pm$ 57.4 Wh per 1000 queries across languages. Its relatively high variance stems from a pronounced sensitivity to language-dependent sequence lengths, which we analyze further in Section 4.5. In contrast, Falcon-Mamba-7B (461.3 $\pm$ 10.2 Wh) and Qwen2-

7B-Instruct (464.6 ± 7.9 Wh) operate at the high-energy extreme, consuming approximately three times as much energy as Mistral under identical workloads. The Phi-family models occupy an intermediate regime as follows: Phi-4-mini-instruct at 251.3 ± 8.6 Wh and Phi-3-mini-128k-instruct at 272.1 ± 6.3 Wh. Notably, the compact Phi models (3.8B parameters) achieve substantially lower energy consumption than the 7B-parameter Falcon-Mamba and Qwen2, while, as shown in Section 4.2, delivering comparable or superior reasoning performance.

The heatmap in Figure 2 provides a fine-grained view of energy consumption across all model–language combinations, revealing both the dominant model-level effect and subtler language-dependent variation within each model.

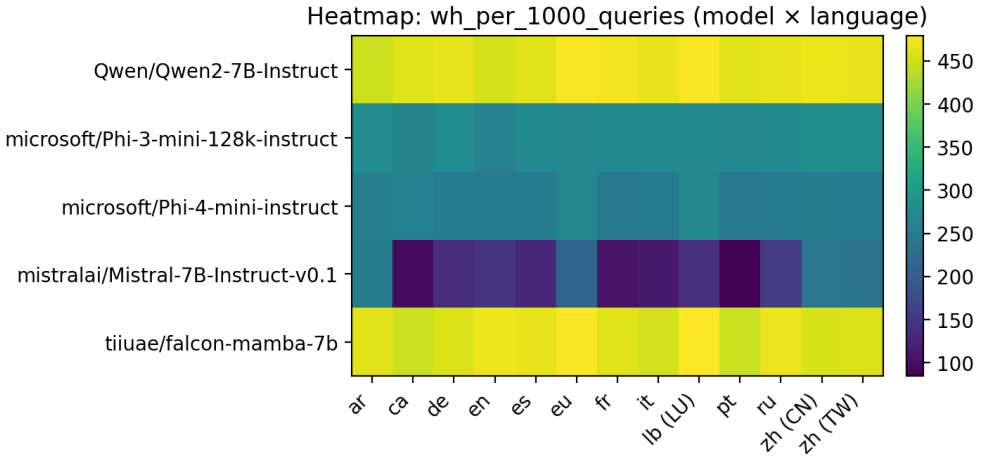

Figure 2. Heatmap of energy consumption (Wh per 1000 queries) across models (rows) and languages (columns). The dominant source of variation is the model, but within-model language-dependent differences are visible, particularly for Mistral-7B.

Statistical Significance

Friedman tests across the five models, using the thirteen languages as paired blocks, indicate highly significant differences for all energy and performance metrics. Table 2 reports the Friedman test statistic and p-value for each metric evaluated across all five models with thirteen language blocks. Energy metrics yield $\chi^2 = 49.42$ ($p = 4.78 \times 10^{-10}$), and performance metrics yield $\chi^2 \geq 29.66$ ($p \leq 5.74 \times 10^{-6}$), confirming that model-level differences in energy consumption, performance, and efficiency are systematic rather than incidental.

Table 2. Friedman test results across five models using thirteen languages as paired blocks. All metrics show highly significant differences, confirming systematic model-level effects on both energy and performance.

Metric	Friedman χ^2	p-Value	n (Blocks)
Wh per 1000 queries	49.42	4.78×10^{-10}	13
Total GPU energy (Wh)	49.42	4.78×10^{-10}	13
Wh per prompt	49.42	4.78×10^{-10}	13
Overall task score	46.58	1.86×10^{-9}	13
Step accuracy	43.32	8.87×10^{-9}	13
Consistency	29.66	5.74×10^{-6}	13
Performance efficiency	44.00	6.42×10^{-9}	13

Pairwise Wilcoxon signed-rank tests with Holm correction confirm that the majority of model pairs differ significantly in energy consumption. Table 3 reports the pairwise comparisons for Wh per 1000 queries, including Holm-corrected p-values and paired Cohen's d effect sizes. The three energy regimes (low: Mistral; mid: Phi-3, Phi-4; and high:

Falcon-Mamba, Qwen2) are statistically separable at $\alpha = 0.05$ after correction for multiple comparisons. Paired Cohen's d effect sizes between energy regimes are uniformly large ($|d| > 2.0$), indicating that the observed differences are not only statistically significant but also of substantial practical magnitude.

Table 3. Pairwise Wilcoxon signed-rank tests (Holm-corrected) for Wh per 1000 queries. Paired Cohen's d quantifies the practical magnitude of each pairwise difference across thirteen languages. Positive d indicates that model A consumes more energy than model B.

Model A	Model B	W	p_{Holm}	Reject	Cohen's d
Qwen2-7B	Phi-4-mini	0.0	0.0024	Yes	23.38
Phi-4-mini	Falcon-Mamba-7B	0.0	0.0024	Yes	−23.59
Qwen2-7B	Phi-3-mini	0.0	0.0024	Yes	22.48
Phi-3-mini	Falcon-Mamba-7B	0.0	0.0024	Yes	−15.17
Mistral-7B	Falcon-Mamba-7B	0.0	0.0024	Yes	−5.49
Qwen2-7B	Mistral-7B	0.0	0.0024	Yes	5.24
Phi-3-mini	Mistral-7B	0.0	0.0024	Yes	2.11
Phi-3-mini	Phi-4-mini	0.0	0.0024	Yes	1.82
Phi-4-mini	Mistral-7B	0.0	0.0024	Yes	1.69
Qwen2-7B	Falcon-Mamba-7B	30.0	0.3054	No	0.32

Figure 3 presents the mean energy consumption ranking across languages, providing a concise summary of the energy hierarchy.

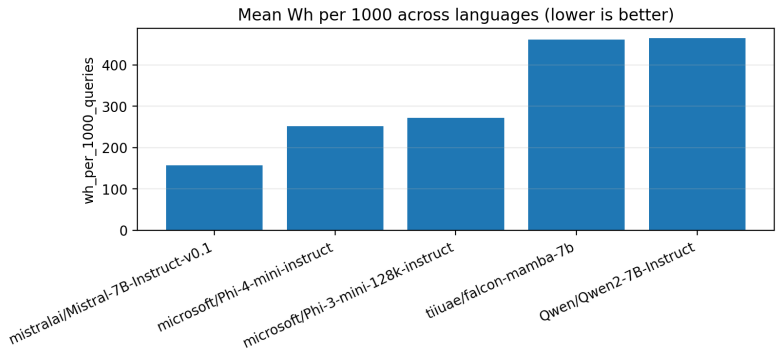

Figure 3. Mean energy consumption (Wh per 1000 queries) across languages, by model. Lower values are preferable.

4.2. Task Performance

Table 4 summarizes the mean energy consumption and performance metrics across all thirteen languages. The models exhibit a clear performance hierarchy that does not align with their energy ordering.

Table 4. Cross-model summary of energy consumption and performance metrics (mean ± std across 13 languages). The AI Energy Score is computed as the ratio of the cross-lingual mean overall task score to the cross-lingual mean energy consumption (Equation (3) applied to the tabulated averages), scaled by 10^3 for readability. Models are ordered by AI Energy Score (higher is better).

Model	Wh/1000q	Overall	Step Acc.	Consist.	AI En. Score ($\times 10^3$)
Mistral-7B-Instruct-v0.1	157.3 ± 57.4	0.332 ± 0.037	0.389 ± 0.029	0.046 ± 0.007	2.11
Phi-4-mini-instruct	251.3 ± 8.6	0.507 ± 0.070	0.660 ± 0.077	0.065 ± 0.015	2.03
Phi-3-mini-128k-instruct	272.1 ± 6.3	0.476 ± 0.070	0.652 ± 0.073	0.058 ± 0.008	1.75
Qwen2-7B-Instruct	464.6 ± 7.9	0.467 ± 0.046	0.639 ± 0.057	0.049 ± 0.009	1.01
Falcon-Mamba-7B	461.3 ± 10.2	0.384 ± 0.052	0.514 ± 0.048	0.057 ± 0.006	0.83

Phi-4-mini-instruct achieves the highest overall task score (0.507 ± 0.070) and step accuracy (0.660 ± 0.077), followed closely by Phi-3-mini-128k-instruct (0.476 ± 0.070) and Qwen2-7B-Instruct (0.467 ± 0.046). Falcon-Mamba-7B achieves moderate performance (0.384 overall), while Mistral-7B-Instruct-v0.1 occupies the lowest performance tier (0.332 overall). Crucially, Qwen2 achieves an overall score comparable to Phi-3 (0.467 vs. 0.476, $p = 0.305$) while consuming 70% more energy, indicating a substantial efficiency gap.

Figure 4 shows the distribution of overall task scores across languages for each model, and Figure 5 presents the full model-by-language heatmap.

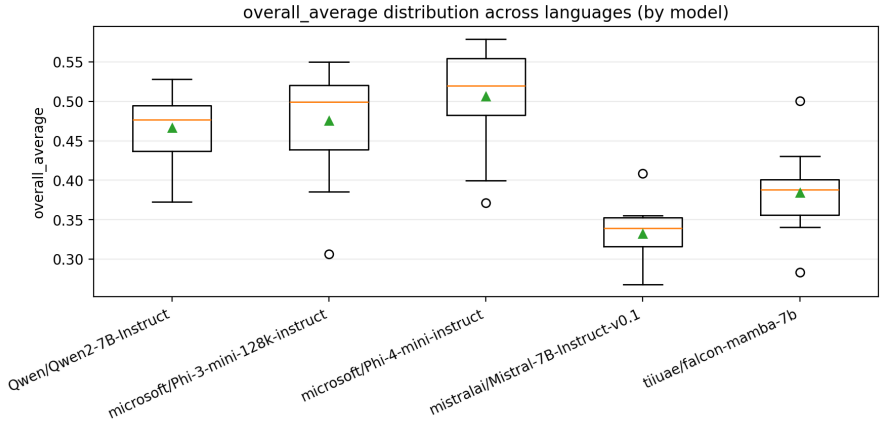

Figure 4. Distribution of overall task scores across languages for each evaluated model. The Phi-4 model achieves the highest median performance with moderate variance, while Mistral shows the lowest scores despite having the lowest energy consumption.

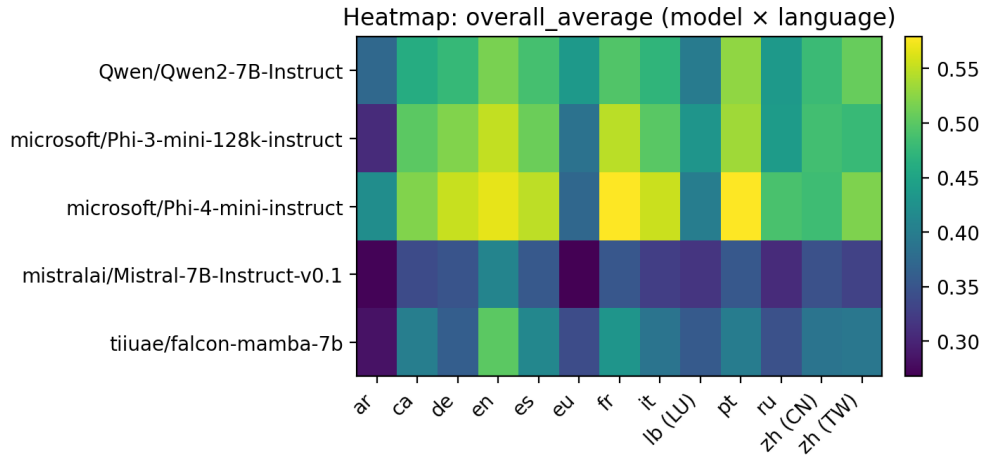

Figure 5. Heatmap of overall task scores across models (rows) and languages (columns). Performance varies both across models and within models across languages, with the largest language-dependent fluctuations observed for Phi-4 and Qwen2.

