

EFFICIENT AND TRUSTWORTHY ARTIFICIAL INTELLIGENCE SYSTEMS: EDGE MODEL OPTIMIZATION AND ADVERSARIAL ROBUSTNESS IN LARGE LANGUAGE MODELS

**Thesis Submitted
in Partial Fulfillment of the Requirements for the
Degree of**

MASTER OF TECHNOLOGY

in

Computer Science and Engineering

by

ABHISHEK KUMAR

(Roll No. 2K24/CSE/23)

Under the Supervision of

Dr. Pawan Singh Mehra

Assistant Professor
Department of Computer Science and Engineering
Delhi Technological University



**Department of Computer Science and Engineering
DELHI TECHNOLOGICAL UNIVERSITY**

(Formerly Delhi College of Engineering)
Shahbad Daultapur, Main Bawana Road, Delhi-110042, India

May, 2026



DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-110042, India

ACKNOWLEDGEMENT

I express my sincere gratitude to my supervisor, **Dr. Pawan Singh Mehra**, Assistant Professor, Department of Computer Science and Engineering, Delhi Technological University, for his continuous guidance, technical insight, and constant encouragement throughout the preparation of this thesis report. His detailed feedback at every stage of the work helped me sharpen the focus and structure of this research.

I am grateful to the Department of Computer Science and Engineering, Delhi Technological University, for providing the academic environment, computing resources, and learning culture that enabled me to study the rapidly evolving research areas of edge artificial intelligence, model compression, lightweight neural networks, and the emerging security concerns surrounding large language models. The exposure to systematic literature review methodology, benchmarking practice, and comparative analysis was crucial in shaping the methodology of this work.

I am especially thankful to the researchers and open-source communities whose work forms the foundation of this thesis. Contributions to lightweight convolutional architectures, TinyML deployment, small language models, knowledge distillation, quantization, prompt-injection security, retrieval-augmented generation attacks, and benchmark design provided the technical basis for the thesis-level expansion. I also acknowledge the role of open benchmark datasets such as ImageNet, COCO, GLUE, TruthfulQA, and HaluEval, which have made systematic evaluation across the two domains possible.

Finally, I thank my family, friends, and peers, who supported me directly or indirectly during this academic work. Their constant motivation helped me complete this report with dedication and discipline.

Abhishek Kumar

Roll No: 2K24/CSE/23

M.Tech (CSE)

Delhi Technological University



DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-42

CANDIDATE'S DECLARATION

I, Abhishek Kumar (Roll No. 2K24/CSE/23), hereby certify that the work which is being presented in the thesis entitled **“Efficient and Trustworthy Artificial Intelligence Systems: Edge Model Optimization and Adversarial Robustness in Large Language Models”** in partial fulfillment of the requirements for the award of the Degree of **Master of Technology**, submitted in the Department of Computer Science and Engineering, Delhi Technological University is an authentic record of my own work carried out during the period from August, 2024 to May, 2026 under the supervision of Dr. Pawan Singh Mehra.

The matter presented in the thesis has not been submitted by me for the award of any other degree of this or any other Institute.

Candidate's Signature

This is to certify that the student has incorporated all the corrections suggested by the examiners in the thesis and the statement made by the candidate is correct to the best of our knowledge.

Signature of Supervisor(s)

Signature of External Examiner



DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-42

CERTIFICATE BY THE SUPERVISOR(s)

Certified that **Abhishek Kumar** (Enrollment No. 2K24/CSE/23) has carried out their research work presented in this thesis entitled **“Efficient and Trustworthy Artificial Intelligence Systems: Edge Model Optimization and Adversarial Robustness in Large Language Models”** for the award of **Master of Technology** from Department of Computer Science and Engineering, Delhi Technological University, Delhi, under my supervision. The thesis embodies results of original work, and studies are carried out by the student himself and the contents of the thesis do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Signature

(Dr. Pawan Singh Mehra)

(Assistant Professor)

(Department of Computer Science and Engineering,
Delhi Technological University, Delhi-110042, India)

Date: _____

Efficient and Trustworthy Artificial Intelligence Systems: Edge Model Optimization and Adversarial Robustness in Large Language Models

Abhishek Kumar

ABSTRACT

Modern AI systems must satisfy two distinct but equally important requirements: *efficiency*, the ability to deliver intelligence under tight memory, latency, and energy budgets at the edge of the network; and *trustworthiness*, the ability of large generative models to remain factually reliable and robust against adversarial manipulation. This thesis investigates both through two complementary contributions.

The first contribution is a structured survey of Edge AI covering 2015–2025. It analyses lightweight CNNs (MobileNet, EfficientNet, SqueezeNet, ShuffleNetV2), real-time object detectors (YOLOv8, SSD-MobileNet), TinyML systems (MCUNet, DeepConv-LSTM, TinyBERT), and sub-10B SLMs (LLaMA 3, Phi-3 Mini, Qwen2.5-VL, GLM-4), with four compression and routing techniques: quantization, pruning, knowledge distillation, and Mixture-of-Experts. Twelve representative models are compared across accuracy, parameter count, memory, target hardware, and trade-offs. The survey is supplemented by a focused experimental validation of the INT8 quantization trade-off on MobileNetV2: three TFLite regimes (FP32, INT8 dynamic-range, INT8 full-integer) are compared on Imagenette, confirming the published $\sim 3.5\times$ size compression at ≤ 1.3 percentage points accuracy drop and revealing that the latency benefit of quantization is hardware-dependent. Open challenges include activation-memory bottlenecks, hardware portability, on-device training, multimodal edge intelligence, energy benchmarking, and edge security.

The second contribution is a systematic survey of adversarial hallucination attacks against LLMs, comparing open-source models (LLaMA 2, Mistral 7B, Falcon 40B) and closed-source models (GPT-4, Gemini 1.5 Pro, Claude 3 Opus). It introduces a taxonomy by attacker access: white-box (gradient and fine-tuning backdoor attacks, open-source only), black-box (factual corruption, CoT hijacking, authority persona; both classes), and RAG-layer (knowledge-base poisoning and indirect injection; both classes). Benchmarks (TruthfulQA, HaluEval, FActScore) are reviewed for adversarial suitability and shown to be inadequate for measuring robustness under active attack. The central finding is that closed-source release does not buy safety against retrieval-layer attacks: while open-source weights expose a wider attack surface (gradient-based attacks, fine-tuning backdoors), closed-source models—despite stronger alignment—remain

equally vulnerable to RAG injection because retrieval is architecturally decoupled from alignment. A risk-graded attack-surface matrix and an evidence-based rubric for the High/Medium/Low risk grading are provided.

Together, the contributions give a unified view of contemporary AI deployment: edge optimization explains how intelligence can be made efficient enough to run close to the data, and adversarial robustness explains how it can be made trustworthy enough to rely upon. The thesis identifies the security of compressed and quantized models—the very models deployed at the edge—as the single most important cross-cutting research gap, supports the claim with a meta-analysis synthesised from prior work, and outlines future directions in hardware-algorithm co-design, secure RAG, adversarial hallucination benchmarking, and alignment-as-security.

Keywords: Edge AI, TinyML, model compression, quantization, small language models, large language models, adversarial attacks, hallucinations, prompt injection, RAG, trustworthy AI.

LIST OF PUBLICATIONS

1. **A. Kumar and P. S. Mehra, “Edge AI Models: A Comparative Survey of Architectures, Optimization Techniques, and Deployment Strategies,” accepted for presentation at the 4th International Conference on Networks and Cryptology (NETCRYPT-2026), 2026.**
2. **A. Kumar and P. S. Mehra, “A Systematic Survey on Adversarial Hallucination Attacks in LLMs: Open-Source vs. Closed-Source Models,” submitted to the International Conference on Intelligent Computing, Control Communication (ICICCC-2026), 2026.**

CONTENTS

Acknowledgement	i
Candidate's Declaration	ii
Certificate	iii
Abstract	iv
List of Figures	x
List of Tables	xi
1 Introduction	1
1.1 Background and Motivation	1
1.2 Problem Domain Overview	1
1.3 Existing Challenges	2
1.4 Problem Statement	2
1.5 Research Objectives	3
1.6 Scope of the Research	3
1.7 Applications and Relevance	4
1.8 Thesis Organization	4
2 Literature Review	5
2.1 Introduction	5
2.2 Edge Computing Hierarchy and Foundations	5
2.3 Lightweight CNNs, Edge Detection, TinyML and SLMs	5
2.4 Foundations of Model Compression	6
2.5 LLM Architecture, Alignment, and Hallucination	6
2.6 Adversarial Attack Techniques against LLMs	6
2.7 Defenses, Open-Source vs. Closed-Source, and Comparative Review .	7
2.8 Research Gaps and Summary	8
3 Proposed Work Part I: Edge AI Models, Optimization Techniques, and Deployment Strategies	9
3.1 Chapter Introduction and Motivation	9
3.2 System Overview: Edge AI Taxonomy	9

