WISCONSIN
UNIVERSITY OF WISCONSIN–MADISON

PalmBench: A Comprehensive Benchmark of Compressed Large Language Models on Mobile Platforms

Yilong Li[1], Jingyu Liu[1], Hao Zhang[1], M Badri Narayanan[1], Utkarsh Sharma[1], Shuai Zhang[2], Yijing Zeng[1], Jayaram Raghuram[1], Suman Banerjee[1]

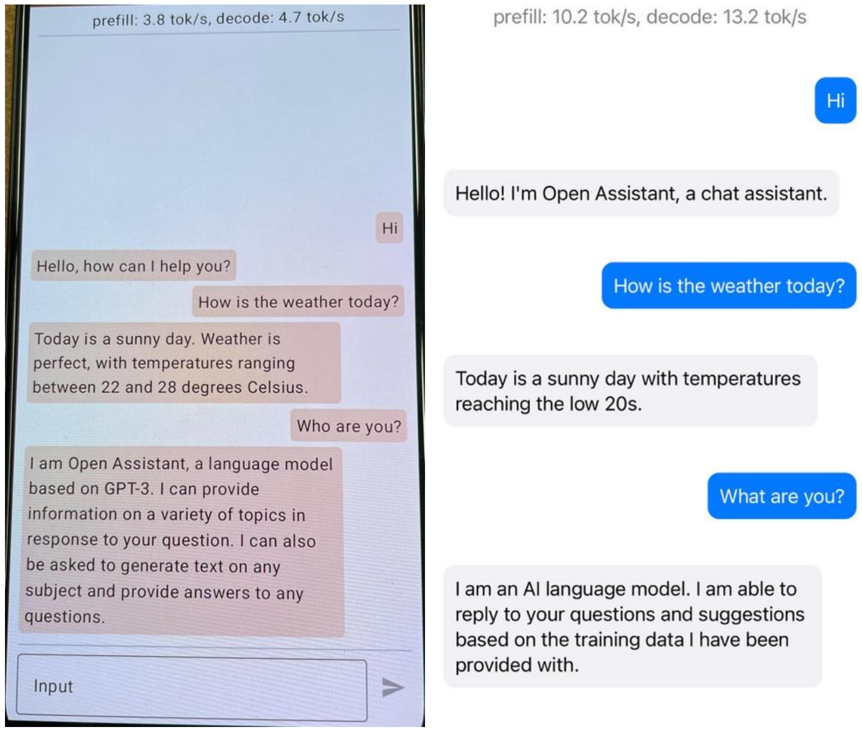
[1]University of Wisconsin-Madison, [2]Amazon AWS AI