Statistical Significance of Performance Differences

Friedman tests confirm significant overall differences in all three performance metrics across models ($p \leq 5.74 \times 10^{-6}$). Table 5 reports the pairwise Wilcoxon signed-rank tests with Holm correction for overall task score. Phi-4 significantly outperforms Falcon-Mamba and Mistral on overall score (Holm-corrected $p = 0.0024$, Cohen's $d = 2.44$ and 3.79, respectively), while the difference between Phi-4 and Phi-3 is smaller in magnitude ($d = -0.86$, $p = 0.018$). Notably, Qwen2 does not significantly outperform Phi-3 on overall score ($d = -0.25$, $p = 0.305$) despite consuming 70% more energy, underscoring the weak coupling between energy expenditure and performance.

Table 5. Pairwise Wilcoxon signed-rank tests (Holm-corrected) for overall task score. Positive Cohen's *d* indicates that model A achieves higher performance than model B.

Model A	Model B	W	p_{Holm}	Reject	Cohen's d
Qwen2-7B	Mistral-7B	0.0	0.0024	Yes	4.75
Phi-4-mini	Mistral-7B	0.0	0.0024	Yes	3.79
Phi-3-mini	Mistral-7B	0.0	0.0024	Yes	3.58
Qwen2-7B	Falcon-Mamba-7B	0.0	0.0024	Yes	2.63
Phi-3-mini	Falcon-Mamba-7B	0.0	0.0024	Yes	2.47
Phi-4-mini	Falcon-Mamba-7B	0.0	0.0024	Yes	2.44
Mistral-7B	Falcon-Mamba-7B	0.0	0.0024	Yes	−2.21
Qwen2-7B	Phi-4-mini	10.0	0.0210	Yes	−0.96
Phi-3-mini	Phi-4-mini	8.0	0.0183	Yes	−0.86
Qwen2-7B	Phi-3-mini	30.0	0.3054	No	−0.25

4.3. Category-Level Analysis

Decomposing performance by task category reveals that the aggregate scores reported in Section 4.2 mask substantial heterogeneity across reasoning modalities. Figures 6–10 present category-by-language heatmaps of the average performance for each model, exposing category-specific strengths and weaknesses that differ across architectures.

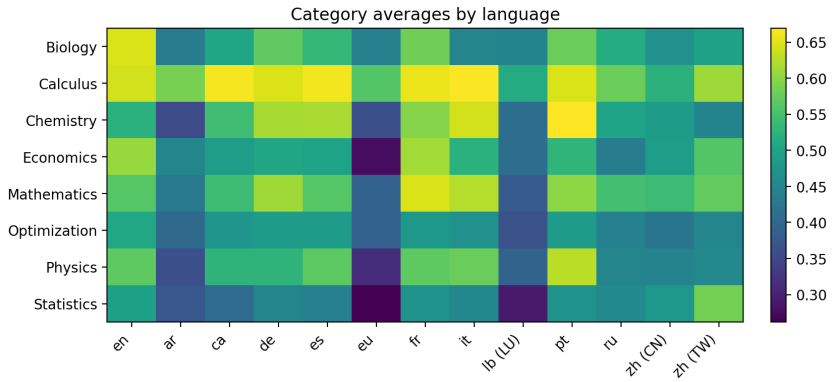

Figure 6. Category-by-language performance heatmap for Phi-4-mini-instruct. The model exhibits strong performance across most categories, with visible language-dependent variation in arithmetic and logical reasoning tasks.

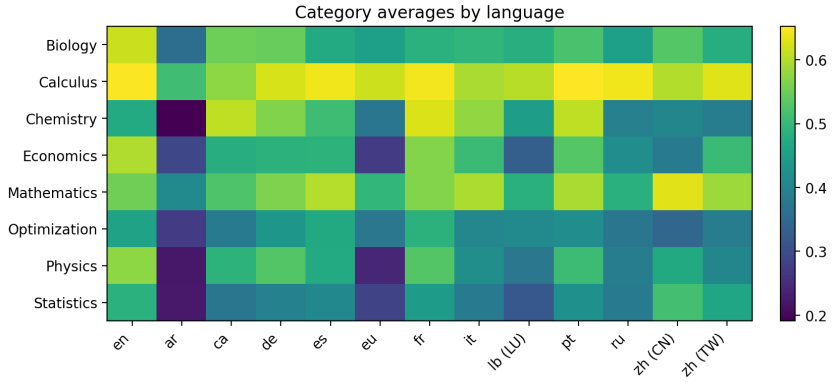

Figure 7. Category-by-language performance heatmap for Phi-3-mini-128k-instruct. The profile is qualitatively similar to Phi-4 but with slightly lower absolute scores, particularly on multi-step reasoning categories.

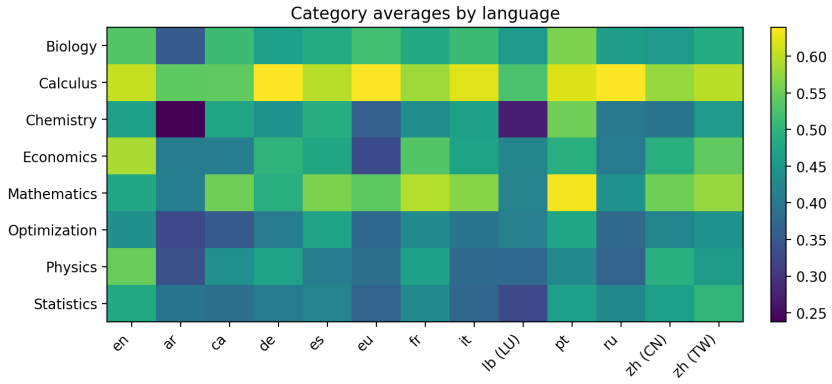

Figure 8. Category-by-language performance heatmap for Qwen2-7B-Instruct. Notable CJK-language strength is visible in certain categories, consistent with Qwen2's larger multilingual vocabulary.

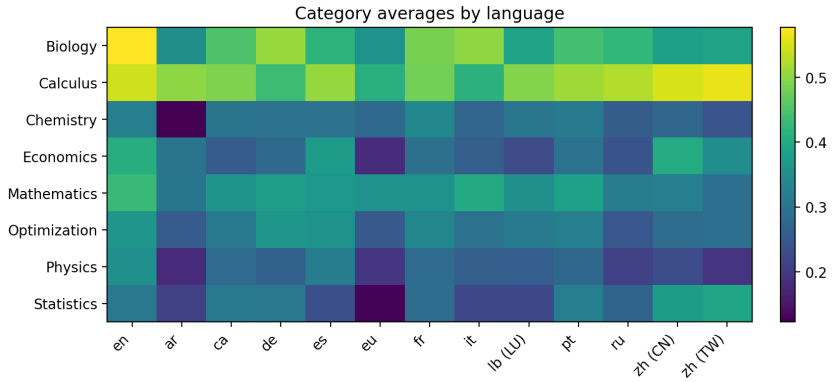

Figure 9. Category-by-language performance heatmap for Mistral-7B-Instruct-v0.1. Performance is uniformly low across categories, with minimal language-dependent variation.

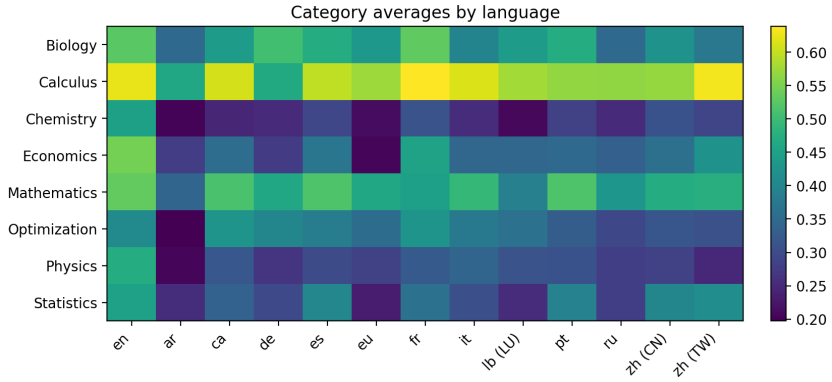

Figure 10. Category-by-language performance heatmap for Falcon-Mamba-7B. Moderate performance with a distinctive pattern: the SSM architecture shows relative strength in pattern-recognition tasks but weaker performance on multi-step logical deduction.

Category Deviations from English Baseline

To isolate language-dependent category effects, Figures 11 and 12 present category-by-language heatmaps of the performance deviation relative to English ($\Delta = S_{\text{category},\ell} - S_{\text{category,en}}$). We select two representative models for this analysis as follows: Phi-4-mini-instruct (Figure 11), as the best-performing model overall, and Qwen2-7B-Instruct (Figure 12), as the model with the largest multilingual vocabulary and hence the most distinctive cross-lingual behavior. These reveal that certain category–language combinations exhibit systematic deviations from the English baseline, with the direction and magnitude depending on the model.

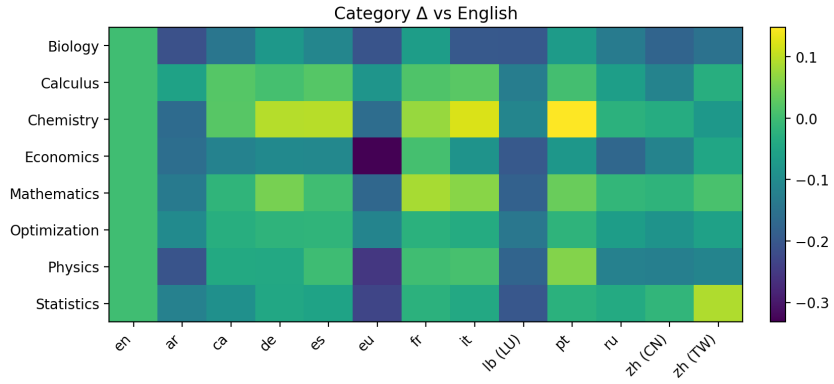

Figure 11. Category-by-language performance deviation relative to English for Phi-4-mini-instruct. Warm colors indicate improvement over English; cool colors indicate degradation. Romance languages (ca, es, pt) show modest improvements in several categories.

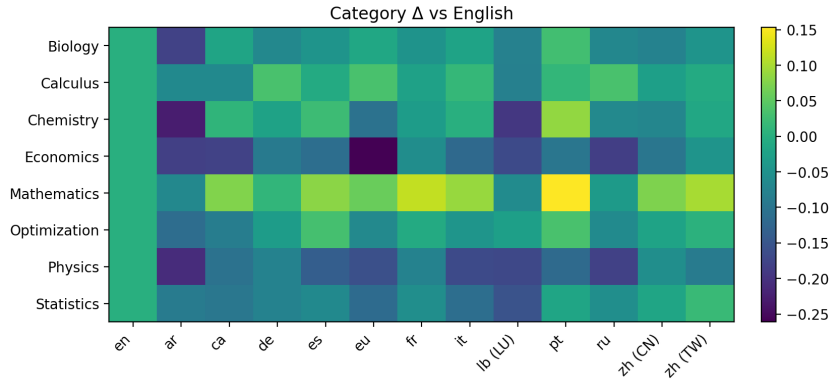

Figure 12. Category-by-language performance deviation relative to English for Qwen2-7B-Instruct. Chinese variants (zh-cn, zh-tw) show positive deviations in several categories, reflecting the model's CJK-optimized training.

These category-level results demonstrate that aggregate scores can obscure important information about where models succeed and fail across reasoning modalities and languages. For deployment scenarios that prioritize specific task categories, the fine-grained profiles presented here provide actionable guidance beyond what overall rankings can offer.

4.4. Difficulty-Level Analysis

The decomposition by difficulty level reveals how energy consumption and performance scale with task complexity. Figures 13–17 present difficulty-by-language heatmaps for all five models.

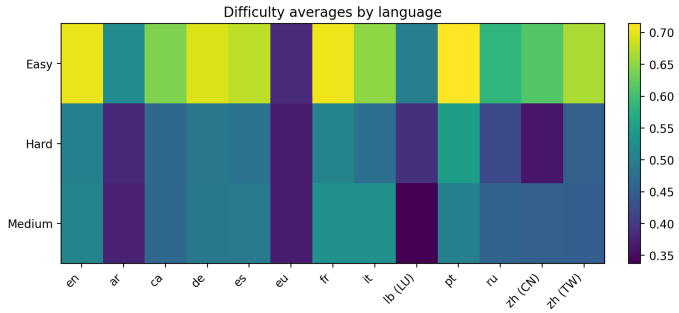

Figure 13. Difficulty-by-language performance heatmap for Phi-4-mini-instruct. Performance degrades gracefully with increasing difficulty, maintaining a relatively consistent language ranking across difficulty levels.