3.3	Foundations of Model Compression	9
3.3.1	Quantization	9
3.3.2	Pruning	10
3.3.3	Knowledge Distillation	10
3.3.4	Mixture of Experts (MoE)	10
3.4	Literature Search Methodology	11
3.5	Lightweight CNN Architectures	11
3.6	Object Detection, TinyML, and Small Language Models	12
3.7	Comparative Analysis: Twelve Representative Models	12
3.8	Accuracy vs. Model Size Trade-off and Discussion	13
3.9	Experimental Validation: Quantization Trade-off	14
3.9.1	Experimental Setup	14
3.9.2	Results	15
3.9.3	Discussion	15
3.10	Challenges, Research Gaps and Future Directions	16
3.11	Chapter Summary	17
4	Proposed Work Part II: Adversarial Hallucination Attacks on Large Language Models	18
4.1	Chapter Introduction and Motivation	18
4.2	Background: LLM Architecture and Hallucination Taxonomy	18
4.3	Threat Model	19
4.4	System Overview: Adversarial Hallucination Attack Taxonomy	19
4.5	Literature Search Methodology	19
4.6	Hallucination Benchmarks, Attack Techniques and Defenses	21
4.7	Comparative Analysis: Six Representative Models	22
4.7.1	Model Overview and Configuration	22
4.7.2	Attack Surface Comparison	22
4.7.3	Performance Under Adversarial Pressure and Approaches Compared	23
4.8	Challenges, Discussion, and Future Directions	24
4.8.1	Challenges and Research Gaps	24
4.8.2	Discussion and Future Directions	24
4.9	Chapter Summary	25
5	Results Analysis and Discussion	26
5.1	Introduction	26
5.2	Analysis of the Two Surveys	26
5.2.1	Edge AI Survey Results	26
5.2.2	Adversarial Robustness Survey Results	26
5.3	Unified Interpretation of Both Contributions	27
5.4	Cross-Cutting Research Gaps	28
5.4.1	Robustness of Compressed and Quantized Models under Attack	28

5.4.2	Secure RAG, Tiered Benchmarks, On-Device Defenses, and Federated Robustness	29
5.5	Complexity, Security Implications, and Practical Applicability	30
6	Conclusion, Future Scope and Social Impact	32
6.1	Conclusion	32
6.2	Major Findings	32
6.3	Research Contributions	33
6.4	Limitations	34
6.5	Future Scope	34
6.6	Social Impact	35
6.7	Closing Remarks	35
	References	36

LIST OF FIGURES

3.1	Taxonomy of Edge AI model families covered in this survey, organised by architecture type and target hardware tier.	10
3.2	PRISMA-style flow diagram for the Edge AI literature search.	11
3.3	Accuracy vs. model size for edge vision models. MCUNet achieves strong accuracy within a 1 MB RAM constraint via hardware-aware co-design, outperforming same-size conventional CNNs.	14
3.4	Empirical quantization trade-off for MobileNetV2 on Imagenette. INT8 quantization reduces on-disk size by approximately $3.5\times$ with an accuracy degradation of only 0.91–1.27 percentage points relative to the FP32 baseline.	15
4.1	Threat model: an adversarial hallucination attack is characterised by the tuple (access level, attack goal, attack vector). White-box attacks require open weights; RAG-layer attacks apply to both open- and closed-source classes.	19
4.2	Adversarial hallucination attack taxonomy. “White-box” (red) attacks demand access to the weights and apply only to open-source models. Black-box (green) and RAG-layer (purple) attacks impact both classes equally.	20
4.3	PRISMA-style flow diagram for the adversarial hallucination literature search.	21
5.1	TruthfulQA scores across the six surveyed LLMs. Three open-source models (blue) and one closed-source model (orange) have published scores on this benchmark. Gemini 1.5 Pro and Claude 3 Opus post-date the original benchmark and report on internal evaluations; their scores are reported as N/A here to avoid misleading cross-benchmark comparison.	27
5.2	Meta-analysis: effect of precision reduction on clean accuracy versus adversarial robustness, synthesised from [1][2][3]. Adversarial robustness degrades disproportionately to clean accuracy, and the gap widens as bit-width decreases. The pattern motivates evaluating adversarial robustness on the actual deployment configuration rather than the FP32 reference.	30

LIST OF TABLES

2.1	Comparative overview of selected research directions across efficient and trustworthy AI.	7
3.1	Comparative analysis of Edge AI models. Performance metrics are reported on the benchmark column appropriate to each model family and are <i>not</i> directly comparable across rows from different families (ImageNet top-1, COCO mAP, GLUE retention, and language-model benchmarks measure different things). The table is intended for within-family and cross-family deployment-tier comparison. Min RAM is the deployment memory at the indicated quantization level. † denotes INT4 deployment requirement, ‡ denotes FP16 deployment requirement. . .	13
3.2	Experimental quantization trade-off for MobileNetV2 on the Imagenette validation set. Latency: single-thread CPU, mean of 100 runs. . . .	15
4.1	Comparative analysis of LLMs under adversarial hallucination study. †Parameter count not publicly disclosed. ‡GPT-4 base-model score; RLHF post-training yields substantially higher scores per the same report. *These models post-date the original TruthfulQA benchmark; cross-benchmark comparison would be misleading.	22
4.2	Attack surface comparison: open-source vs. closed-source LLMs. Risk levels: H = High, M = Medium, L = Low, N/A = Not Applicable. . . .	23
5.1	Unified comparison of the two thesis contributions.	28
5.2	Meta-analysis: effect of compression on clean accuracy vs. adversarial robustness, synthesised from cited primary studies. Rows are drawn from the surveys and primary studies cited in this thesis; the table itself is the contribution.	29

LIST OF SYMBOLS, ABBREVIATIONS AND NOMENCLATURE

ACM	Association for Computing Machinery	MCU	Microcontroller Unit
AI	Artificial Intelligence	ML	Machine Learning
API	Application Programming Interface	MLPerf	ML Performance Benchmark
BERT	Bidirectional Encoder Representations from Transformers	MMLU	Massive Multitask Language Understanding
BNN	Binary Neural Network	MoE	Mixture of Experts
CNN	Convolutional Neural Network	NAS	Neural Architecture Search
COCO	Common Objects in Context	NCS	Neural Compute Stick
CoT	Chain of Thought	NLP	Natural Language Processing
CPU	Central Processing Unit	NPU	Neural Processing Unit
FLOPs	Floating Point Operations per Second	ONNX	Open Neural Network Exchange
FL	Federated Learning	RAG	Retrieval-Augmented Generation
FP32	32-bit Floating Point	RAM	Random Access Memory
GCG	Greedy Coordinate Gradient	RLHF	Reinforcement Learning from Human Feedback
GLM	General Language Model	RPi	Raspberry Pi
GLUE	General Language Understanding Evaluation	SFT	Supervised Fine-Tuning
GPU	Graphics Processing Unit	SiLU	Sigmoid Linear Unit
HAR	Human Activity Recognition	SLM	Small Language Model
IID	Independent and Identically Distributed	SRAM	Static Random Access Memory
INT4	4-bit Integer	SSD	Single Shot MultiBox Detector
INT8	8-bit Integer	TPU	Tensor Processing Unit
IoT	Internet of Things		
KB	Knowledge Base		
KD	Knowledge Distillation		
LLaMA	Large Language Model Meta AI		
LLM	Large Language Model		
LoRA	Low-Rank Adaptation		
LSTM	Long Short-Term Memory		
mAP	mean Average Precision		

CHAPTER 1

INTRODUCTION

1.1 Background and Motivation

Artificial intelligence has matured from a research discipline into a deployed engineering practice, and development is no longer constrained primarily by accuracy but by the practical conditions of deployment. Two such conditions motivate this thesis. The first is the constraint of compute, memory, latency, and energy at the edge of the network, where an increasing share of inference workloads must execute on smartphones, microcontrollers, and embedded accelerators rather than on cloud servers. The second is the constraint of trust, where Large Language Models (LLMs) increasingly mediate critical decisions in medicine, law, finance, and security, and where any deviation from factual reliability can be exploited by an adversary. These constraints share a common origin: raw model capability is no longer sufficient. The edge case demands that capability be *compressed* to be served at the data source under strict resource budgets; the trust case demands that capability be *constrained* so that it remains reliable when an attacker probes it. This thesis studies these two pillars in a single framework, covering both the efficiency techniques that make models small enough to deploy and the robustness analysis that determines whether they can be trusted once deployed.

1.2 Problem Domain Overview

The problem domain spans two intersecting areas. The first is Edge AI, concerned with running machine learning models on resource-limited hardware close to where data is generated [4][5]. The edge is a continuum—from cloud servers with hundreds of gigabytes of GPU memory, through edge servers and gateways, mobile devices with one to a few gigabytes of RAM, and finally TinyML-class microcontrollers with hundreds of kilobytes of SRAM and milliwatt-level power budgets [6]. Each tier admits a distinct family of models and a distinct set of optimisation techniques.