1. Introduction

Background

MLC-LLM

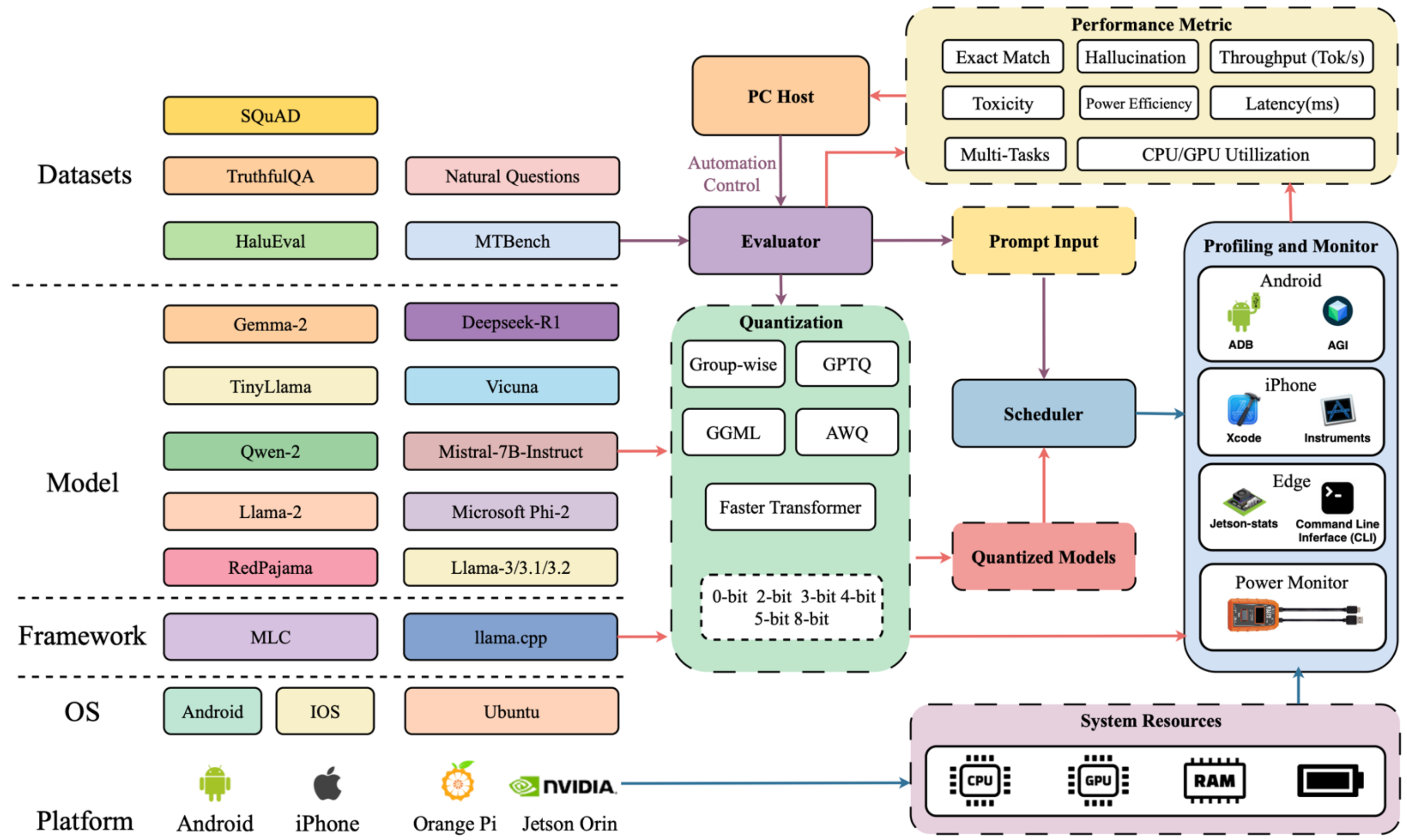
Llama.cpp



Running LLMs locally on smartphones is becoming increasingly important—for applications where privacy, reliability, and low latency matter.

1. Introduction

Overview of framework



Overview and workflow of PalmBench--our evaluation and benchmarking framework for Large Language Models (LLMs) on mobile devices.

2. Methods

Metrics

Table 2: Metrics for evaluating the performance of LLMs on mobile devices. Memory usage includes both the model loaded to the memory and the framework program running on devices.

Metric	Definition
CPU Utilization (%)	Percentage of the total processor cycles consumed by LLM
GPU Utilization (%)	Percentage of the total GPU computing resource during LLM inference
Memory Footprint (GB)	Measurement of main memory used by the LLM application
Memory Utilization (%)	Percentage of main memory used by the LLM application
Throughput (Tok / s)	Number of output tokens per second generated by the LLM
Output Matching	Accuracy degradation of the compressed model relative to the original model
Toxicity	Toxicity score calculated on 25k sentences by Perspective API
Hallucination (%)	Percentage of erroneous or random outputs not related to the questions

2. Methods

Test Devices

Table 6: Mobile and edge devices for evaluation.

Device	SoC	Memory (GB)	Framework Support
iOS 17.6.1			
iPhone 12 Pro	A14 Bionic	6GB	MLC
iPhone 15 Pro	A17 Bionic	8GB	MLC
iPhone 16 Pro	A18 Pro	8GB	MLC
Android 15			
Pixel 4	Snapdragon 855	6GB	MLC/llama.cpp
Pixel 5a	Snapdragon 765G	6GB	MLC/llama.cpp
Pixel 7	Exynos 5300	8GB	MLC/llama.cpp
S22 Ultra	Snapdragon 8 Gen 1	8GB	MLC/llama.cpp
Ubuntu 14.04.06 LTS			
Orange Pi 5	RK3588	8GB	MLC/llama.cpp
Jetson Orin Nano	NVIDIA Orin	8GB	MLC/llama.cpp

2. Methods

Datasets

Stanford Question Answering Dataset (SQuAD) is a reading comprehension dataset, consisting of questions posed by crowdworkers on a set of Wikipedia articles, where the answer to every question is a segment of text, or *span*, from the corresponding reading passage, or the question might be unanswerable.