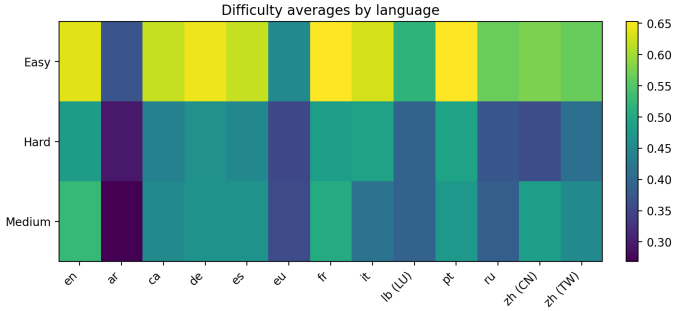

Figure 14. Difficulty-by-language performance heatmap for Phi-3-mini-128k-instruct. A similar degradation pattern to Phi-4, with slightly steeper performance drops at the highest difficulty levels.

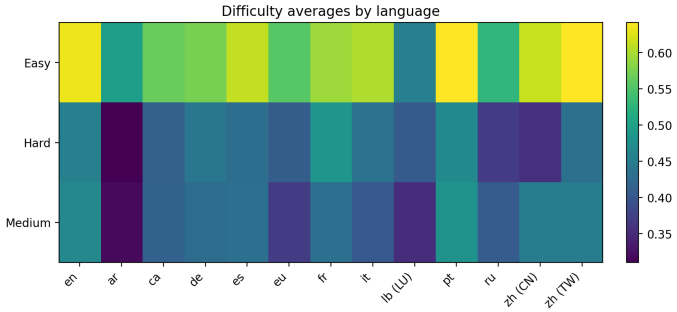

Figure 15. Difficulty-by-language performance heatmap for Qwen2-7B-Instruct. Despite its higher energy consumption, Qwen2 does not maintain a performance advantage over the Phi models at the highest difficulty levels.

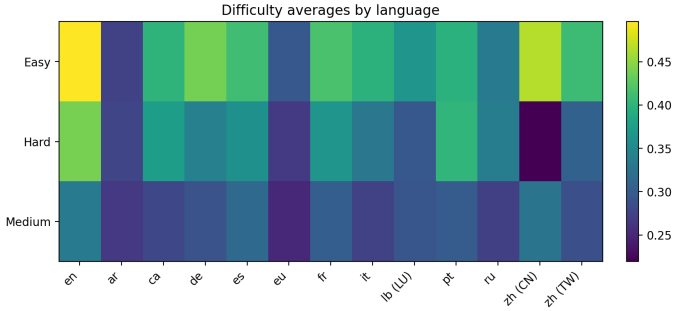

Figure 16. Difficulty-by-language performance heatmap for Mistral-7B-Instruct-v0.1. Performance is strongly difficulty-dependent, with near-zero scores on the hardest problems across all languages.

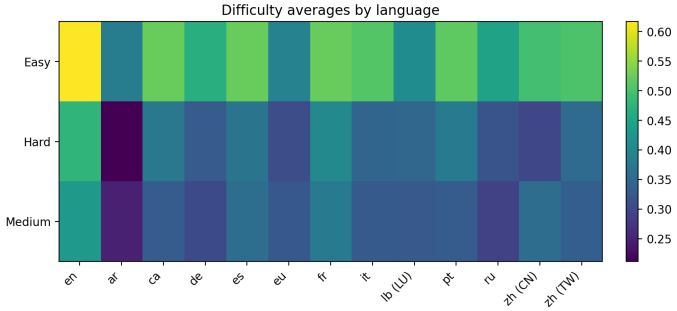

Figure 17. Difficulty-by-language performance heatmap for Falcon-Mamba-7B. The SSM architecture shows a distinctive difficulty profile, with a sharper performance cliff at intermediate difficulty levels compared to Transformer-based models.

Difficulty Deviations from English

We present difficulty-level deviations relative to English for two representative models selected to illustrate contrasting architectural behaviors as follows: Phi-4-mini-

instruct (Figure 18), as the top-performing Transformer with GQA, and Falcon-Mamba-7B (Figure 19), as the sole SSM architecture, which exhibits the most distinctive difficulty-dependent language sensitivity pattern.

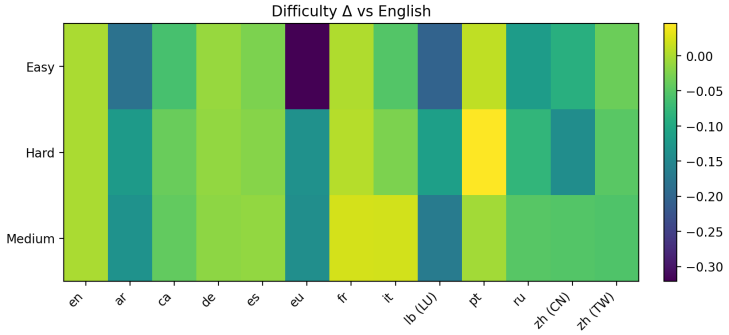

Figure 18. Difficulty-by-language deviation relative to English for Phi-4-mini-instruct. Language sensitivity increases with difficulty level: easy problems show minimal cross-lingual variation, while hard problems exhibit larger and more language-dependent deviations.

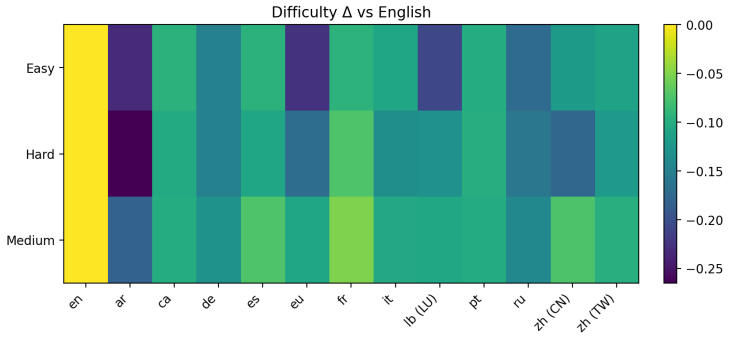

Figure 19. Difficulty-by-language deviation relative to English for Falcon-Mamba-7B. The SSM architecture shows asymmetric language sensitivity: certain languages (eu, lb) exhibit larger negative deviations on harder problems.

A key finding from the difficulty-level analysis is that language sensitivity is *not* uniform across difficulty levels. For all models, cross-lingual performance variance increases with task difficulty, indicating that the hardest problems amplify language-dependent effects. This observation has practical implications as follows: deployment scenarios involving complex reasoning tasks require more careful attention to multilingual performance variation than those involving simpler tasks.

4.5. Language Sensitivity and Baseline Deviations

To quantify the impact of language choice on energy consumption and performance, we compute deviations from the English baseline for each model across all metrics. This analysis reveals that language choice acts as a measurable deployment parameter affecting both cost and quality.

4.5.1. Energy Deviations

Figures 20 and 21 present the energy deviation relative to English ($\Delta E = E_{1000,\ell} - E_{1000,\text{en}}$) for two representative models that bracket the range of cross-lingual energy behavior as follows: Phi-4-mini-instruct (Figure 20), representative of the moderate and symmetric deviations observed across the GQA-based Transformers, and Falcon-Mamba-7B (Figure 21), whose SSM architecture produces a qualitatively different language-energy profile.

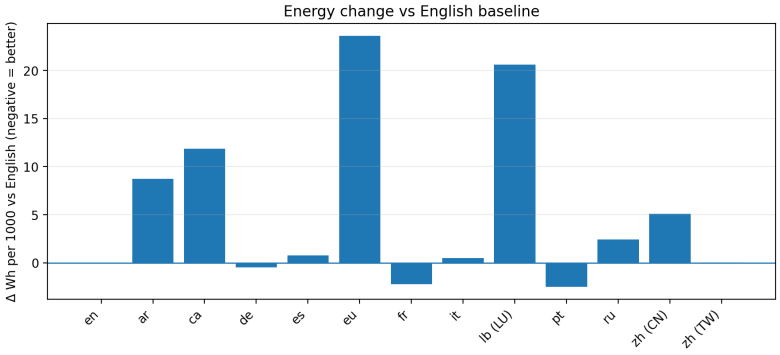

Figure 20. Energy consumption deviation relative to English for Phi-4-mini-instruct. Negative values indicate lower energy consumption than English. Several Romance languages (ca, es, pt) and German consistently require less energy.

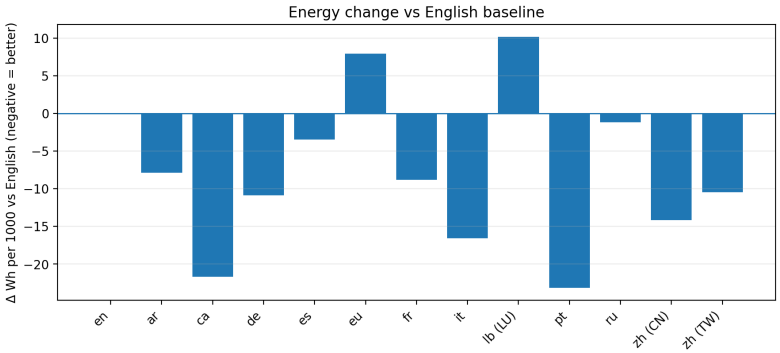

Figure 21. Energy consumption deviation relative to English for Falcon-Mamba-7B. The SSM architecture shows a different language-energy profile, with Arabic and Chinese variants incurring notably higher energy costs.

4.5.2. Performance Deviations

Figures 22 and 23 present the performance deviation relative to English for two models. We show results for Phi-4-mini-instruct (Figure 22), as the overall top performer, and Qwen2-7B-Instruct (Figure 23), whose CJK-optimized vocabulary produces a qualitatively distinct deviation profile.

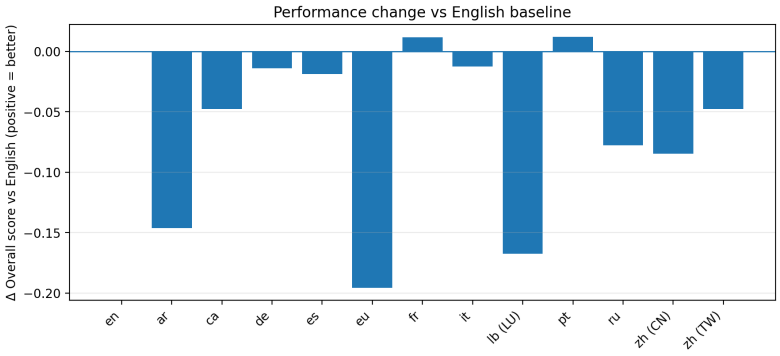

Figure 22. Overall task score deviation relative to English for Phi-4-mini-instruct. Most languages perform within ± 0.05 of English, with Catalan and Spanish showing slight improvements and Basque and Luxembourgish showing the largest degradation.

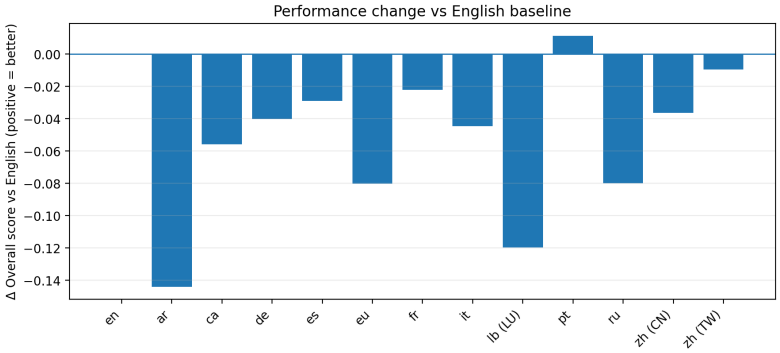

Figure 23. Overall task score deviation relative to English for Qwen2-7B-Instruct. Chinese variants show positive deviations, consistent with the model's training data composition.

4.5.3. The Romance Language Efficiency Cluster

Table 6 reports the descriptive statistics aggregated by language across all five models.

Table 6. Descriptive statistics by language, averaged across all five models. Energy consumption (Wh/1000q) and overall task score are reported as mean ± std. Languages are ordered by ascending mean energy consumption.