The second area is the security and trustworthiness of LLMs. Modern LLMs are trained on web corpora, instruction-tuned via supervised fine-tuning, and aligned to human preferences via RLHF or related techniques [7][8]. Despite this alignment, LLMs *hallucinate*—they generate plausible-sounding but unsupported content [9]—and recent work has shown hallucinations can be *induced* on demand by an adversary through prompt manipulation, retrieval poisoning, chain-of-thought hijacking, or fine-

tuning-based weight poisoning [10][11][12]. This reframes hallucination from a quality issue into a security issue.

This thesis treats Edge AI and LLM adversarial robustness as two complementary slices of one larger question: how do we build AI systems both *efficient* enough to be deployed where they are needed and *trustworthy* enough to be relied upon once deployed?

1.3 Existing Challenges

On the efficiency side, a standard ResNet-50 requires ~ 100 MB of memory, exceeding Cortex-M SRAM capacity by two to three orders of magnitude [6]. In TinyML, the binding constraint is peak *activation memory*, not parameter count, yet most NAS methods optimise for FLOPs or parameters and may produce architectures that are not deployable [6]. Edge hardware is heterogeneous and the portability of quantized models across backends remains fragile [13]; on-device training is harder than inference, federated learning faces communication and non-IID convergence problems [14], and energy-per-inference is rarely reported in a standardised way [15].

On the trust side, existing hallucination benchmarks [16][17][18] measure *passive* factual reliability and were not designed to evaluate models under an active adversary. Closed-source models remain vulnerable at the RAG layer because retrieval is decoupled from alignment [12][19], white-box attacks such as GCG [11] succeed on open-source models and transfer at attenuated rates to closed-source ones, and long-context and vision-language models open new attack surfaces that remain poorly characterised [20]. Current alignment techniques optimise for helpfulness and harmlessness in benign settings, not for adversarial truthfulness, and often fail under active attack [21].

1.4 Problem Statement

The central problem is the dual requirement of efficiency and trustworthiness in modern AI systems:

How can deep neural networks and language models be made efficient enough to operate under the resource constraints of edge hardware, and at the same time how can their generative counterparts be made robust to adversarial hallucination attacks, given that open-source and closed-source release models present fundamentally different attack surfaces?

This problem is addressed through two complementary contributions. Chapter 3 surveys Edge AI architectures, compression techniques, and deployment strategies, and provides a comparative analysis of twelve representative models across the device-tier spectrum. Chapter 4 surveys adversarial hallucination attacks on LLMs, builds an access-level taxonomy, and compares six widely-used models—three open-source and three closed-source—across attack vectors and benchmark behaviour. Chapter 5 synthesises the two contributions and discusses the joint research gaps they expose.

1.5 Research Objectives

1. To present the foundations of Edge AI, including the edge computing hierarchy and the core compression techniques used at each tier.
2. To survey lightweight CNN architectures (MobileNet, EfficientNet, SqueezeNet, ShuffleNetV2), real-time object detectors (YOLOv8, SSD-MobileNet), TinyML systems (MCUNet, DeepConv-LSTM, TinyBERT), and small language models (LLaMA 3, Phi-3 Mini, Qwen2.5-VL, GLM-4) within a single comparative framework.
3. To analyse model compression techniques—quantization, pruning, knowledge distillation, and MoE routing—in terms of accuracy, memory, and latency.
4. To formulate the threat model for adversarial hallucination attacks against LLMs along three axes: access level, attack goal, and attack vector.
5. To construct a taxonomy of adversarial hallucination attacks distinguishing white-box, black-box, and RAG-layer attacks, mapped to the open-source vs. closed-source distribution model.
6. To compare six representative LLMs (LLaMA 2, Mistral 7B, Falcon 40B, GPT-4, Gemini 1.5 Pro, Claude 3 Opus) across attack surface and adversarial benchmark behaviour.
7. To synthesise the findings of both contributions into a unified view of efficient and trustworthy AI and outline future directions.
8. To experimentally validate the INT8 quantization trade-off (size, latency, accuracy) for a representative edge CNN (MobileNetV2) across three TFLite deployment regimes, and to interpret the results in the context of hardware-algorithm co-design.

1.6 Scope of the Research

The scope is academic survey and comparative analysis based on the two research contributions and the literature they draw upon, supplemented by a focused experimental validation of the quantization trade-off (§3.9). The thesis does not run new adversarial attacks against production LLMs or rerun benchmark suites on dedicated edge hardware; instead, it expands background, taxonomy, and comparative analysis into thesis form and reports one targeted experiment to validate a central claim used throughout Chapter 3. Covered areas include Edge AI architectures across cloud, edge-server, and device/client tiers; lightweight CNNs, object detection, TinyML, small language models and MoE routing; compression and acceleration (quantization, pruning, knowledge distillation, MoE); adversarial hallucination attacks (factual corruption, CoT hijacking, RAG injection, fine-tuning backdoors); hallucination benchmarks (TruthfulQA, HaluEval, FActScore); and comparative analysis of open- and closed-source LLMs. The thesis

does not include source code for attacks, executable exploit instructions, or any material whose purpose is to enable a real adversary.

1.7 Applications and Relevance

Edge AI optimisation is relevant wherever inference must occur close to data: autonomous driving (on-vehicle perception within tens of milliseconds without reliable cloud connectivity), smart manufacturing (anomaly detection on embedded controllers), healthcare wearables (activity recognition and vital-sign analysis under tight energy budgets), agriculture and environmental monitoring (TinyML on solar- or battery-powered sensors), and consumer electronics (on-device assistants that reduce latency, work offline, and preserve user privacy).

Adversarial robustness for LLMs is relevant wherever a generative model influences a decision—in medicine (an adversarial prompt nudging a clinical assistant toward an incorrect dosage), in law and finance (injected documents in a RAG pipeline causing fabricated citations), and in cybersecurity (a backdoored open-source LLM approving insecure patterns under a trigger). The growth of agentic systems and long-context multimodal models broadens this attack surface. The two areas converge as SLMs such as Phi-3 Mini [22] and Qwen2.5-VL [23] are deployed at the edge: a model that fits in 3–6 GB of RAM on a phone or laptop now sits at the intersection of both research questions.

1.8 Thesis Organization

Chapter 2 presents the literature review covering edge AI optimisation, model compression, hallucination characterisation, prompt injection, RAG security, and adversarial attack techniques. Chapter 3 presents the first proposed work, the Edge AI survey, supplemented by a focused experimental validation of the quantization trade-off. Chapter 4 presents the second proposed work, the adversarial hallucination survey. Chapter 5 provides a unified analysis of the two contributions, identifies cross-cutting research gaps, and discusses combined implications. Chapter 6 concludes the thesis and outlines future research directions.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter reviews the literature underpinning the two contributions of this thesis: foundations and progression of Edge AI from the efficiency-oriented CNN architectures of 2016–2019 to the small language model era of 2024–2025; and foundations of LLMs, characterisation of hallucinations, and adversarial attacks that intentionally induce hallucinations through prompting, retrieval poisoning, and fine-tuning. The two streams are linked through research gaps motivating both contributions. To minimise duplication, this chapter restricts itself to positioning prior work and identifying gaps; detailed treatment of architectures, compression techniques, threat models, and comparative analysis appears in Chapters 3 and 4.

2.2 Edge Computing Hierarchy and Foundations

The computing continuum relevant to Edge AI has three tiers [4][24]: the *cloud tier* (data centres, hundreds of GB of GPU memory, tens of ms to seconds of latency); the *edge server tier* (base stations and gateways, tens to hundreds of GB of RAM); and the *device/client tier* (smartphones, single-board computers, IoT devices, and microcontrollers with GB to sub-MB memory and mW–W power budgets). TinyML and on-device AI focus on the device tier; the mismatch between standard architectures (ResNet-50 at ~ 100 MB [6]) and Cortex-M MCU SRAM capacity (hundreds of KB to a few MB) motivates the architectural-efficiency and post-training compression literature.