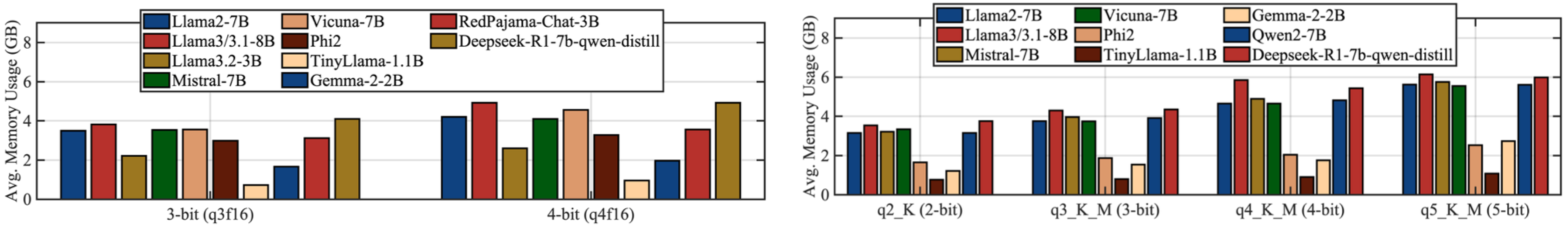
Natural Questions contains real user questions submitted to Google search, with answers provided by annotators from Wikipedia. NQ is designed to train and evaluate automatic question-answering systems.

MMLU (Massive Multitask Language Understanding) is a benchmark created to measure the knowledge acquired during pretraining by evaluating models exclusively in zero-shot and few-shot settings.

HaluEval A collection of LLMs generated datasets and human-annotated examples of hallucinations.

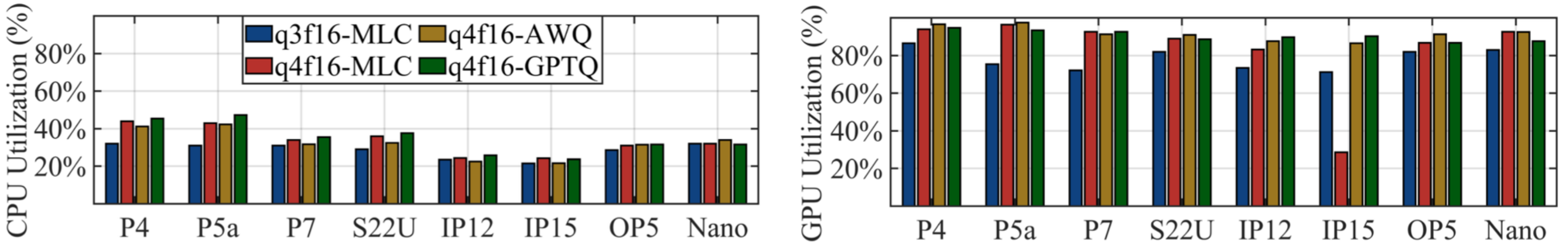
TruthfulQA A benchmark to measure whether a language model is truthful in generating answers to questions.

Resource Usage



(a) MLC (b) llama.cpp

Figure 2: Average memory usage (GB) while running MLC and llama.cpp.



(a) CPU (b) GPU

Figure 3: CPU and GPU usage during inference of RedPajama-INCITE-3B across different quantizations.