Language	Wh/1000q (Mean ± Std)	Overall (Mean ± Std)
pt	301.2 ± 156.2	0.479 ± 0.097
ca	305.1 ± 151.1	0.444 ± 0.074
it	310.3 ± 148.8	0.448 ± 0.091
fr	311.0 ± 155.2	0.481 ± 0.091
en	315.0 ± 140.5	0.509 ± 0.062
es	315.1 ± 145.9	0.462 ± 0.078
de	315.9 ± 143.1	0.452 ± 0.093
ru	321.4 ± 139.3	0.403 ± 0.075
lb	326.8 ± 147.8	0.380 ± 0.044
zh-tw	337.7 ± 115.1	0.445 ± 0.082
ar	338.2 ± 107.1	0.331 ± 0.064
zh-cn	338.7 ± 113.3	0.436 ± 0.065
eu	340.5 ± 125.8	0.360 ± 0.062

Across multiple models, several Romance languages, Catalan (ca), Spanish (es), and Portuguese (pt), frequently exhibit both lower energy consumption and comparable or slightly improved performance relative to English. This pattern is consistent with the tokenization efficiency hypothesis as follows: Romance languages with Latin script and vocabulary overlap with English tend to produce shorter token sequences, reducing the number of autoregressive generation steps and hence the total energy consumed [42,43]. The effect is most pronounced for models with smaller vocabularies (Mistral, Phi-3, and Phi-4) where the tokenizer has limited capacity to represent diverse morphological forms compactly.

4.5.4. High-Cost Languages

Conversely, languages with non-Latin scripts (Arabic, Chinese, and Russian) or agglutinative morphology (Basque) tend to incur higher energy costs, particularly on models with smaller vocabularies. Qwen2-7B-Instruct, with its 151,936-token vocabulary and extensive CJK coverage, partially mitigates this effect for Chinese, exhibiting smaller energy penalties for zh-cn and zh-tw compared to other models.

The per-model energy and performance bar charts across languages, shown in Figures 24 and 25, provide the complete language rankings for two representative models.

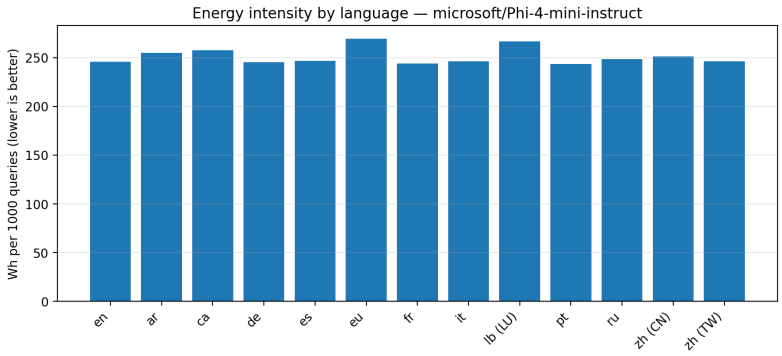

Figure 24. Energy consumption (Wh per 1000 queries) by language for Phi-4-mini-instruct. The compact model shows relatively tight energy clustering across languages, with Arabic and Chinese at the high end.

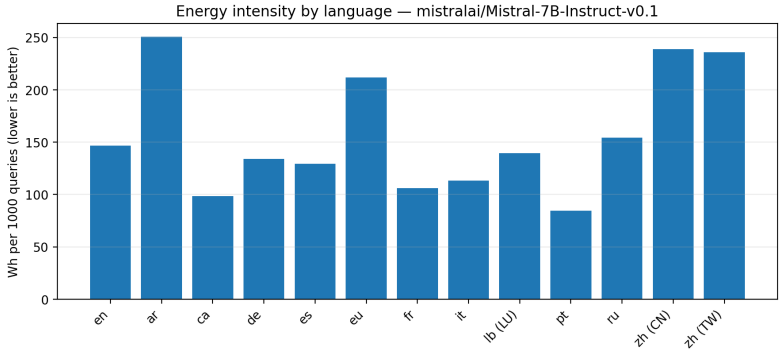

Figure 25. Energy consumption (Wh per 1000 queries) by language for Mistral-7B-Instruct-v0.1. This model shows the largest cross-lingual energy variance among all evaluated models, with up to twofold differences between the cheapest and most expensive languages.

4.6. Energy–Performance Correlation Analysis

To formally characterize the relationship between energy consumption and task performance, we compute Spearman rank correlations across the full set of 65 model–language configurations.

4.6.1. Pooled Correlations

Table 7 reports the Spearman rank correlations between energy consumption and all performance metrics, computed across the full set of 65 model–language configurations. The pooled Spearman correlation between E_{1000} and S_{overall} is weak and non-significant ($r_s = 0.109$, $p = 0.386$), confirming that higher energy consumption does not reliably predict higher reasoning performance across models and languages. Similarly, the correlation between E_{1000} and step accuracy is weak ($r_s = 0.226$, $p = 0.071$), and the correlation with consistency is negligible ($r_s = 0.175$, $p = 0.163$).

Table 7. Spearman rank correlations between energy consumption (Wh per 1000 queries) and performance metrics, pooled across all 65 model–language configurations. None of the correlations reach statistical significance at $\alpha = 0.05$, confirming weak energy–performance coupling.

Performance Metric (y)	r_s	p-Value	n
Overall task score	0.109	0.386	65
Step accuracy	0.226	0.071	65
Consistency	0.175	0.163	65

The key finding is the absence of a meaningful positive correlation between energy and raw performance as follows: spending more energy does not purchase better reasoning. Step accuracy approaches marginal significance ($p = 0.071$) but with a weak effect size ($r_s = 0.226$), indicating that, even if a subtle trend exists, it explains less than 6% of the variance.

The scatter plot of overall task score versus energy consumption for all 65 model–language configurations, with Pareto-optimal points additionally marked, is presented in Section 4.8. The absence of a coherent positive trend across the full scatter confirms weak energy–performance coupling, while the sparse Pareto frontier illustrates which configurations are non-dominated.

4.6.2. Per-Model Correlations

When Spearman correlations are computed within each model (across its 13 language configurations), the results are heterogeneous. For Mistral-7B, which exhibits the largest cross-lingual energy variance, the within-model correlation between E_{1000} and S_{overall} is moderately negative, suggesting that languages for which Mistral consumes more energy (due to longer token sequences) tend to yield lower performance. For the other models, within-model correlations are weak and inconsistent in sign, indicating that within an architecture, language-dependent energy variation is largely decoupled from language-dependent performance variation.

These results demonstrate that the weak energy–performance coupling is not an artifact of aggregation across heterogeneous models as follows: it holds both globally and, for most models, within-model across languages. This finding has important implications for model selection, as it indicates that choosing higher-energy models does not provide a reliable pathway to improved reasoning quality.

4.7. AI Energy Scores and Efficiency Rankings

The AI Energy Score, defined as the ratio of overall task performance to energy consumption per 1000 queries (Equation (3)), provides a composite measure of energy–performance efficiency. Figure 26 shows the distribution of performance efficiency ($\eta = S_{\text{overall}}/E_{1000}$) across languages for each model, and Figure 27 presents the mean efficiency ranking.

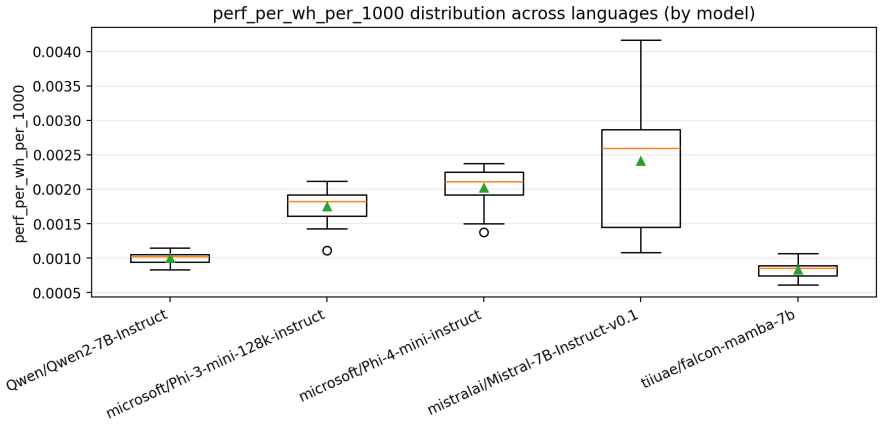

Figure 26. Distribution of performance efficiency (overall score per Wh per 1000 queries) across languages for each model. Higher values indicate more favorable energy–performance trade-offs.

As shown in Table 4, Mistral-7B achieves the highest AI Energy Score (2.11×10^{-3}) despite having the lowest absolute performance, owing to its very low energy consumption. Phi-4-mini-instruct follows closely (2.03×10^{-3}), combining high performance with moderate energy demand. Phi-3-mini-128k-instruct ranks third (1.75×10^{-3}), while Qwen2 (1.01×10^{-3}) and Falcon-Mamba (0.83×10^{-3}) are substantially less efficient.

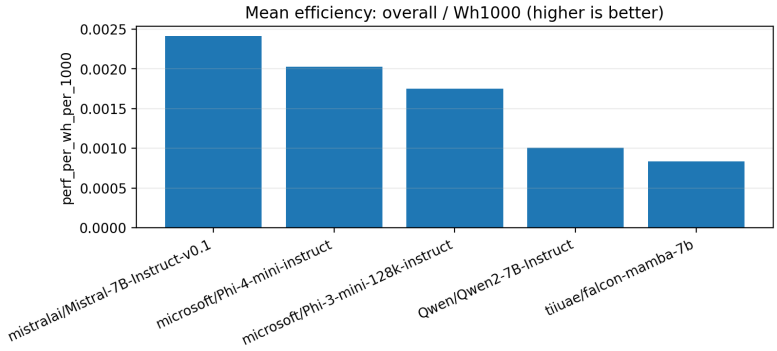

Figure 27. Mean performance efficiency across languages, by model. Error bars indicate standard deviation. The ranking differs from both the pure performance and pure energy rankings, illustrating the value of composite evaluation.

The efficiency heatmap in Figure 28 reveals that the efficiency ranking is not stable across languages, as follows: Mistral's high efficiency is concentrated in languages where its energy consumption is lowest (typically high-resource Latin-script languages), while its advantage diminishes for languages that require longer token sequences. This language-dependent efficiency variation motivates the per-language Pareto analysis in the following section.

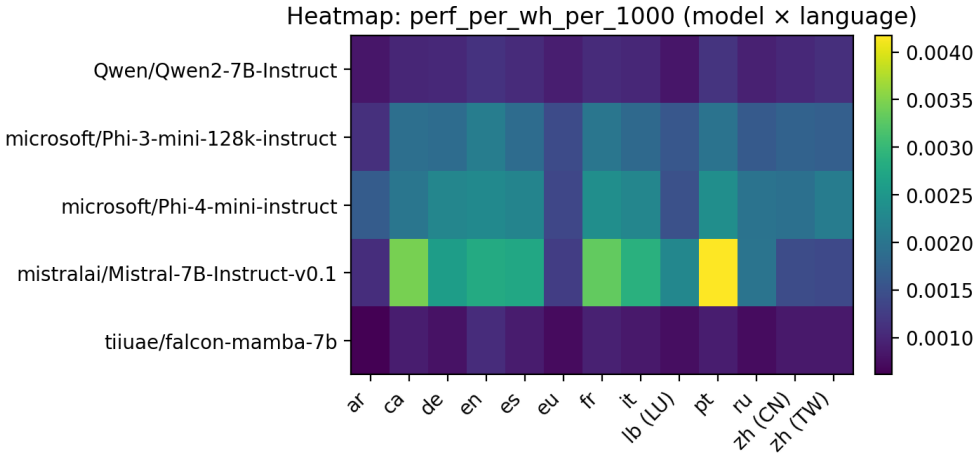

Figure 28. Heatmap of performance efficiency (overall score per Wh per 1000 queries) across models and languages. Efficiency rankings shift across languages, with Mistral's advantage concentrated in high-resource Latin-script languages.

A key observation is that the efficiency ranking does not correspond to either the pure performance ranking (where Phi-4 leads) or the pure energy ranking (where Mistral leads). This discrepancy illustrates the value of composite metrics as follows: a model may rank favorably on individual axes but poorly on the trade-off between them, and vice versa. The AI Energy Score captures this trade-off in a quantity suitable for deployment decisions where both accuracy and sustainability matter.

4.8. Pareto Frontier Analysis

To capture the multi-objective structure of the energy–performance space, we compute the Pareto frontier across all 65 model–language configurations. Figure 29 presents the global Pareto frontier, identifying the non-dominated configurations that jointly minimize energy consumption and maximize overall task performance.