2.3 Lightweight CNNs, Edge Detection, TinyML and SLMs

The lineage of efficient CNNs runs from MobileNetV1’s depthwise separable convolutions [25] through MobileNetV2’s inverted residuals [26], MobileNetV3’s NAS-driven design [27], EfficientNet’s compound scaling [28][29], SqueezeNet’s fire modules [30], and ShuffleNetV2’s channel-shuffle design [31]; quantitative comparisons are deferred to §3.5. Object detection on resource-constrained hardware combines a lightweight backbone with a single-stage detector head: SSD-MobileNet runs at 30–50 ms per frame on RPi 4 [15]; YOLOv8 [32] represents the current state of the art (YOLOv8-Nano: 37.3 mAP on COCO at 3.2M parameters, 1.9 ms GPU). TinyML targets devices using

<1 mW and 32–512 KB of SRAM; Lin et al. [6] identified peak activation memory as the binding constraint and proposed MCUNet (NAS + TinyEngine, 70.7% top-1 on ImageNet under a 1 MB budget). TinyBERT [33] compresses BERT $7.5\times$ via two-stage distillation retaining 96.8% of GLUE; DistilBERT [34] achieves a similar trade-off in one stage; DeepConv-LSTM [35] reaches 98.24% on UCI-HAR, shrinking from 513 KB to 136 KB on an Arduino Nano 33 BLE Sense. Since 2023, sub-10B SLMs have become viable for the edge: LLaMA 3.1-8B [36] reaches 66.7% MMLU at 4 GB under INT4; Phi-3 Mini [22] reaches 68.8% MMLU at 3–4 GB; Qwen2.5-VL-7B [23] extends to multimodal (71.2% MMLU, 82.3% VQAv2); GLM-4-9B [37] delivers 71.8% HumanEval; federated adaptation is studied in [14].

2.4 Foundations of Model Compression

Four techniques dominate the compression literature relevant to edge deployment: quantization [1], pruning [2], knowledge distillation [3][22], and Mixture-of-Experts routing [24]. Formal definitions and quantitative trade-offs are in §3.3. A key observation revisited in Chapter 5 is that aggressive quantization and pruning reduce robustness to adversarial perturbations [1][2]—a finding that motivates the unified analysis in this thesis.

2.5 LLM Architecture, Alignment, and Hallucination

The modern LLM stack is built on the Transformer [38] and follows a three-stage pipeline: unsupervised pre-training, supervised fine-tuning, and alignment via RLHF [7]. Constitutional AI [8] replaces part of the human-feedback loop with rule-based self-critique. Closed-source models (GPT-4 [39], Gemini 1.5 Pro [20], Claude 3 Opus [40]) follow proprietary variants of this pipeline; open-source models (LLaMA 2 [41], Mistral 7B [42], Falcon [43]) are released as base or RLHF-tuned chat variants.

Ji et al. [9] classify hallucinations as *intrinsic* (output inconsistent with input/context) or *extrinsic* (output introducing information not present in input). Three benchmarks dominate evaluation: TruthfulQA [16] (817 adversarially-designed questions; GPT-3 reaches 58% against a 94% human baseline); HaluEval [17] (35,000 examples; ChatGPT hallucinates at 19.5% baseline); FActScore [18] (atomic-fact verification against a reference). None evaluates models under an active adversary.

2.6 Adversarial Attack Techniques against LLMs

The adversarial attack literature on LLMs spans direct prompt injection [10], broader vulnerability surveys [44], gradient-based white-box attacks such as GCG [11], indirect prompt injection via retrieved documents [12], chain-of-thought manipulation [45][46], retrieval-augmented generation attacks [19][47], and training-data extraction from open-weight models [48]. Detailed treatment of each technique, with attacker-access

requirements and reported success rates, is given in §4.6 of Chapter 4. The key positioning point for this thesis is that these vectors are not equally available to attackers of open- and closed-source models, a distinction that motivates the access-level taxonomy developed in Chapter 4.

2.7 Defenses, Open-Source vs. Closed-Source, and Comparative Review

The defensive literature is comparatively underexplored: post-hoc verifiers such as FactScore [18] are computationally expensive and fail against adversarially fabricated source documents; alignment defences (RLHF [7], Constitutional AI [8]) lower the baseline hallucination rate but do not protect against active attacks [11]; prompt sanitisation mitigates only the simplest injection variants; and knowledge-base authentication and retrieval anomaly detection are emerging but lack systematic evaluation [12]. Open-source models [41][42] expose full weights, enabling gradient-based attacks [11] and fine-tuning backdoors, while closed-source models [20][39] served via inference APIs remove the gradient surface but leave prompt-engineering and RAG-layer attacks open [12]. Table 2.1 summarises representative areas of prior work intersecting the two contributions.

Table 2.1: Comparative overview of selected research directions across efficient and trustworthy AI.

Research tion	Direc-	Main Idea	Strength		Limitation
Lightweight CNNs		Depthwise/factorized convolutions for mobile inference	Drastic and reduction	FLOP parameter	Limited accuracy ceiling vs. large nets
TinyML NAS+engine co-design	with	Joint search of architecture and runtime for MCU SRAM	Fits sub-1 MB hardware		Domain-limited; mostly vision
Object detection at the edge		Lightweight backbone + single-stage detector	Real-time on RPi/Jetson/Coral		Lower mAP than full detectors on complex scenes
Knowledge distillation		Student imitates teacher distribution	Strong accuracy per parameter		Teacher quality bounds student
Quantization (INT8, INT4)		Reduce precision of weights/activations	4–8× memory reduction		Robustness loss under attack
Small language models		Sub-10B parameter LLMs for edge GPUs/laptops	On-device language understanding		Still require GBs of RAM
MoE routing		Sparse activation over expert sub-networks	Latency-constant scaling		Routing overhead and load balance

Research tion	Direc-	Main Idea	Strength	Limitation
Hallucination benchmarks		Measure passive factual reliability	Standardized comparison	Not adversarial
GCG / gradient attacks		Weight-level adversarial suffix search	High ASR on open-source LLMs	Requires white-box access
Indirect prompt injection (RAG)		Inject instructions through retrieved documents	Equally effective on closed-source LLMs	Hard to filter without losing utility
RLHF & Constitutional AI		Align outputs to human or rule preferences	Reduce baseline hallucination	Not robust under active attack

2.8 Research Gaps and Summary

The reviewed literature reveals several gaps motivating the two contributions: (i) no single architecture or compression technique satisfies the full spectrum of edge scenarios; (ii) NAS objectives targeting peak activation memory rather than FLOPs are missing [6]; (iii) portability of quantized models across heterogeneous edge backends is brittle; (iv) energy-per-inference benchmarks across hardware tiers are not standardised; (v) no widely-used benchmark evaluates LLMs under adversarial hallucination conditions [16][17][18]; (vi) RAG security remains underexplored [12]; (vii) long-context and multimodal attack surfaces are largely uncharacterised; and (viii) most importantly, compressed and quantized models—the models deployed at the edge—are less robust to adversarial attacks than full-precision models [1][2], an intersection neither pillar fully addresses on its own. The next chapter presents the first proposed work in detail.

CHAPTER 3

PROPOSED WORK PART I: EDGE AI MODELS, OPTIMIZATION TECHNIQUES, AND DEPLOYMENT STRATEGIES

3.1 Chapter Introduction and Motivation

This chapter presents a comparative survey of Edge AI models, optimisation techniques, and deployment strategies covering 2015–2025. Cloud-only deployment is unacceptable in latency- or privacy-critical applications such as autonomous driving, medical monitoring, and industrial fault detection [4][5]; Edge AI moves computation closer to the data source, spanning microcontrollers with kilobytes of RAM through smartphones with gigabytes to edge servers with tens of gigabytes. The chapter is organised around four model families, four compression techniques, a comparison of twelve representative models, the trade-offs that emerge, and a focused experimental validation of the INT8 quantization trade-off (§3.9).

3.2 System Overview: Edge AI Taxonomy

Figure 3.1 shows the taxonomy of model families, organised by architecture type and target hardware tier. Four families are identified: lightweight CNNs, object detection networks, small language models, and TinyML / MCU models, with power budgets spanning <1 mW for microcontrollers to tens of watts for edge servers.

3.3 Foundations of Model Compression

Three principal compression techniques and one sparse-activation routing strategy bridge the gap between full-precision deep networks and edge hardware. A standard ResNet-50 requires ~ 100 MB, exceeding Cortex-M SRAM (256 KB–2 MB) by two to three orders of magnitude [6].

3.3.1 Quantization

Quantization reduces precision from FP32 to INT8 or INT4, producing $4\text{--}8\times$ smaller models with little accuracy loss [1]. For a real-valued weight w , INT8 quantization

Fig. 1. Taxonomy of Edge AI Model Families by Architecture and Target Hardware Tier

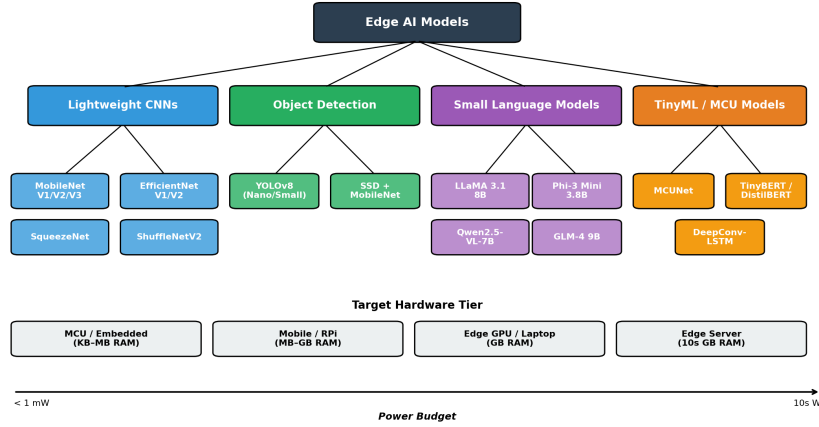


Figure 3.1: Taxonomy of Edge AI model families covered in this survey, organised by architecture type and target hardware tier.

with scale s and zero point z is $q(w) = \text{round}(w/s) + z$, with $q(w) \in \{-128, \dots, 127\}$; INT4 maps weights into 16 discrete levels. Aggressive precision reduction also reduces robustness to adversarial perturbations—an issue revisited in Chapter 5.