3. Results

Resource Usage

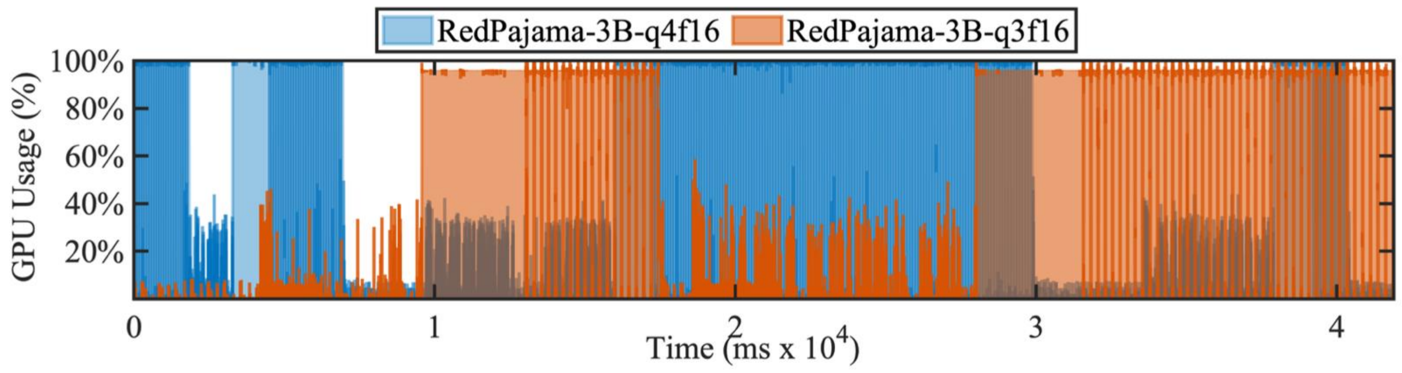
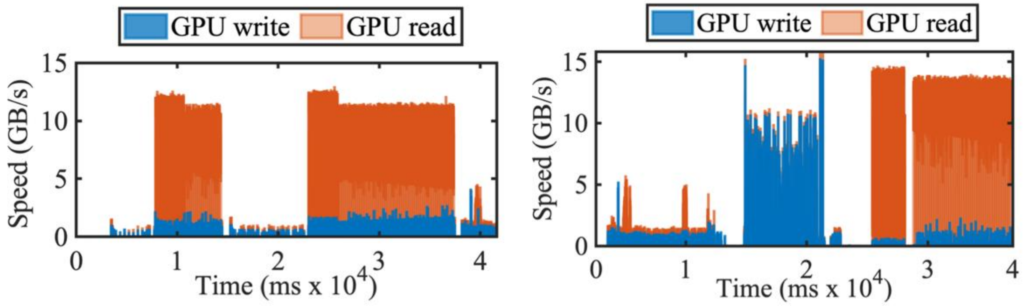


Figure 4: GPU Utilization (%) timeline for 3-bit and 4-bit quantized RedPajama models on Google Pixel 7.