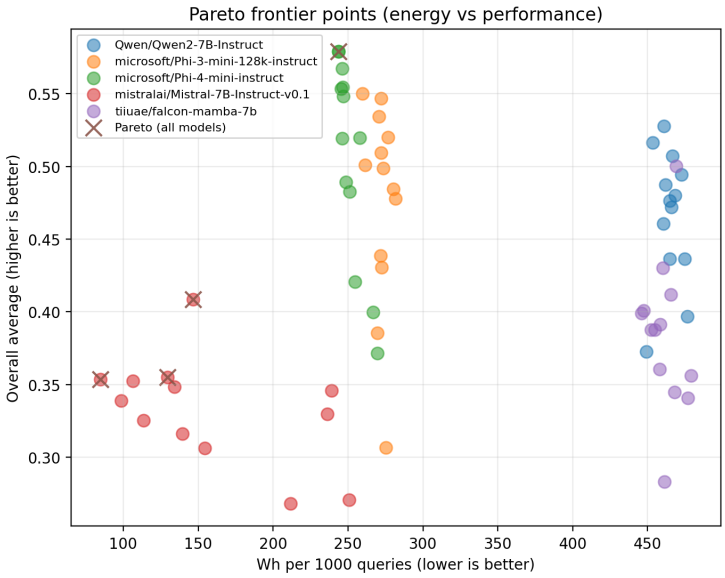

Figure 29. Global Pareto frontier of energy consumption versus overall task performance across all models and languages. Each point represents a model–language configuration; crosses indicate Pareto-optimal (non-dominated) points. The frontier spans from the low-energy/low-performance Mistral regime through the balanced Phi-4 region.

The Pareto frontier reveals several key structural features of the energy–performance space as follows:

4.8.1. Frontier Composition

Table 8 lists the Pareto-optimal configurations, sorted by ascending energy consumption. The non-dominated set consists of only four configurations as follows: three Mistral-7B points occupying the low-energy extreme (Portuguese at 84.8 Wh, Spanish at 129.4 Wh, and English at 146.7 Wh) and a Phi-4-mini-instruct point on Portuguese (243.5 Wh) occupying the high-performance region. The sparsity of the frontier is itself significant as follows: out of sixty-five model–language configurations, only four are non-dominated, indicating that the vast majority of configurations are strictly inferior to at least one alternative. Critically, neither Falcon-Mamba-7B, Qwen2-7B-Instruct, nor Phi-3-mini-128k-instruct contribute any Pareto-optimal point; their energy–performance combinations are uniformly dominated by Mistral or Phi-4 configurations that achieve comparable or better accuracy at lower energy cost.

Table 8. Pareto-optimal (non-dominated) configurations in the energy–performance space, sorted by ascending energy consumption. Only model–language configurations that are not dominated by any other configuration are included.

Model	Language	Wh/1000q	Overall Score
Mistral-7B-Instruct-v0.1	pt	84.8	0.354
Mistral-7B-Instruct-v0.1	es	129.4	0.355
Mistral-7B-Instruct-v0.1	en	146.7	0.408
Phi-4-mini-instruct	pt	243.5	0.579

Pareto optimality is defined with respect to overall task score and energy consumption; alternative dominance definitions incorporating consistency or step accuracy may yield broader frontiers.

4.8.2. The Shape of the Frontier

The frontier spans a wide energy range (84.8–243.5 Wh) but only a modest performance range (0.354–0.579 overall score). The transition from the Mistral regime to the Phi-4 regime incurs a 1.7× energy increase (from 146.7 to 243.5 Wh) for a 1.4× performance improvement (from 0.408 to 0.579), indicating diminishing returns at the high-performance end. The gap between the last Mistral frontier point (en, 146.7 Wh) and the Phi-4 frontier point (pt, 243.5 Wh) is notably large, suggesting that this intermediate zone is entirely dominated and that no evaluated model provides an efficient energy–performance trade-off in this region.

4.8.3. Language-Dependent Frontier Shifts

Portuguese appears on the frontier for both Mistral and Phi-4, occupying the most energy-efficient position for each model. This is consistent with the tokenization efficiency analysis, as follows: Portuguese yields compact token sequences on models with 32K-token vocabularies, reducing generation steps and hence energy. The presence of three distinct languages (pt, es, en) on the Mistral portion of the frontier confirms that language choice materially affects frontier membership, as follows: the same model can shift from dominated to non-dominated depending on the input language.

4.8.4. Implications for Model Selection

The Pareto analysis yields a remarkably sparse frontier, as follows: only two models (Mistral-7B and Phi-4-mini) and three languages (pt, es, en) survive the dominance filter. For applications requiring the highest reasoning quality, Phi-4-mini-instruct on Portuguese offers the best energy–performance configuration (0.579 overall at 243.5 Wh). For scenarios where energy budget is the primary constraint and approximate answers are acceptable, Mistral-7B-Instruct-v0.1 on Portuguese provides an extreme low-energy option (0.354 overall at only 84.8 Wh, a 2.9× energy reduction relative to Phi-4 at the cost of a 39% performance drop). The three remaining models, Phi-3-mini-128k-instruct, Falcon-Mamba-7B, and Qwen2-7B-Instruct, are Pareto-dominated across all language configurations and would require specific use-case justification to warrant their selection, such as Qwen2's superior CJK performance or Phi-3's extended 128K context window.

5. Discussion

The results presented in this study reveal a rich and multi-dimensional structure in the energy–performance landscape of Large Language Models operating under multilingual inference conditions. In this section, we interpret these findings along four axes: architectural implications (Section 5.1), the multilingual efficiency gap (Section 5.2), task complexity and energy scaling (Section 5.3) and practical deployment guidelines (Section 5.4).

5.1. Architectural Implications

The most salient finding from the energy analysis is the emergence of three distinct energy regimes that correspond closely to architectural families rather than to parameter count alone. This stratification challenges the common heuristic that larger models are necessarily more expensive to run and that energy consumption scales primarily with the number of parameters.

5.1.1. SSM vs. Transformer Energy Regimes

Falcon-Mamba-7B, the sole State Space Model in our evaluation, operates in the high-energy regime (461 Wh per 1000 queries) despite its theoretically favorable linear-time complexity with respect to sequence length [22]. This apparent paradox can be attributed to several factors. First, the Mamba architecture's selective scan mechanism,

while asymptotically efficient, involves the materialization of intermediate states that may not map optimally onto current GPU hardware, which is architecturally optimized for matrix multiplications characteristic of attention-based Transformers [47]. Second, at the moderate sequence lengths encountered in our evaluation (typical of reasoning tasks rather than long-document processing), the constant factors in the SSM computation dominate over the asymptotic advantage, and the linear-time benefit does not materialize into measurable energy savings. Third, the Falcon-Mamba implementation may not yet benefit from the same degree of inference-time kernel optimization that Transformer-based models have accumulated over several years of engineering effort.

These observations suggest that architectural novelty does not automatically translate into energy efficiency at inference time and that the practical energy profile of a model depends critically on the interaction between algorithm, hardware, and software optimization maturity. For practitioners, this implies that theoretical complexity analysis is an unreliable proxy for real-world energy consumption and that empirical measurement under representative workloads remains essential.

5.1.2. GQA Efficiency Gains

Among the Transformer-based models, Grouped-Query Attention provides a measurable efficiency advantage. Phi-4-mini-instruct (GQA, 3.84B parameters, 251 Wh) consumes 7% less energy than Phi-3-mini-128k-instruct (full multi-head attention, 3.82B parameters, 272 Wh) while achieving higher performance on all three metrics. This difference, modest but consistent across all thirteen languages, is attributable to the reduced KV cache memory footprint and lower memory bandwidth requirements of GQA during autoregressive generation [45]. The effect is amplified on the H200 GPU, whose HBM3e memory subsystem rewards bandwidth-efficient access patterns.

The comparison between Mistral-7B (GQA, 7.24B) and Qwen2-7B (GQA, 7.07B) further illustrates that GQA alone does not determine energy consumption. Despite sharing the GQA mechanism and similar parameter counts, these models differ by a factor of three in energy demand. This gap is driven by differences in generation behavior as follows: Mistral produces substantially shorter outputs, completing inference in fewer autoregressive steps and hence consuming less total energy. This observation underscores that the number of generated tokens, which depends on the model's learned generation dynamics, not only its architecture, is a dominant factor in inference-time energy consumption.

5.1.3. The Compact Model Advantage

Perhaps the most practically significant finding is that the compact Phi-family models (3.8B parameters) achieve a more favorable energy–performance balance than any of the 7B-parameter alternatives. Phi-4-mini-instruct delivers the highest overall task score at less than 55% of the energy cost of Qwen2 and Falcon-Mamba. This result extends the observation of Schwartz et al. [12] that efficiency-oriented design can yield disproportionate returns, and it supports the growing evidence that raw parameter count is a poor predictor of reasoning capability in the presence of targeted training and architectural optimization [23]. For deployment contexts where inference cost is a binding constraint, compact models represent a Pareto-dominant choice under all but the most specialized linguistic or task requirements.

5.2. The Multilingual Efficiency Gap

A central contribution of this study is the demonstration that language choice has a measurable and systematic impact on both energy consumption and performance efficiency. This multilingual efficiency gap has not been previously quantified in the literature and carries direct implications for the deployment cost of LLM-based systems in linguistically diverse environments.

5.2.1. Tokenization as a Hidden Energy Driver

The primary mechanism underlying the multilingual efficiency gap is tokenization. Models with smaller vocabularies (32K tokens for Mistral, Phi-3, and Phi-4) encode non-English text with higher fertility, more tokens per semantic unit, leading to longer input and output sequences [42]. Since autoregressive generation cost scales linearly with the number of generated tokens (each requiring a full forward pass through the model), higher fertility directly translates into higher energy consumption. This effect is most pronounced for Mistral-7B, which exhibits up to twofold energy variation across languages (Figure 25), with Arabic and Chinese at the expensive end and Romance languages at the efficient end.

Qwen2-7B-Instruct, with its 151,936-token vocabulary designed for extensive CJK and multilingual coverage [43], partially mitigates this effect as follows: its cross-lingual energy variance is among the lowest of all evaluated models (7.9 Wh standard deviation vs. 57.4 Wh for Mistral). However, this mitigation comes at the cost of a larger embedding matrix and higher baseline energy consumption, resulting in a net efficiency that is still inferior to the compact Phi models for most languages.

5.2.2. Language Choice as a Deployment Parameter

The observation that Romance languages (Catalan, Spanish, and Portuguese), on average, achieve lower energy consumption than English across multiple models, while maintaining comparable or slightly improved reasoning performance, suggests that language choice itself may function as a tunable deployment parameter. In scenarios where the input language is flexible (e.g., machine translation pipelines or multilingual question answering with internal language routing), selecting a more token-efficient language for internal processing could yield measurable energy savings without sacrificing output quality.

This finding also has equity implications. Languages with non-Latin scripts or complex morphology (Arabic, Basque, and Chinese) incur higher per-query energy costs, effectively imposing a "computational tax" on speakers of these languages [43]. As LLM-based systems become embedded in public services, education, and communication infrastructure, this linguistic energy asymmetry becomes a matter of equitable access and merits explicit consideration in model selection and vocabulary design.

5.2.3. Instability of Model Rankings Across Languages

The language-dependent shifts in model efficiency rankings observed in Section 4.7 have a practical consequence, outlined as follows: benchmarks conducted in English alone can produce misleading model rankings for multilingual deployments. A model that appears energy-efficient in English may lose its advantage, or become dominated, when evaluated on typologically different languages. This instability reinforces the need for multilingual evaluation as a standard component of energy-aware benchmarking, particularly for systems intended to serve diverse linguistic populations.

5.3. Task Complexity and Energy Scaling

The difficulty-level decomposition presented in Section 4.4 reveals a previously uncharacterized interaction between task complexity, language sensitivity, and model architecture that has implications for energy-aware inference design.

5.3.1. Non-Linear Energy Scaling with Difficulty

While our experimental design measures energy at the level of complete evaluation runs rather than individual problems, the difficulty-level performance analysis provides indirect evidence of non-linear energy scaling. Harder problems elicit longer chain-of-

thought sequences from the models, increasing the number of generated tokens and hence the energy consumed per problem. This effect compounds with the language-dependent fertility discussed above as follows: for a given hard problem, the energy cost difference between an efficient language (e.g., Spanish) and an expensive language (e.g., Arabic) is larger than for an easy problem, because the multiplicative interaction between sequence length and fertility amplifies at higher difficulty levels.