3.3.2 Pruning

Pruning removes near-zero connections, yielding 20–40% smaller models retaining over 95% accuracy [2]. Structured pruning (channels, filters, layers) is preferred for edge deployment because it yields actual computational savings on dense matrix-multiplication hardware.

3.3.3 Knowledge Distillation

Knowledge distillation [3] trains a smaller *student* to mimic the output distribution of a larger *teacher*. The distillation loss combines cross-entropy with a soft-target term:

$$\mathcal{L}_{KD} = (1 - \alpha) \text{CE}(y, p_S) + \alpha T^2 \text{KL}\left(p_T^{(T)} \parallel p_S^{(T)}\right), \quad (3.1)$$

where p_S and p_T are student and teacher distributions, T is the distillation temperature, and α balances the terms. Phi-3 Mini [22] uses this approach with a curated “textbook quality” dataset.

3.3.4 Mixture of Experts (MoE)

MoE is a sparse-activation routing strategy [24] in which a gating function routes each token to a small subset of expert sub-networks: $y = \sum_{i \in \text{TopK}(g(x))} g_i(x) \cdot E_i(x)$. Only the

top- K experts are active, allowing latency to remain roughly constant as total parameters grow—valuable for edge-GPU deployment of LLMs.

3.4 Literature Search Methodology

Literature was collected from IEEE Xplore, arXiv, the ACM Digital Library, Springer, and Google Scholar (terms such as “Edge AI”, “TinyML”, “on-device inference”, “lightweight neural network”, “model compression”, “small language model deployment”; window January 2015–March 2025). From an initial pool of over 180 candidates, 27 primary articles meeting three criteria—edge-deployment focus, reputable venue, and quantitative results on a standard benchmark—formed the basis of this contribution. **Inclusion criteria:** studies addressing edge-hardware deployment (MCU, mobile, edge GPU, or edge server), peer-reviewed or highly-cited preprints, and quantitative results on at least one standard benchmark (ImageNet, COCO, GLUE, MMLU, HumanEval, VQAv2, UCI-HAR). **Exclusion criteria:** cloud-only focus, no quantitative result, or duplication of earlier included work. The selection workflow is summarised in Figure 3.2.

Literature search and screening flow (Edge AI survey)

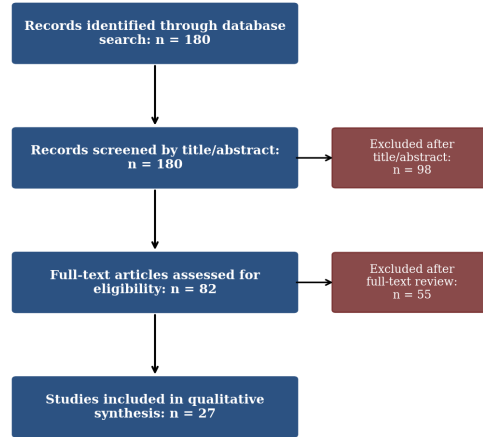


Figure 3.2: PRISMA-style flow diagram for the Edge AI literature search.

3.5 Lightweight CNN Architectures

Howard et al. [25] introduced depthwise separable convolutions in MobileNetV1, factorising a convolution of cost $D_k^2 \cdot M \cdot N \cdot D_F^2$ into a depthwise step of cost $D_k^2 \cdot M \cdot D_F^2$ and a pointwise step of cost $M \cdot N \cdot D_F^2$, giving a reduction of $1/N + 1/D_k^2$, or $8\text{--}9\times$

for $D_k = 3$. MobileNetV2 [26] added inverted residuals; MobileNetV3 [27] combined NAS with hard-swish, reaching 67.4% top-1 (MobileNetV3-Small) on ImageNet with 2.5M parameters. Tan and Le [28] introduced compound scaling—jointly scaling depth $d = \alpha^\phi$, width $w = \beta^\phi$, and resolution $r = \gamma^\phi$ subject to $\alpha\beta^2\gamma^2 \approx 2$ —with EfficientNetV2-S reaching 83.9% top-1 at 84 MB [29]. SqueezeNet [30] introduced fire modules (squeeze of 1×1 filters then mixed $1 \times 1/3 \times 3$ expand), reaching AlexNet-level accuracy at ~ 0.7 MB ($50\times$ reduction). ShuffleNetV2 [31] introduced channel split and channel shuffle, ranking among the fastest models at any given accuracy on mobile CPUs.

3.6 Object Detection, TinyML, and Small Language Models

For object detection, SSD with a MobileNet backbone enables real-time detection on Raspberry Pi (30–50 ms per frame on RPi 4) [15]; YOLOv8 [32] extends the YOLO family with an anchor-free head, C2f backbone, and SiLU activations, with YOLOv8-Nano reaching 37.3 mAP on COCO at 3.2M parameters and 1.9 ms GPU latency. CPU-only devices benefit from accelerators such as the Intel Neural Compute Stick and Google Coral Edge TPU. For TinyML, MCUNet [6] co-designs NAS with the TinyEngine inference runtime, reaching 70.7% top-1 on ImageNet under a 1 MB SRAM budget—a $12\times$ reduction relative to MobileNetV2, with peak activation memory (not parameter count) as the binding constraint. For edge NLP, TinyBERT [33] compresses BERT $7.5\times$ smaller and $9.4\times$ faster via two-stage distillation retaining 96.8% of GLUE; DistilBERT [34] achieves a similar trade-off in one stage. For HAR, DeepConvLSTM [35] reaches 98.24% on UCI-HAR with F1 of 98.23% on an Arduino Nano 33 BLE Sense; full-integer quantization shrinks the model from 513 KB to 136 KB.

For small language models, sub-10B models have become viable at the edge since 2023. LLaMA 3.1-8B [36] reaches 66.7% MMLU and 62.0% HumanEval at 4 GB under INT4. Phi-3 Mini [22] (3.8B) is trained via “textbook quality” data curation and distillation from GPT-4, reaching 68.8% MMLU and 58.1% HumanEval at 3–4 GB—the strongest model in its parameter class. Qwen2.5-VL-7B [23] extends to multimodal input (71.2% MMLU, 82.3% VQAv2). GLM-4-9B [37] achieves 71.8% HumanEval with function-calling support. Jiang et al. [14] study personalised wireless federated learning for SLMs.

3.7 Comparative Analysis: Twelve Representative Models

Table 3.1 compares twelve representative Edge AI models. Performance metrics are from the primary studies cited at the quantization level in each row.

Table 3.1: Comparative analysis of Edge AI models. Performance metrics are reported on the benchmark column appropriate to each model family and are *not* directly comparable across rows from different families (ImageNet top-1, COCO mAP, GLUE retention, and language-model benchmarks measure different things). The table is intended for within-family and cross-family deployment-tier comparison. Min RAM is the deployment memory at the indicated quantization level. † denotes INT4 deployment requirement, ‡ denotes FP16 deployment requirement.

Model	Year	Params	Dataset	Acc./mAP	Min RAM	Quant.	Target HW	Notes
MobileNetV3-S [27]	2019	2.5M	ImageNet	67.4%	1.5 MB	FP32	MCU / Mobile	Lightweight; fast CPU
SqueezeNet [30]	2016	1.2M	ImageNet	57.5%	0.7 MB	FP32	Embedded	50× smaller than AlexNet
EfficientNetV2-S [28]	2021	21M	ImageNet	83.9%	84 MB	FP32	Mobile GPU	Compound scaling
ShuffleNetV2 [31]	2018	2.3M	ImageNet	69.4%	5 MB	FP32	Mobile CPU	Low memory access
YOLOv8-Nano [15][32]	2023	3.2M	COCO	37.3 mAP	8 MB	FP32	Jetson / RPi	Anchor-free
SSD-MobileNetV1 [15]	2017	6.8M	COCO	19.0 mAP	20 MB	INT8	RPi / MCU	Energy-efficient
MCUNet [6]	2021	0.7M	ImageNet	70.7%	1 MB	INT8	Cortex-M	NAS + engine co-design
TinyBERT [33]	2020	14.5M	GLUE	96.8% of BERT	60 MB	FP32	Mobile	7.5× smaller than BERT
DeepConv-LSTM [35]	2024	0.13M	UCI-HAR	98.24%	0.5 MB	INT8	Arduino Nano	Temporal + spatial
LLaMA 3.1-8B [36]	2024	8B	MMLU / HumanEval	66.7% / 62.0%	4 GB [†]	INT4	Edge server	Open-source
Phi-3 Mini [22]	2024	3.8B	MMLU / HumanEval	68.8% / 58.1%	3 GB [†]	INT4	Laptop / Phone	Best acc./param <4B
Qwen2.5-VL-7B [23]	2024	7B	MMLU / VQAv2	71.2% / 82.3%	6 GB [‡]	FP16	Edge GPU	Multimodal

3.8 Accuracy vs. Model Size Trade-off and Discussion

Figure 3.3 shows the trade-off between ImageNet top-1 accuracy and model size (MB, log scale): SqueezeNet dominates the size-efficient corner at 57.5%, EfficientNetV2-S reaches 83.9% at 84 MB, and MCUNet sits above the conventional-CNN trend line at 70.7% under a 1 MB constraint via hardware-aware co-design.