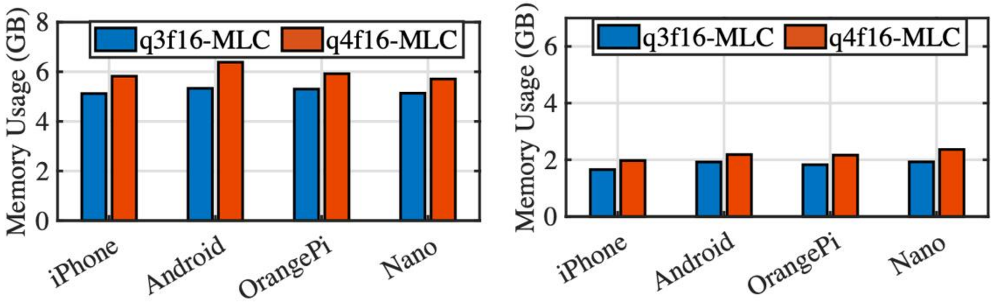
3. Results

Resource Usage



(a) Llama-3-8B-q3f16 (b) Llama-3-8B-q4f16

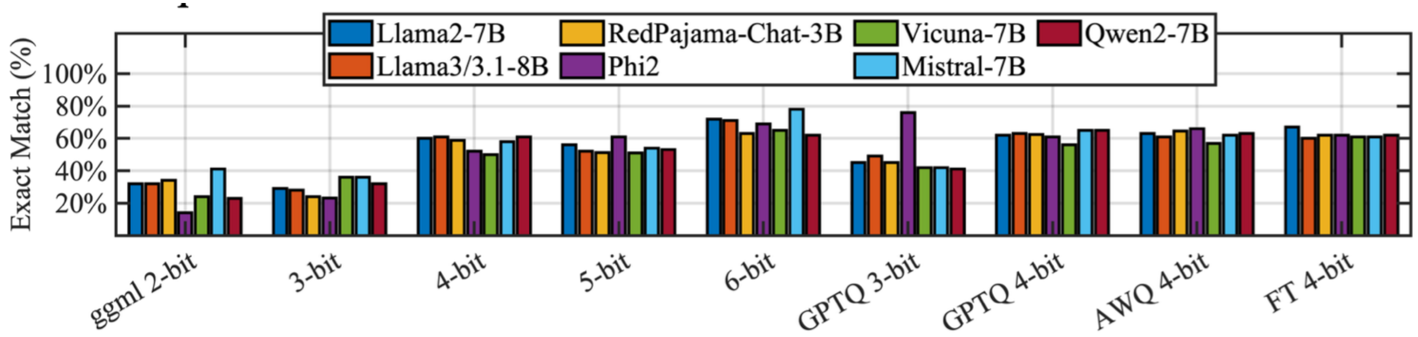
Figure 5: GPU memory read/write speed while running LLaMa-3-8B-Instruct in 3-bit and 4-bit quantization on Pixel 7.



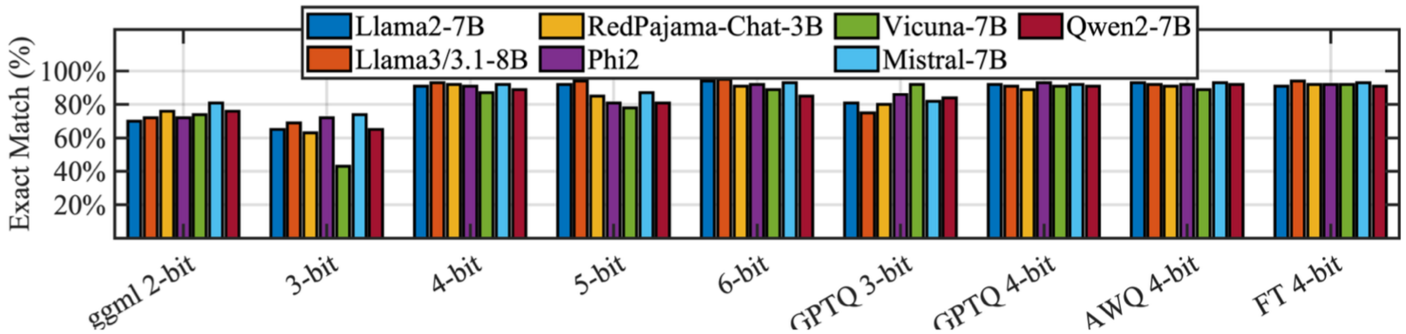
(a) Llama-3-8B for all platforms (b) Gemma-2B for all platforms

Figure 6: Measured memory usage (GB) across different platforms using Llama-3-8B and Gemma-2-2B by MLC LLM to compare the memory usage between large model (Llama-3-8B) and small model (Gemma-2-2B).

3. Results



(a) Exact Match



(b) F1 score

Figure 8: Scores of exact match and F1 score to examine the performance loss after models are quantized.

3. Results

Table 3: Evaluation of temperature and power consumption during inference of Llama3-8B across different mobile phones