5.3.2. Amplification of Cross-Lingual Variance with Difficulty

A key finding is that the cross-lingual variance in performance increases with task difficulty across all models (Figures 18 and 19). Easy problems are solved at similar rates across languages, while hard problems expose substantial language-dependent performance gaps. This amplification effect suggests that the linguistic robustness of a model's reasoning capability degrades under cognitive load, consistent with findings on prompt sensitivity in challenging reasoning tasks [27,57].

5.3.3. Implications for Adaptive Routing

These findings motivate adaptive inference strategies that route queries based on both estimated difficulty and language characteristics. In a production system handling diverse workloads, an energy-aware router could assign easy queries to a low-energy model (e.g., Mistral-7B), reserve hard queries for a more capable model (e.g., Phi-4), and adjust routing thresholds based on the input language's expected energy overhead. Such strategies could substantially reduce average energy consumption without proportional performance degradation, particularly in mixed-workload deployments where easy queries dominate the distribution. The difficulty-level and language-level efficiency profiles reported in this study provide the empirical basis for calibrating such routing policies.

5.4. Practical Deployment Guidelines

The multi-dimensional analysis presented in this study supports a set of actionable guidelines for practitioners selecting LLMs under energy, performance, and language constraints. We summarize these as a decision framework organized by deployment priority.

5.4.1. Operational Cost and Carbon Footprint at Deployment Scale

To ground the observed energy differences in practical terms, we translate the empirical E_{1000} measurements into illustrative estimates of electricity cost and carbon footprint at two representative serving scales as follows: 1 million and 10 million queries per day. These estimates use the standard data center electricity pricing of \$0.10 kWh^{-1} [5] and the average EU grid carbon intensity of 0.233 $kg\,CO_2\,kWh^{-1}$ (2023 European Environment Agency figures); they should be treated as order-of-magnitude benchmarks subject to infrastructure-specific variation.

At 1 million queries per day, Mistral-7B-Instruct-v0.1 (157.3 Wh per 1000 queries) consumes approximately 157 kWh per day, corresponding to an electricity cost of roughly \$5700 per year and a carbon footprint of approximately 13 tCO_2 per year on an average EU grid. Phi-4-mini-instruct (251.3 Wh/1000q) incurs approximately \$9200 per year (161% of Mistral's cost) for the same query volume. Qwen2-7B-Instruct and Falcon-Mamba-7B (both ≈462 Wh/1000q) reach approximately \$16,900 per year, a difference of \$11,200 per year relative to Mistral and \$7700 per year relative to Phi-4, purely in electricity cost, at this modest serving scale. At 10 million queries per day, these gaps scale proportionally, as follows: the energy cost differential between the low-energy (Mistral) and high-energy (Qwen2, Falcon-Mamba) regimes reaches approximately \$112,000 per year, and the associated carbon difference is approximately 261 tCO_2 per year on an average EU grid, equivalent to roughly 57 transatlantic round-trip flights [14].

When language choice is additionally optimized, for example, by routing Portuguese inputs to Mistral-7B (84.8 Wh/1000q, the most energy-efficient Pareto-optimal configuration) rather than to Qwen2-7B (typically ≈460 Wh/1000q for the same inputs), the energy saving at 1 million queries per day reaches approximately 375 kWh per day, or roughly $13,700 per year in electricity cost alone. These figures illustrate that the language-aware model selection guidelines derived from our Pareto analysis are not merely of academic interest but carry meaningful operational and environmental consequences at real-world deployment scales.

5.4.2. Priority: Maximum Reasoning Quality

When task accuracy is the primary objective and energy budget is secondary, Phi-4-mini-instruct is the recommended choice. It achieves the highest overall task score and step accuracy among all evaluated models while consuming approximately half the energy of comparably performing 7B-parameter alternatives. Its GQA architecture and compact parameter count yield an AI Energy Score that is competitive with the most energy-efficient model in the evaluation.

5.4.3. Priority: Minimum Energy Consumption

When the energy budget is the binding constraint and approximate answers are acceptable, for example, in high-throughput screening, draft generation, or preliminary triage, Mistral-7B-Instruct-v0.1 provides the lowest energy cost per query. However, its suitability is language-dependent as follows: the model's energy advantage is most pronounced on high-resource Latin-script languages and diminishes substantially for Arabic, Chinese, and agglutinative languages. Practitioners serving multilingual populations should account for this variance when estimating deployment costs.

5.4.4. Priority: Balanced Efficiency

For scenarios requiring a balance between reasoning quality and energy consumption, Phi-4-mini-instruct on Portuguese represents the Pareto-optimal configuration at the high-performance end of the frontier (0.579 overall at 243.5 Wh). More broadly, Phi-4 on Romance languages (Catalan, Spanish, Portuguese, French, and Italian) on average achieves favorable energy–performance combinations near the frontier. Phi-3-mini-128k-instruct provides a viable alternative with slightly lower performance but comparable energy consumption, and it may be preferred in contexts where the longer context window (128K tokens) is advantageous, despite being Pareto-dominated in the strict two-objective sense.

5.4.5. Priority: CJK Language Support

For deployments primarily serving Chinese, Japanese, or Korean users, Qwen2-7B-Instruct offers the best language-specific performance on CJK inputs and the lowest cross-lingual energy variance for these languages, owing to its large multilingual vocabulary. However, this advantage is specific to the CJK language family; for other languages, Qwen2 is Pareto-dominated by the Phi models.

5.4.6. When to Avoid High-Energy Models

The results provide clear evidence that Falcon-Mamba-7B and, in most configurations, Qwen2-7B-Instruct are Pareto-dominated in the energy–performance space. Their substantially higher energy consumption does not translate into proportional performance gains relative to the compact Phi models. Unless specific architectural properties (e.g., Mamba's linear-time scaling for very long sequences) or vocabulary characteristics (e.g., Qwen2's CJK coverage) are required, these models are not recommended for energy-conscious deployments.

5.4.7. Multilingual Deployment Cost Estimation

Practitioners should not assume that English-based energy benchmarks generalize to multilingual deployments. We recommend evaluating energy consumption on a representative sample of target languages, or applying the language-dependent correction factors derived from our analysis, to obtain realistic cost estimates. As a rule of thumb, languages with non-Latin scripts or agglutinative morphology may incur 20–100% higher energy costs than English on models with 32K-token vocabularies, depending on the model and task characteristics.

5.4.8. Limitations and Dataset Scope

A limitation that warrants explicit acknowledgment concerns the size of the evaluation dataset. The 79-problem set used in this study is modest relative to large-scale accuracy benchmarks such as MMLU [7] (14,000+ items) or BIG-bench [8], and this constraint has three implications that practitioners should bear in mind when interpreting our results.

Statistical stability of performance estimates. With 79 problems per language, the standard error of the overall task score S_{overall} is of order $\sqrt{p(1-p)/79} \approx 0.056$ at $p = 0.5$. The model-level standard deviations reported in Table 4 (ranging from ± 0.037 for Mistral to ± 0.070 for Phi-4) are therefore comparable in magnitude to this sampling uncertainty, meaning that performance differences of less than approximately 0.05 should be interpreted with caution. For this reason, we deliberately ground all performance comparisons in the Wilcoxon signed-rank framework with Holm correction, treating languages as paired blocks, rather than in point estimates of task score. The resulting p-values and Cohen's d effect sizes reflect the *cross-lingual consistency* of model differences; the uniformly large effect sizes observed for energy-regime separations ($|d| > 2.0$, Table 3) and for the dominant performance contrasts ($|d| > 2.4$, Table 5) suggest that these findings would survive a larger prompt set. Differences near the boundary of statistical significance, notably the Qwen2 vs. Phi-3 overall score comparison ($d = -0.25$, $p = 0.305$), should be treated as inconclusive pending replication on a larger benchmark.

Sensitivity to prompt-specific effects. Because the same 79 problems are used across all models and languages, any systematic bias in the problem selection (e.g., an over-representation of a particular reasoning category or difficulty tier) would propagate uniformly across all configurations. This design choice controls for prompt variability when *comparing* models and languages, the primary goal of the study, but it means that absolute performance values should not be taken as representative of a model's capability on an arbitrary reasoning corpus. The category and difficulty decompositions in Sections 4.3 and 4.4 partially mitigate this concern by revealing whether conclusions hold across the available task types, but they cannot substitute for validation on independently sampled problem sets.

Generalizability to broader benchmarks. The problem set was selected from a prior study on semantic invariance [48] because it provides fully annotated intermediate-step decompositions and machine-verifiable reference solutions across thirteen languages, a requirement for the step accuracy and consistency metrics that is difficult to satisfy at scale. Constructing an equivalent annotation at the scale of MMLU or BIG-bench would require substantial manual effort beyond the scope of this work. We expect that the *energy consumption* results (model-level stratification, language-dependent tokenization effects, and the Pareto-optimal frontier) would generalize well to larger benchmarks, because these are driven by architectural and tokenization properties that are independent of the specific problems evaluated. The *performance* results, particularly fine-grained category rankings and language-specific deviations, are more sensitive to benchmark composition and should be interpreted as indicative rather than definitive. Extending this evaluation to larger,

natively multilingual benchmarks, ideally ones with step-level annotations, is identified as a priority for future work (Section 6).

Aggregated energy measurement and prompt-level granularity. All energy measurements in this study are collected at the level of complete evaluation runs, integrating GPU power over the entire batch of 79 problems per model–language configuration rather than over individual prompts. This design choice is standard in the energy measurement literature [36] and eliminates noise from GPU warm-up transients and OS-level power scheduling that would affect short single-prompt traces. However, it prevents direct decomposition of energy consumption by prompt difficulty, output length, or reasoning complexity. The claims in Section 5.3 regarding non-linear energy scaling with difficulty and chain-of-thought length effects are therefore inferred indirectly from performance patterns rather than measured directly from per-prompt power traces. Specifically, we observe that harder problems elicit longer model outputs and yield lower performance, and we infer from these two observations that energy per problem scales upward with difficulty; however, we cannot quantify this scaling precisely from run-level measurements alone. Fine-grained per-problem energy profiling, leveraging hardware instrumentation beyond NVML (e.g., RAPL counters or hardware performance monitoring units), would be required to substantiate these claims empirically, and it is identified as a priority for future work in Section 6. Practitioners who require prompt-level energy estimates for production cost modeling should therefore treat our run-level figures as averages over the empirical difficulty distribution of the evaluation set, rather than as predictions for individual queries.

Output length as a confounding variable. The paper correctly identifies generation length as a major determinant of inference energy consumption, and it qualitatively attributes Mistral-7B's low energy footprint partly to its tendency to produce shorter outputs (Section 5.1). However, the experimental design does not explicitly control for output length, as follows: we do not report average generated token counts per model and language, nor do we normalize energy by the number of generated tokens. This is a genuine limitation of the current study. Because autoregressive generation cost scales approximately linearly with the number of decoded tokens, a model that produces systematically shorter responses will appear more energy-efficient than an architecturally identical but more verbose model, irrespective of the underlying computational efficiency of each forward pass. We therefore caution that the energy advantage of Mistral-7B partially reflects its generation behavior (short, often incomplete responses at the harder difficulty levels) in addition to its architectural design. A normalized metric such as energy per generated token ($E_{\text{token}} = E_{\text{total}}/N_{\text{tokens}}$) would disentangle these two effects and would be a valuable complement to the Wh/1000-query metric used here. We were unable to retrospectively extract per-run token counts from the NVML power traces in the present study; reporting E_{token} alongside E_{1000}, together with mean token counts per model and language, is identified as a concrete recommendation for future work building on this benchmark. Cross-model comparisons of E_{1000} should therefore be interpreted as reflecting the *operational* energy cost of deploying each model on this task distribution, including its characteristic output length, rather than the *architectural* energy cost per token in isolation.

Decoding configuration and output length ceiling. All evaluations used low-temperature sampling (temperature $= 0.2$, three independent runs per problem), which approximates near-deterministic generation while retaining sufficient stochasticity to measure output stability empirically via the consistency metric C. This configuration is representative of production deployments for reasoning tasks but does not cover the full range of serving strategies encountered in practice, such as higher-temperature creative generation, top-p nucleus sampling, or beam search. A fixed max_tokens $= 300$ ceiling is applied uniformly across all models and languages; problems that would naturally elicit longer chain-of-

thought sequences are therefore truncated at the same budget regardless of model verbosity. This cap partially controls for output-length confounding by bounding the maximum autoregressive steps per problem, but it also means that the reported E_{1000} values do not reflect the cost of unconstrained generation and may underestimate energy for models whose natural response length exceeds 300 tokens.