Traditional CNNs vs. transformer-based SLMs: the families are operationally complementary—MCUNet operates inside a 1 MB budget while Phi-3 Mini requires at least 3 GB, roughly three orders of magnitude apart [6][22]. **Inference speed vs. accuracy in object detection:** SSD-MobileNetV1 is most energy-efficient on CPU but limited to 19.0 mAP on COCO; YOLOv8-Nano reaches 37.3 mAP at 1.9 ms on GPU, suitable for autonomous-vehicle and surveillance applications, with accelerators such as Intel NCS and Google Coral Edge TPU helpful on CPU-only devices [15]. **SLM benchmark comparison:** Qwen2.5-VL-7B has the highest MMLU (71.2%) and the only native multimodal support; GLM-4-9B leads in coding (71.8% HumanEval);

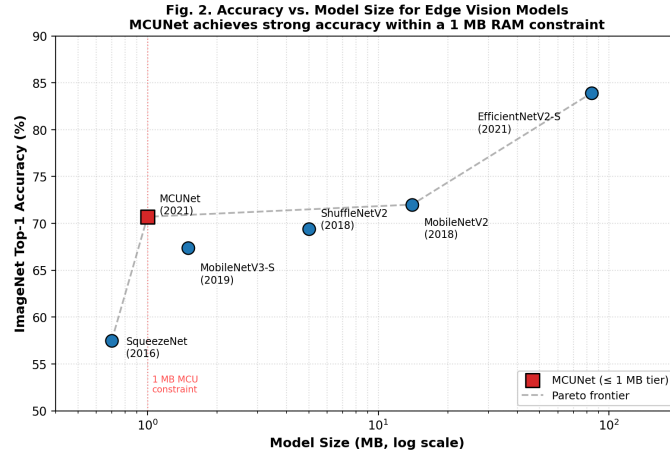


Figure 3.3: Accuracy vs. model size for edge vision models. MCUNet achieves strong accuracy within a 1 MB RAM constraint via hardware-aware co-design, outperforming same-size conventional CNNs.

Phi-3 Mini has the lowest memory requirement (3 GB at INT4). **Scalability via MoE:** MoE-based SLMs scale better than dense SLMs because only a subset of experts is active per token, so latency stays roughly constant as total parameter count grows [24], whereas dense CNNs scale linearly with depth or width.

3.9 Experimental Validation: Quantization Trade-off

The literature reviewed in §3.3 and the comparative analysis in Table 3.1 cite published claims that INT8 quantization yields roughly $4\times$ size compression with under 1% accuracy degradation [1]. This section reports a focused reproduction study on MobileNetV2 [26] evaluated under three TFLite deployment regimes on the same trained checkpoint.

3.9.1 Experimental Setup

The pretrained MobileNetV2 (width multiplier $\alpha = 1.0$, 224×224 input, ImageNet-1000 weights, 3.54M parameters) is converted to TFLite in three quantization regimes: (a) **FP32**—32-bit floating-point baseline; (b) **INT8 dynamic-range**—weights quantized to INT8 offline, activations dequantized at runtime; and (c) **INT8 full-integer**—weights and activations quantized to INT8 using a 200-sample calibration set, the deployment target for microcontroller-class NPUs and DSPs. Top-1 classification accuracy is evaluated on the Imagenette validation split (3,925 images across 10 ImageNet classes), using the standard imagenette-to-ImageNet label mapping. Inference latency is measured on a single CPU thread (Google Colab x86, no GPU) over 100 runs after a 20-run warm-up, simulating a general-purpose edge CPU without dedicated INT8 acceleration. Model sizes are reported as on-disk TFLite file sizes.

3.9.2 Results

Table 3.2 reports the measured size, latency, and accuracy for each regime, together with compression ratio, latency speed-up, and accuracy drop relative to the FP32 baseline. Figure 3.4 plots the size-versus-accuracy trade-off on a log-scaled axis.

Table 3.2: Experimental quantization trade-off for MobileNetV2 on the Imagenette validation set. Latency: single-thread CPU, mean of 100 runs.

Variant	Size (MB)	Compr. ratio	Latency (ms)	Speed-up	Top-1 (%)	Δ Acc. (pp)
FP32 (baseline)	13.98	1.00×	12.41	1.00×	83.97	—
INT8 dynamic-range	3.80	3.68×	23.43	0.53×	83.06	−0.91
INT8 full-integer	4.01	3.49×	24.53	0.51×	82.70	−1.27

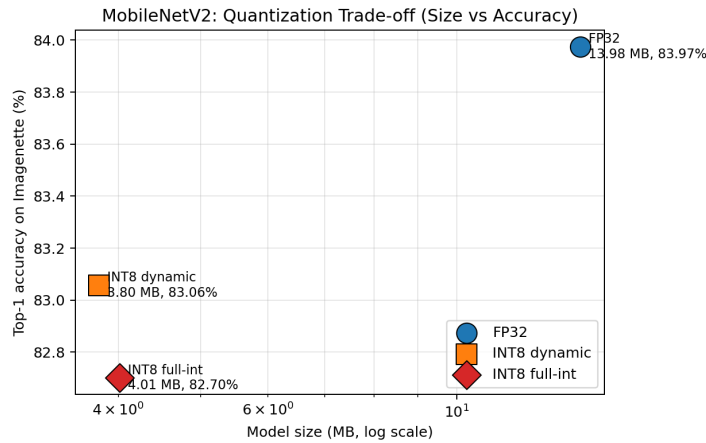


Figure 3.4: Empirical quantization trade-off for MobileNetV2 on Imagenette. INT8 quantization reduces on-disk size by approximately $3.5\times$ with an accuracy degradation of only 0.91–1.27 percentage points relative to the FP32 baseline.

3.9.3 Discussion

The experimental results align closely with the published claim of approximately $4\times$ compression at $\leq 1\%$ accuracy loss for INT8 quantization [1]. The full-integer variant—the target for true edge-NPU deployment—compresses the model from 13.98 MB to 4.01 MB ($3.49\times$ smaller) while retaining 98.5% of the baseline accuracy (82.70% vs. 83.97%, a drop of 1.27 pp). This empirically validates the *size and accuracy* components of the quantization claim and supports the deployment scenarios catalogued in Table 3.1.

Three observations stand out. **(i) Size compression is essentially insensitive to the INT8 variant choice:** the dynamic-range (3.80 MB) and full-integer (4.01 MB) variants

differ by under 6% because most of MobileNetV2’s parameter mass is in depthwise-separable convolution weights, captured in both variants. **(ii) Compression does not automatically imply lower latency on general-purpose CPU:** both INT8 variants are *slower* than the FP32 baseline on Colab x86—dynamic-range at $0.53\times$ throughput (23.43 ms vs. 12.41 ms) and full-integer at $0.51\times$ (24.53 ms). On x86 CPUs without dedicated INT8 SIMD (AVX-512 VNNI), TFLite’s INT8 kernels do not outperform its highly-optimised FP32 XNNPACK kernels because the INT8 kernels were principally tuned for ARM Cortex-M and Cortex-A devices [6]. The size benefit of quantization is hardware-independent; the latency benefit is not. **(iii) The latency picture inverts on dedicated edge hardware:** ARM CMSIS-NN on Cortex-M, Hexagon HVX on Qualcomm DSPs, and the Coral Edge TPU consistently show $4\text{--}10\times$ INT8 speed-ups because they execute INT8 multiply-accumulate natively. The CPU result reported here is therefore not a refutation of the quantization latency claim but rather empirical demonstration of why hardware-algorithm co-design is essential: deploying a quantized model on the wrong hardware delivers all of the accuracy cost and none of the latency benefit.