Llama-3.2-3B 3-bit quantization								
Platforms	Pixel 4	Pixel 5a	Pixel 7	S22 Ultra	iPhone 12 Pro	iPhone 15 Pro	Orange Pi 5	Jetson Nano
Peak Temp. (°)	47.8	53.2	52.1	52.8	47.3	45.3	71.5	61.5
Avg. Temp. (°)	28.3	28.7	28.5	27.2	27.2	25.3	47.5	43.3
Power Consumed (mWh)	13.32	12.98	14.54	13.25	11.21	10.13	25.4	22.3
Llama-3.2-3B 4-bit quantization								
Platforms	Pixel 4	Pixel 5a	Pixel 7	S22 Ultra	iPhone 12 Pro	iPhone 15 Pro	Orange Pi 5	Jetson Nano
Peak Temp. (°)	53.1	54.8	52.6	48.7	47.2	46.3	75.4	69.5
Avg. Temp. (°)	28.2	29.2	30.3	27.8	26.4	24.2	52.4	45.3
Power Consumed (mWh)	14.23	13.51	14.68	15.26	13.12	13.05	27.8	25.6

Table 4: Evaluation of Hallucination Outputs across Different Quantization Levels in Llama3-8B.

Hallucination

Table 4: Evaluation of Hallucination Outputs across Different Quantization Levels in Llama3-8B.

Quantization	2-bit	3-bit	4-bit (GPTQ)	8-bit	4-bit (ggml)	4-bit (AWQ)	4-bit (FT)
Halucination	32.7%	37.5%	9.1%	8.1%	12.5%	8.9%	8.7%
TruthfulQA	76%	73%	92.1%	91.4%	90.1%	92.3%	91.5%
Toxicity	46.243	64.098	28.679	23.965	41.107	30.072	29.405

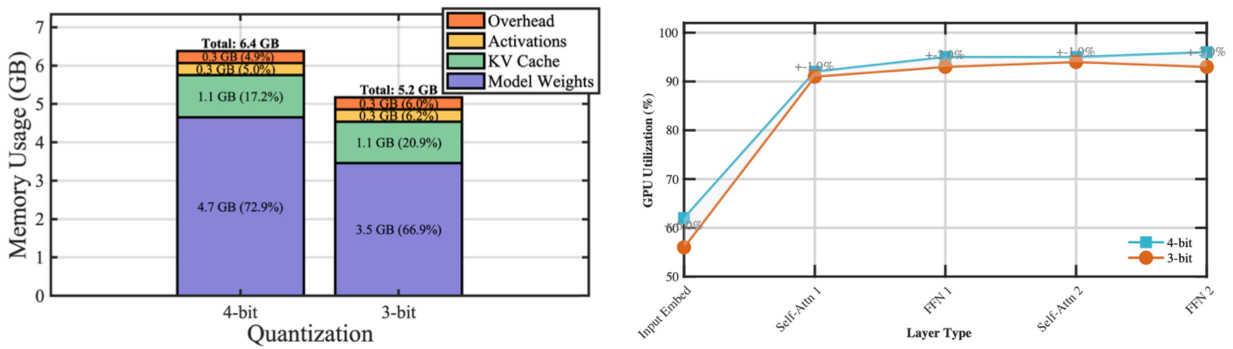
Table 5: Evaluation of Hallucination Outputs across Different Quantization Levels in Gemma-2-2B.

Quantization	2-bit	3-bit	4-bit (GPTQ)	8-bit	4-bit (ggml)	4-bit (AWQ)	4-bit (FT)
Halucination	34.2%	32.5%	9.1%	7.9%	14.5%	8.9%	8.7%
TruthfulQA	72%	73.2%	91.1%	92.4%	84.5%	89.3%	90.5%
Toxicity	36.121	65.22	25.045	23.102	24.215	31.202	25.455

Lower-bit quantization typically increases hallucinations and toxicity. Notably, 3-bit quantization occasionally performs worse than 2-bit group-wise quantization and all 4-bit methods, resulting in more hallucinations and toxic outputs. Initially, we believed the model and quantization algorithms primarily caused hallucinations. However, we recently discovered that inference framework implementations and mismatched parameters can also lead to hallucinated or random outputs.

For instance, quantization and training parameters may not be correctly reflected in the inference configuration, such as the batch size setting in Qwen2 significantly impacting the occurrence of hallucinations in MLC. Since addressing these issues falls outside the scope of this paper, we plan to investigate them in future work.

3. Results



(a) Memory Breakdown (b) Layer-wise GPU Utilization

Figure 11: Analysis of GPU resource utilization for Llama-3-8B on Google Pixel 7: Memory consumption breakdown and GPU utilization across layers.