Model scale and generalizability of conclusions. All five models evaluated in this study fall within the 3.8–7.3 billion parameter range, a deliberate choice motivated by the relevance of this tier to cost-sensitive and edge-adjacent inference deployments [21]. The conclusions presented here, including the three-tier energy stratification, the compact-model Pareto advantage, and the language-dependent efficiency rankings, should therefore be understood as applying specifically to *mid-sized open-weight LLMs* in this parameter band rather than to LLMs in general. At larger scales (13B, 34B, 70B and beyond), several dynamics may alter the energy–performance landscape substantially: multi-GPU tensor parallelism introduces inter-device communication overhead not present in our single-GPU setup; quantization (INT8, INT4) becomes practically essential and can decouple parameter count from memory bandwidth demand; and reasoning-specialized training (e.g., DeepSeek-R1 [58], Hermes-4-70B [59]) may yield qualitatively different energy–performance trade-offs than the instruction-tuned models studied here. Whether the compact-model advantage observed in our evaluation persists at higher capability levels, i.e., whether a well-optimized 7B model continues to dominate a 70B alternative on Pareto-efficiency grounds, is an open empirical question that we identify as a priority for future work in Section 6.

Translation-based dataset and potential translation artifacts. The multilingual evaluation set was constructed by translating the original English problems into twelve additional languages using a three-stage pipeline: (1) automated draft translation via the `deep-translator` (https://github.com/nidhaloff/deep-translator, accessed on 1 February 2026) library with the Google Translate backend, applied field-by-field to problem text, solution narrative, and solution steps while leaving category and difficulty labels untouched; (2) an LLM-based post-editing pass for terminological consistency and numerical preservation; and (3) a final human verification step (Section 3.2). Despite this multi-stage quality control, automated translation cannot fully eliminate translation-induced artifacts.

Hardware specificity of energy measurements. All experiments were conducted on a single NVIDIA H200 GPU in the Boston University cluster. This choice was deliberate; the H200 is one of the reference hardware platforms recommended by the AI Energy Score initiative [17] for standardized energy efficiency benchmarking, which ensures that the E_{1000} values reported here are directly comparable to entries on that leaderboard and to future studies that follow the same convention. Using a community-endorsed reference platform is therefore a methodological strength with respect to reproducibility and cross-study comparability. However, absolute energy values may differ on other hardware platforms due to architecture-specific factors.

Context window heterogeneity and positional encoding effects. The five evaluated models differ substantially in their nominal context window lengths, ranging from 8K tokens for Mistral-7B-Instruct-v0.1 and Falcon-Mamba-7B to 32K for Qwen2-7B-Instruct and 128K for the Phi-family models (Table 1). In principle, this heterogeneity could confound cross-model comparisons if any model were to hit its context limit during evaluation, triggering truncation or degraded positional encoding behavior at long sequence ranges. In practice, this concern does not apply to the present study: the evaluation prompts consist of a fixed instruction prefix followed by a self-contained reasoning problem of at most a few hundred words, and the generation budget is capped at max_tokens $= 300$. The resulting total sequence length (prompt plus generation) remained well below 1000 tokens across all

65 model–language configurations, comfortably within even the smallest context window in the evaluation set. No evidence of context overflow or anomalous attention degradation was observed in the energy traces or wall-clock timing data. However, this conclusion is specific to the short-prompt, single-turn evaluation design used here. Future studies that extend this benchmark to longer reasoning chains, multi-turn dialogue, or retrieval-augmented contexts, where total sequence lengths may approach or exceed the 8 K limit of the smallest-context models, would need to explicitly control for context window effects, either by capping all models at the minimum context size or by excluding models that cannot accommodate the required sequence length.

6. Conclusions and Future Work

This paper presented a comprehensive multilingual evaluation of the energy–performance characteristics of five instruction-tuned Large Language Models spanning three architectural families, standard Transformer, Transformer with Grouped-Query Attention, and State Space Model, across thirteen typologically diverse languages. By jointly analyzing GPU-level energy consumption, multiple reasoning performance metrics, task category and difficulty decompositions, and statistical measures of significance and effect size under controlled inference conditions on NVIDIA H200 hardware, we have provided the most detailed empirical characterization to date of how energy efficiency, reasoning capability, and linguistic context interact in the LLM inference setting.

Our findings can be distilled into four principal conclusions.

First, energy consumption varies by up to threefold across models operating on identical workloads, and these differences are statistically significant across all energy-related metrics ($\chi^2 = 49.42$, $p = 4.78 \times 10^{-10}$, Friedman test). The observed stratification into low-, mid-, and high-energy regimes is driven primarily by architectural design and generation dynamics rather than by parameter count, with compact 3.8B-parameter models consuming less than half the energy of 7B-parameter alternatives while delivering equal or superior reasoning performance.

Second, energy expenditure and reasoning performance are only weakly coupled. Spearman rank correlations across the full set of 65 model–language configurations confirm that higher energy consumption does not reliably predict higher task accuracy ($r_s = 0.109$, $p = 0.386$). This decoupling holds both globally and within individual models across languages, indicating that the pursuit of higher accuracy through more energy-intensive models is not a reliable strategy under the evaluated conditions.

Third, language choice has a measurable and systematic impact on both energy consumption and model efficiency. Romance languages on average achieve lower energy costs than English across multiple models, while languages with non-Latin scripts or complex morphology incur substantially higher costs, driven primarily by tokenization fertility. Cross-lingual performance variance amplifies with task difficulty, and model efficiency rankings shift across languages, rendering English-only benchmarks unreliable guides for multilingual deployment decisions. These findings formalize the existence of a multilingual efficiency gap that has practical consequences for equitable and cost-effective deployment of LLM-based systems.

Fourth, Pareto frontier analysis reveals that out of 65 model–language configurations, only four are non-dominated, three Mistral-7B configurations (on Portuguese, Spanish, and English) and one Phi-4-mini-instruct configuration (on Portuguese). Phi-3-mini-128k-instruct, Falcon-Mamba-7B, and Qwen2-7B-Instruct are Pareto-dominated across all language configurations, with their additional energy expenditure failing to yield proportional performance gains except in language-specific niches.

These conclusions carry direct implications for the evaluation, selection, and governance of Large Language Models. They support the integration of energy efficiency as a first-class metric in LLM benchmarks, alongside accuracy and robustness, and provide empirical grounding for the adoption of AI Energy Scores as a complementary criterion in model selection. The practical deployment guidelines derived from our analysis, including recommendations stratified by accuracy priority, energy budget, and target language mix, offer immediately actionable decision frameworks for practitioners operating under sustainability constraints.

Future Work

Several directions emerge from this study. First, extending the evaluation to additional hardware platforms (A100, L40S, and consumer GPUs) and inference configurations (quantized models, speculative decoding, and batched inference) would assess the robustness of the observed rankings and energy regimes across the heterogeneous infrastructure encountered in real deployments. Second, expanding the model set to include recent reasoning-specialized architectures (e.g., DeepSeek-R1 [58]) and larger-scale models (e.g., 70B-parameter variants [59]) would test whether the compact model advantage persists at higher capability levels and whether reasoning-optimized training further improves energy efficiency. Third, developing natively multilingual evaluation sets, rather than translated benchmarks, would eliminate translation artifacts and enable more controlled analysis of language-specific reasoning patterns. Fourth, fine-grained per-problem energy profiling, leveraging hardware instrumentation beyond NVML, would enable decomposition of energy consumption by computational phase and problem characteristic, supporting targeted optimization at the operator and kernel level. Fifth, the design and empirical evaluation of adaptive inference routers that exploit the difficulty-level and language-dependent efficiency profiles reported here represents a promising path toward systems that dynamically balance energy consumption and reasoning quality at serving time. Finally, longitudinal tracking of energy efficiency across model generations would enable the field to assess whether architectural and training advances are delivering genuine efficiency improvements or merely shifting the cost frontier, contributing to the broader goal of sustainable AI development.

Author Contributions: Conceptualization, J.d.C. and M.L.; data curation, I.d.Z.; formal analysis, J.d.C., I.d.Z., and M.L.; funding acquisition, J.d.C. and I.d.Z.; investigation, I.d.Z., M.L., and J.d.C.; methodology, J.d.C., I.d.Z., and M.L.; software, J.d.C., I.d.Z., and M.L.; supervision, J.d.C. and I.d.Z.; validation, J.d.C., I.d.Z., and C.T.C.; visualization, J.d.C. and I.d.Z.; writing—original draft, J.d.C. and I.d.Z.; writing—review and editing, J.d.C., I.d.Z., and C.T.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the LUXEMBOURG Institute of Science and Technology through the projects "ADIALab-MAST" and "LLMs4EU"; and the BARCELONA Supercomputing Center through the project "TIFON".

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data presented in the study are contained within the manuscript. The complete implementation required to reproduce the experimental analysis and to compute the AI energy scores presented in this paper is publicly available in the following GitHub repository: https://github.com/drdezarza/energy_multilingual_llm_evaluation (accessed on 12 February 2026). This repository contains all necessary code for multilingual data handling, GPU energy measurement aggregation, statistical analysis, and figure generation. Detailed instructions are provided to enable full replication of the results reported in this study and to support further energy–performance evaluation of Large Language Models.

Acknowledgments: This research was supported by the LUXEMBOURG Institute of Science and Technology through the projects "ADIALab-MAST" and "LLMs4EU" (Grant Agreement No 101198470) and the BARCELONA Supercomputing Center through the project "TIFON" (File number MIG-20232039). Mauro Liz would also like to thank Universidad Pontificia Comillas for the opportunity to participate in the international exchange program with the Department of Electrical and Computer Engineering at Boston University.

Conflicts of Interest: The authors declare that they have no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

AI	Artificial Intelligence
CJK	Chinese–Japanese–Korean (languages/scripts)
CO_2	Carbon dioxide
EU	European Union
FP16	16-bit Floating Point (half precision)
GQA	Grouped-Query Attention
GPU	Graphics Processing Unit
HBM3e	High Bandwidth Memory (3e generation)
HELM	Holistic Evaluation of Language Models
KV	Key–Value (cache)
LLM	Large Language Model
MMLU	Massive Multitask Language Understanding
NVML	NVIDIA Management Library
SSM	State Space Model
Wh	Watt-hour

References

1. de Curtò, J.; de Zarzà, I.; Calafate, C.T. LLM Multi-agent Decision Optimization. In *Proceedings of the Agents and Multi-Agent Systems: Technologies and Applications, Madeira, Portugal, 19–21 June 2024*; Jezic, G., Chen-Burger, Y.H., Kušek, M., Šperka, R., Howlett, R.J., Jain, L.C., Eds.; Springer: Singapore, 2025; pp. 3–15.
2. Romera-Paredes, B.; Barekatain, M.; Novikov, A.; Balog, M.; Kumar, M.P.; Dupont, E.; Ruiz, F.J.; Ellenberg, J.S.; Wang, P.; Fawzi, O.; et al. Mathematical Discoveries from Program Search with Large Language Models. *Nature* **2024**, *625*, 468–475. [CrossRef] [PubMed]
3. Thirunavukarasu, A.J.; Ting, D.S.J.; Elangovan, K.; Gutierrez, L.; Tan, T.F.; Ting, D.S.W. Large Language Models in Medicine. *Nat. Med.* **2023**, *29*, 1930–1940. [CrossRef]
4. Kasneci, E.; Seßler, K.; Küchemann, S.; Bannert, M.; Dementieva, D.; Fischer, F.; Gasser, U.; Groh, G.; Günnemann, S.; Hüllermeier, E.; et al. ChatGPT for Good? On Opportunities and Challenges of Large Language Models for Education. *Learn. Individ. Differ.* **2023**, *103*, 102274. [CrossRef]
5. International Energy Agency. Electricity 2024: Analysis and Forecast to 2026. 2024. Available online: https://www.iea.org/reports/electricity-2024 (accessed on 1 February 2026).
6. de Vries, A. The Growing Energy Footprint of Artificial Intelligence. *Joule* **2023**, *7*, 2191–2194. [CrossRef]
7. Hendrycks, D.; Burns, C.; Basart, S.; Zou, A.; Mazeika, M.; Song, D.; Steinhardt, J. Measuring Massive Multitask Language Understanding. In Proceedings of the International Conference on Learning Representations, Vienna, Austria, 4 May 2021.
8. Srivastava, A.; Rastogi, A.; Rao, A.; Shoeb, A.A.M.; Abid, A.; Fisch, A.; Brown, A.R.; Santoro, A.; Gupta, A.; Garriga-Alonso, A.; et al. Beyond the Imitation Game: Quantifying and Extrapolating the Capabilities of Language Models. *Trans. Mach. Learn. Res.* **2023**. Available online: https://openreview.net/forum?id=uyTL5Bvosj (accessed on 1 February 2026).
9. Hendrycks, D.; Burns, C.; Kadavath, S.; Arora, A.; Basart, S.; Tang, E.; Song, D.; Steinhardt, J. Measuring Mathematical Problem Solving With the MATH Dataset. In Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks, Virtual, 6–14 December 2021.
10. Hugging Face. Open LLM Leaderboard. 2024. Available online: https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard (accessed on 1 February 2026).