This experiment reports baseline behaviour under benign inputs only. The robustness of the INT8 variant under *adversarial* inputs—the central question of the second contribution of this thesis—is the principal cross-cutting research gap identified in Chapter 5, where compressed and quantized models are flagged as the very models deployed at the edge yet the least-studied under active adversaries.

3.10 Challenges, Research Gaps and Future Directions

The principal TinyML bottleneck is peak activation memory rather than parameter count [6], yet most NAS methods optimise for FLOPs and produce networks not deployable on MCUs. Quantized models port fragily across TensorFlow Lite, Core ML, and ONNX Runtime [13]; on-device training is hard—sparse continual learning [49] and split learning offer partial solutions and federated learning faces communication and non-IID problems [14]. Most edge AI is unimodal; multimodal and agentic frameworks at sub-1 W remain emerging [50], and energy-per-inference is rarely reported in standardised form [15][51]. Edge deployment also carries security and privacy concerns: quantized and pruned models are less robust to adversarial attacks [1][2]; on-device inference exposes the model to extraction and side-channel attacks; federated learning is subject to data-poisoning and Byzantine attacks [14]; and enclave execution adds overhead difficult to absorb on power-constrained devices [13]—threads that connect to Chapter 5. Future directions include extending hardware-algorithm co-design from CNNs to transformer-based SLMs [6]; extreme quantization (INT2, binary nets) and structural sparsity [1][2]; on-device adaptation via LoRA-style adapters in federated settings [14]; neuromorphic and in-memory computing on the 2027–2030 horizon; and a standardised tiered benchmark measuring accuracy, latency, energy, and peak memory across hardware classes [15][51].

3.11 Chapter Summary

This chapter surveyed Edge AI models, optimisation techniques, and deployment strategies—the edge computing hierarchy, four model families, four compression and routing techniques, and twelve representative models compared on a Pareto-frontier accuracy-vs-size plot. The survey was supplemented by a focused experimental study (§3.9) measuring size, latency, and accuracy of MobileNetV2 under three TFLite quantization regimes, empirically confirming the size-and-accuracy components of the published quantization claim and demonstrating that the latency component is hardware-dependent.

CHAPTER 4

PROPOSED WORK PART II: ADVERSARIAL HALLUCINATION ATTACKS ON LARGE LANGUAGE MODELS

4.1 Chapter Introduction and Motivation

This chapter presents the second contribution: a systematic survey of adversarial hallucination attacks against LLMs, with explicit comparison of open- and closed-source models. As LLMs have moved from research artefacts to deployed systems in medicine, law, cybersecurity, and consumer assistance [20][39][41], hallucination—plausible-sounding but unsupported content [9]—has shifted from a reliability issue to a security one, as it can be deliberately induced by an adversary [10][11][12]. The chapter develops a taxonomy organised around attacker access (white-box, black-box, RAG-layer), reviews benchmarks, surveys defences, and compares six representative models (LLaMA 2, Mistral 7B, Falcon 40B, GPT-4, Gemini 1.5 Pro, Claude 3 Opus). The open vs. closed-source distinction is central: open weights enable gradient-based attacks and fine-tuning backdoors; closed-source APIs leave prompt-engineering and RAG-layer attacks fully open.

4.2 Background: LLM Architecture and Hallucination Taxonomy

Modern LLMs follow the Transformer [38] with a three-stage pipeline: unsupervised pre-training, supervised fine-tuning (SFT), and alignment via RLHF [7]; Constitutional AI [8] replaces part of the human-feedback loop with rule-based self-critique. Alignment is what makes chat models follow instructions, refuse unsafe requests, and remain factual in benign conditions—and what an adversary attempts to bypass. Following Ji et al. [9], hallucinations fall into three categories: *intrinsic* (output does not align with input or in-context document), *extrinsic* (output introduces false statements not supported by the input), and *adversarial* (intentional induction of either through attacker-controlled inputs)—the focus of this contribution.

4.3 Threat Model

The threat model has three axes, illustrated in Figure 4.1. **Access level:** *white-box* (weights, gradients, architecture; open-source only); *grey-box* (partial access, e.g., transfer from a similar open-source model); *black-box* (query API only; either class). **Attack goal:** *targeted* (specific false content) vs. *untargeted* (degrade overall reliability). **Attack vector:** prompt engineering, RAG knowledge-base poisoning (direct and indirect injection), fine-tuning-based backdoors, and chain-of-thought hijacking. Open weights for LLaMA 2 [41], Mistral 7B [42], and Falcon [43] enable gradient-based adversarial prompt search [11] and fine-tuning backdoor injection. Closed-source models (GPT-4 [39], Gemini 1.5 Pro [20], Claude 3 Opus [40]) expose only an inference API, removing the gradient surface but leaving prompt-engineering and RAG-layer attacks open [12]. Black-box access is necessary but not sufficient for adversarial robustness.

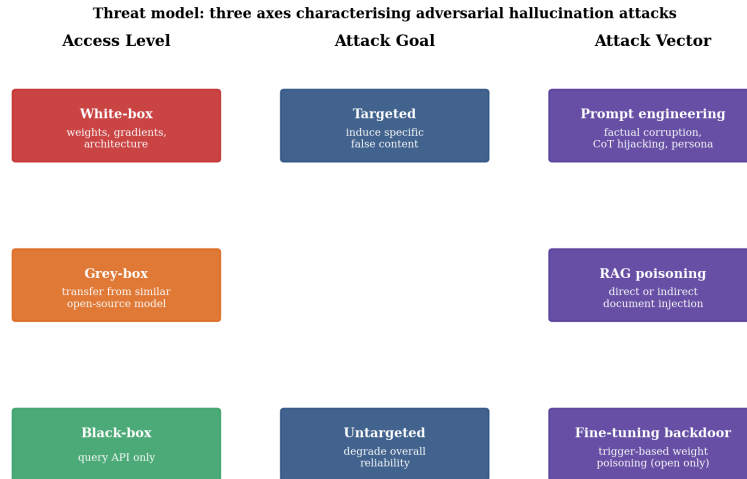


Figure 4.1: Threat model: an adversarial hallucination attack is characterised by the tuple (access level, attack goal, attack vector). White-box attacks require open weights; RAG-layer attacks apply to both open- and closed-source classes.

4.4 System Overview: Adversarial Hallucination Attack Taxonomy

Figure 4.2 shows the taxonomy. Attacks are grouped by attacker access: white-box (red) apply only to open-source models; black-box (green) and RAG-layer (purple) apply to both classes.

4.5 Literature Search Methodology

Following Kitchenham–Charters guidelines [52], databases queried were IEEE Xplore, ACM Digital Library, arXiv (cs.CL, cs.CR, cs.AI), Semantic Scholar, and Google

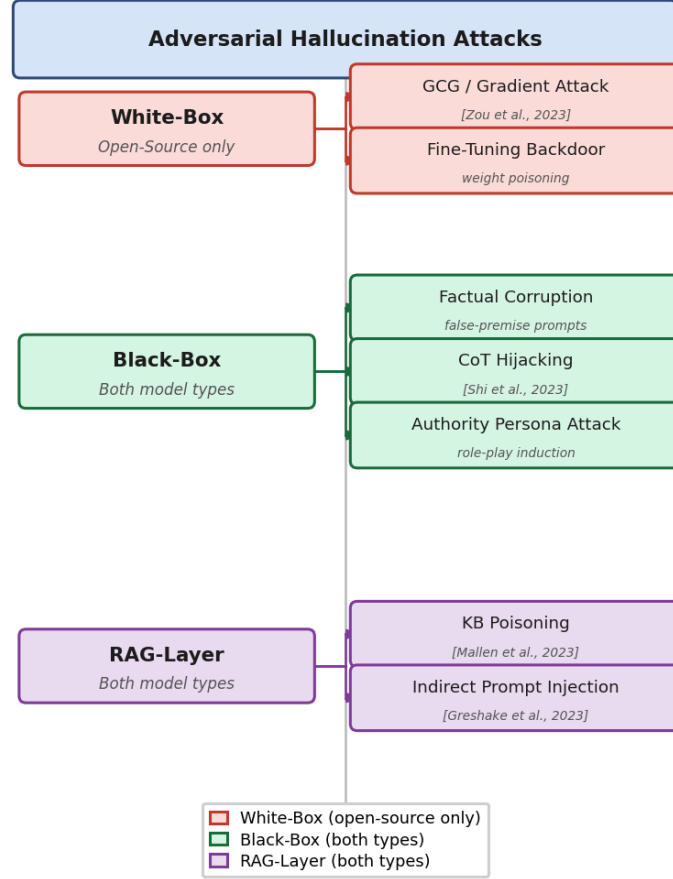
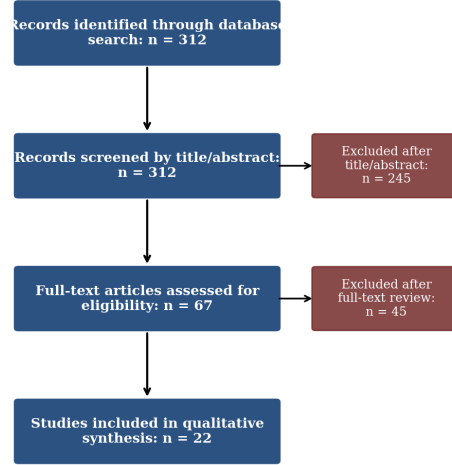


Figure 4.2: Adversarial hallucination attack taxonomy. “White-box” (red) attacks demand access to the weights and apply only to open-source models. Black-box (green) and RAG-layer (purple) attacks impact both classes equally.