11. Liang, P.; Bommasani, R.; Lee, T.; Tsipras, D.; Soylu, D.; Yasunaga, M.; Zhang, Y.; Narayanan, D.; Wu, Y.; Kumar, A.; et al. Holistic Evaluation of Language Models. *Trans. Mach. Learn. Res.* **2023**. Available online: https://openreview.net/forum?id=iO4LZibEqW (accessed on 1 February 2026).
12. Schwartz, R.; Dodge, J.; Smith, N.A.; Etzioni, O. Green AI. *Commun. ACM* **2020**, *63*, 54–63. [CrossRef]
13. European Parliament and Council of the European Union. Regulation (EU) 2024/1689 of the European Parliament and of the Council Laying Down Harmonised Rules on Artificial Intelligence (AI Act). *Off. J. Eur. Union* **2024**, *L*, 1–144. Available online: http://data.europa.eu/eli/reg/2024/1689/oj (accessed on 1 February 2026).
14. Strubell, E.; Ganesh, A.; McCallum, A. Energy and Policy Considerations for Deep Learning in NLP. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; pp. 3645–3650.
15. Lacoste, A.; Luccioni, A.; Schmidt, V.; Dandres, T. Quantifying the Carbon Emissions of Machine Learning. *arXiv* **2019**, arXiv:1910.09700. [CrossRef]
16. Courty, B.; Schmidt, V.; Luccioni, S.; Goyal-Kamal; Coutarel, M.; Feld, B.; Lecourt, J.; Connell, L.; Saboni, A.; Inimaz; et al. CodeCarbon: mlco2/codecarbon v2.4.1. 2024. Available online: https://zenodo.org/records/11171501 (accessed on 1 February 2026) [CrossRef]
17. Luccioni, S.; Jernite, Y.; Pierrard, R.; Moutawwakil, I.; Mitchell, M.; Gamazaychikov, B.; Qu, J.; Bi, B.; Weimann, M.; Downey, H.; et al. AI Energy Score: Standardized Energy Efficiency Ratings for AI Models. 2025. Available online: https://huggingface.github.io/AIEnergyScore/ (accessed on 1 February 2026).
18. Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; Bowman, S.R. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In Proceedings of the International Conference on Learning Representations, New Orleans, LA, USA, 6–9 May 2019.
19. Wang, A.; Pruksachatkun, Y.; Nangia, N.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; Bowman, S.R. SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 1–15.
20. Bender, E.M.; Koller, A. On the Dangers of Statistical Language Modeling. In Proceedings of the 19th European Conference on Artificial Intelligence, Lisbon, Portugal, 16–20 August 2011; pp. 704–709.
21. Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. Llama 2: Open Foundation and Fine-Tuned Chat Models. *arXiv* **2023**, arXiv:2307.09288. [CrossRef]
22. Gu, A.; Dao, T. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. In Proceedings of the International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024.
23. Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q.V.; Zhou, D. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 24824–24837.
24. Wang, X.; Wei, J.; Schuurmans, D.; Le, Q.; Chi, E.; Narang, S.; Chowdhery, A.; Zhou, D. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In Proceedings of the International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023.
25. Zhou, L.; Schellaert, W.; Martínez-Plumed, F.; Moros-Daval, Y.; Ferri, C.; Hernández-Orallo, J. Larger and more instructable language models become less reliable. *Nature* **2024**, *634*, 61–68. [CrossRef] [PubMed]
26. Elazar, Y.; Kassner, N.; Ravfogel, S.; Ravichander, A.; Hovy, E.; Schütze, H.; Goldberg, Y. Measuring and Improving Consistency in Pretrained Language Models. *Trans. Assoc. Comput. Linguist.* **2021**, *9*, 1012–1031. [CrossRef]
27. Lu, Y.; Bartolo, M.; Moore, A.; Riedel, S.; Stenetorp, P. Fantastically Ordered Prompts and Where to Find Them: Overcoming Few-Shot Prompt Order Sensitivity. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, Dublin, Ireland, 22–27 May 2022; Volume 1, pp. 8086–8098. [CrossRef]
28. Zhu, K.; Zhao, Q.; Chen, H.; Wang, J.; Xie, X. PromptBench: A Unified Library for Evaluation of Large Language Models. *J. Mach. Learn. Res.* **2024**, *25*, 1–22.
29. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [CrossRef]
30. Cai, H.; Gan, C.; Wang, T.; Zhang, Z.; Han, S. Once-for-All: Train One Network and Specialize It for Efficient Deployment. In Proceedings of the International Conference on Learning Representations, Virtual, 26–30 April 2020.
31. Henderson, P.; Hu, J.; Romoff, J.; Brunskill, E.; Jurafsky, D.; Pineau, J. Towards the Systematic Reporting of the Energy and Carbon Footprints of Machine Learning. *J. Mach. Learn. Res.* **2020**, *21*, 1–43.
32. Luccioni, A.S.; Viguier, S.; Ligozat, A.L. Power Hungry Processing: Watts Driving the Cost of AI Deployment? In Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency, Rio de Janeiro, Brazil, 3–6 June 2024; pp. 85–99.
33. Patterson, D.; Gonzalez, J.; Le, Q.; Liang, C.; Munguia, L.M.; Rothchild, D.; So, D.; Texier, M.; Dean, J. Carbon Emissions and Large Neural Network Training. *arXiv* **2021**, arXiv:2104.10350. [CrossRef]
34. Dodge, J.; Prewitt, T.; des Combes, R.T.; Odber, E.; Schwartz, R.; Strubell, E.; Luccioni, A.S.; Smith, N.A.; DeCario, N.; Buchanan, W. Measuring the Carbon Intensity of AI in Cloud Instances. In Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency, Seoul, Republic of Korea, 21–24 June 2022; pp. 1877–1894.

35. García-Martín, E.; Rodrigues, C.F.; Riley, G.; Grahn, H. Estimation of Energy Consumption in Machine Learning. *J. Parallel Distrib. Comput.* **2019**, *134*, 75–88. [CrossRef]
36. Bannour, N.; Ghannay, S.; Nevéol, A.; Ligozat, A.L. Evaluating the Carbon Footprint of NLP Methods: A Survey and Analysis of Existing Tools. In Proceedings of the Second Workshop on Simple and Efficient Natural Language Processing, Virtual, 5 August 2021; pp. 11–21.
37. Hu, J.; Ruder, S.; Siddhant, A.; Neubig, G.; Firat, O.; Johnson, M. XTREME: A Massively Multilingual Multi-task Benchmark for Evaluating Cross-lingual Generalisation. In *Proceedings of the 37th International Conference on Machine Learning, Virtual, 12–18 July 2020*; PMLR: Cambridge, MA, USA, 2020; pp. 4411–4421.
38. Liang, Y.; Duan, N.; Gong, Y.; Wu, N.; Guo, F.; Qi, W.; Gong, M.; Shou, L.; Jiang, D.; Cao, G.; et al. XGLUE: A New Benchmark Dataset for Cross-lingual Pre-training, Understanding and Generation. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, Virtual, 16–20 November 2020; pp. 6008–6018.
39. Conneau, A.; Khandelwal, K.; Goyal, N.; Chaudhary, V.; Wenzek, G.; Guzmán, F.; Grave, E.; Ott, M.; Zettlemoyer, L.; Stoyanov, V. Unsupervised Cross-lingual Representation Learning at Scale. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Virtual, 5–10 July 2020; pp. 8440–8451.
40. Ahuja, K.; Diddee, H.; Hada, R.; Ochieng, M.; Ramesh, K.; Jain, P.; Nambi, A.; Ganu, T.; Segal, S.; Ahmed, M.; et al. MEGA: Multilingual Evaluation of Generative AI. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Singapore, 3–7 December 2023; pp. 4232–4267.
41. Lai, V.D.; Ngo, N.T.; Veyseh, A.P.B.; Man, H.; Dernoncourt, F.; Bui, T.; Nguyen, T.H. ChatGPT Beyond English: Towards a Comprehensive Evaluation of Large Language Models in Multilingual Learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*; Association for Computational Linguistics: Singapore, 2023; pp. 13171–13189.
42. Rust, P.; Pfeiffer, J.; Vulić, I.; Ruder, S.; Gurevych, I. How Good is Your Tokenizer? On the Monolingual Performance of Multilingual Language Models. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics, Virtual, 1–6 August 2021; pp. 3118–3135.
43. Petrov, A.; La Malfa, E.; Torr, P.; Biber, A. Language Model Tokenizers Introduce Unfairness Between Languages. *Adv. Neural Inf. Process. Syst.* **2024**, *36*, 36963–36990.
44. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1–11.
45. Ainslie, J.; Lee-Thorp, J.; de Jong, M.; Zemlyanskiy, Y.; Lebrón, F.; Sanghai, S. GQA: Training Generalized Multi-Query Transformer Models from Multi-Head Checkpoints. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Singapore, 3–7 December 2023; pp. 4895–4901.
46. Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. Transformers: State-of-the-Art Natural Language Processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Virtual, 19 November 2020; pp. 38–45.
47. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; Volume 32.
48. de Curtò, J.; de Zarzà, I. Metamorphic Testing for Semantic Invariance in Large Language Models. *IEEE Access* **2025**, *13*, 214772–214791. [CrossRef]
49. Ribeiro, M.T.; Wu, T.; Guestrin, C.; Singh, S. Beyond Accuracy: Behavioral Testing of NLP Models with CheckList. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Virtual, 5–10 July 2020; pp. 4902–4912.
50. Turpin, M.; Michael, J.; Perez, E.; Bowman, S. Language Models Don't Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting. *Adv. Neural Inf. Process. Syst.* **2024**, *36*, 74952–74965.
51. Virtanen, P.; Gommers, R.; Oliphant, T.E.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J.; et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nat. Methods* **2020**, *17*, 261–272. [CrossRef] [PubMed]
52. Friedman, M. The Use of Ranks to Avoid the Assumption of Normality Implicit in the Analysis of Variance. *J. Am. Stat. Assoc.* **1937**, *32*, 675–701. [CrossRef]
53. Mann, H.B.; Whitney, D.R. On a Test of Whether One of Two Random Variables Is Stochastically Larger than the Other. *Ann. Math. Stat.* **1947**, *18*, 50–60. [CrossRef]
54. Holm, S. A Simple Sequentially Rejective Multiple Test Procedure. *Scand. J. Stat.* **1979**, *6*, 65–70.
55. Cohen, J. *Statistical Power Analysis for the Behavioral Sciences*, 2nd ed.; Lawrence Erlbaum Associates: Mahwah, NJ, USA, 1988.
56. Spearman, C. The Proof and Measurement of Association between Two Things. *Am. J. Psychol.* **1904**, *15*, 72–101. [CrossRef]
57. Shi, F.; Chen, X.; Misra, K.; Scales, N.; Dohan, D.; Chi, E.H.; Schärli, N.; Zhou, D. Large Language Models Can Be Easily Distracted by Irrelevant Context. In *Proceedings of the 40th International Conference on Machine Learning, Honolulu, HI, USA, 23–29 July 2023*; PMLR: London, UK, 2023; Volume 202, pp. 31210–31227.

58. DeepSeek-AI; Guo, D.; Yang, D.; Zhang, H.; Song, J.; Wang, P.; Zhu, Q.; Xu, R.; Zhang, R.; Ma, S.; et al. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. *arXiv* **2025**, arXiv:2501.12948. [CrossRef]
59. Teknium, R.; Jin, R.; Suphavadeeprasit, J.; Mahan, D.; Quesnelle, J.; Li, J.; Guang, C.; Sands, S.; Malhotra, K. Hermes 4 Technical Report. *arXiv* **2025**, arXiv:2508.18255. [CrossRef]