Scholar. The base query was (“LLM” OR “large language model”) AND (“hallucination” OR “adversarial attack” OR “prompt injection”) AND (“security” OR “vulnerability” OR “robustness”). **Inclusion criteria:** publication window January 2017–March 2025; peer-reviewed venue or highly-cited arXiv preprint (citations ≥ 50); direct evaluation of an adversarial attack or hallucination phenomenon on a text-based LLM; reproducible methodology with sufficient detail to assess attack-success rates. **Exclusion criteria:** surveys-of-surveys without primary contribution; non-text modalities only; qualitative observations without quantitative metrics; duplicate/superseded preprints. The search yielded 312 candidates; 67 were reviewed in detail, 22 met the inclusion criteria, and 18 were highly relevant. The selection workflow is in Figure 4.3.

Literature search and screening flow (Adversarial Hallucination survey)

**Figure 4.3:** PRISMA-style flow diagram for the adversarial hallucination literature search.

4.6 Hallucination Benchmarks, Attack Techniques and Defenses

Ji et al. [9] classify hallucinations by origin (training data, architecture, decoding) but do not address adversarial induction; Weidinger et al. [21] adopt a socio-centric view of misinformation risk. HaluEval [17] provides 35,000 QA/dialogue/summarisation examples (ChatGPT hallucinates at 19.5% baseline); FActScore [18] verifies atomic facts but presumes a benign reference; TruthfulQA [16] contains 817 adversarial questions (GPT-3 at 58% vs. 94% human baseline) but measures imitation of common misconceptions, not active truth manipulation; Carlini et al. [48] show GPT-2 reveals verbatim training data, an additional exfiltration surface.

Attack techniques. *Prompt injection* [10][44]: natural-language instructions override system instructions and safety constraints. *Gradient-based white-box (GCG)* [11]: gradient-guided token substitution on open-source models constructs adversarial suffixes (success >80% on open-source, 20–40% transfer to closed-source); applicability to inducing *factual* hallucinations remains open. *Indirect prompt injection* [12]: adversarial instructions placed inside RAG-retrieved documents succeed against both classes because retrieval is decoupled from alignment. *Chain-of-thought hijacking* [45][46]: LLMs are highly susceptible to distractions in reasoning chains, letting an attacker manipulate the start and influence the final answer. *RAG knowledge-base poisoning* [19][47]: LLMs cannot reliably distinguish valid from false retrieved context, enabling targeted hallucinations when the adversary controls the knowledge base. *Fine-tuning backdoor*: weight-level fine-tuning inserts a backdoor trigger producing attacker-controlled outputs; combined with public release on model hubs, this is a

supply-chain risk.

Defense literature. Post-hoc verifiers such as FActScore [18] identify hallucinated outputs but fail against adversarially generated source documents and are computationally expensive. Alignment methods (RLHF [7], Constitutional AI [8]) lower baseline hallucination rates but provide no protection under active attack [11]. Prompt sanitisers filter the most obvious injection variants [10] but do not generalise. Knowledge-base authentication and retrieval anomaly detection are proposed but lack systematic evaluation [12]. The absence of a unified, evaluated defence framework is the principal gap.

4.7 Comparative Analysis: Six Representative Models

4.7.1 Model Overview and Configuration

Table 4.1 compares the six models, drawn from the open-source ecosystem (LLaMA 2 [41], Mistral 7B [42], Falcon 40B [43]) and the closed-source commercial landscape (GPT-4 [39], Gemini 1.5 Pro [20], Claude 3 Opus [40]). TruthfulQA scores for open-source models are from the Open LLM Leaderboard [53]; the GPT-4 score is from the GPT-4 Technical Report [39]. Alignment level is a qualitative assessment based on each model’s public technical report.

Table 4.1: Comparative analysis of LLMs under adversarial hallucination study. [†]Parameter count not publicly disclosed. [‡]GPT-4 base-model score; RLHF post-training yields substantially higher scores per the same report. ^{*}These models post-date the original TruthfulQA benchmark; cross-benchmark comparison would be misleading.

Model	Year	Type	Params	Eval. Dataset	Truthful-QA	Align.	Advantages / Limits
LLaMA 2-70B [41]	2023	Open	70B	TruthfulQA, HaluEval [53]	57.4% [53]	Medium	Full weights; RLHF; large community. Vulnerable to FT backdoors.
Mistral 7B [42]	2023	Open	7B	TruthfulQA [53]	42.2% [53]	Low	Efficient; sliding-window attention. Minimal alignment.
Falcon 40B [43]	2023	Open	40B	TruthfulQA [53]	45.1% [53]	None	Multilingual; permissive license. No RLHF hardening.
GPT-4 [39]	2023	Closed	[†]	TruthfulQA [16]	59.0% [‡] [39]	High	Best alignment; moderation API. Opaque; CoT hijacking risk.
Gemini 1.5 Pro [20]	2024	Closed	[†]	MMLU, HaluEval [20]	N/A [*]	High	1M-token context; multimodal. Long context widens attack surface.
Claude 3 Opus [40]	2024	Closed	[†]	Internal evals [40]	N/A [*]	High	Constitutional AI; persona resistance. RAG injection risk.

4.7.2 Attack Surface Comparison

Table 4.2 summarises the attack surface between open- and closed-source models. Open-source models are exposed to white-box, grey-box, and black-box attacks. Closed-

source models are exposed primarily to black-box and RAG-layer attacks.

To make the assessment reproducible rather than purely qualitative, each row in Table 4.2 is graded against the following rubric, derived from attack-success rates (ASR) reported in the cited primary studies: **High (H)**—attack succeeds at $\text{ASR} \geq 70\%$ (or its prerequisite, e.g., weight access, is trivially available); **Medium (M)**—attack succeeds at $30\% \leq \text{ASR} < 70\%$, or at high rates only under specific configurations; **Low (L)**—attack succeeds at $\text{ASR} < 30\%$, typically reflecting attenuated transfer from a white-box setting; **N/A**—the prerequisite (e.g., gradient or weight access) is structurally absent for that model class. Two caveats apply: ASR thresholds are coarse-grained with underlying numbers drawn from heterogeneous studies, so the table is intended for relative comparison; and alignment investment is a confounder for the closed-source column—a “Medium” rating reflects published evidence that an attack remains effective in 30–70% of attempts after alignment, not that the attack is intrinsically harder to construct.

Table 4.2: Attack surface comparison: open-source vs. closed-source LLMs. Risk levels: H = High, M = Medium, L = Low, N/A = Not Applicable.

Attack Vector	Open	Closed	Reference
Factual corruption	H	M	[10]
CoT hijacking	H	M	[46]
RAG injection	H	H	[12]
Fine-tuning backdoor	H	N/A	[11]
Gradient-based (GCG)	H	L	[11]
Authority persona	H	M	[44]
Weight extraction	H	N/A	[48]
Prompt leakage	M	M	[10]

4.7.3 Performance Under Adversarial Pressure and Approaches Compared

On TruthfulQA [16], the Open LLM Leaderboard [53] reports: Mistral 7B 42.2%, Falcon 40B 45.1%, LLaMA 2-70B 57.4%; base GPT-4 reaches $\sim 59\%$, substantially higher after RLHF [39]. Standardised public TruthfulQA results for Gemini 1.5 Pro and Claude 3 Opus are not available; readers are referred to the respective reports [20][40]. For GCG-style suffixes, Zou et al. [11] report alignment bypass on open-source models is effectively certain, while black-box transfer to closed-source ranges 20–40%. For RAG injection, Greshake et al. [12] and Mallen et al. [19] demonstrate both classes achieve high attack success because the retrieval layer is decoupled from model alignment. Early approaches that aimed for LLM truthfulness through retrieval augmentation and post-hoc verification [47] achieve markedly higher hallucination rates than contemporary RLHF and Constitutional AI alignment [7][8]; however, none are robust to targeted adversarial attacks [11][12]. Securing LLMs against adversarial

attacks requires a threat-modelled approach beyond the reliability-oriented framing of conventional alignment, as implied by Weidinger et al. [21].

4.8 Challenges, Discussion, and Future Directions

4.8.1 Challenges and Research Gaps

Four gaps stand out. (i) *Absence of adversarial hallucination benchmarks*—TruthfulQA, HaluEval, FActScore [16][17][18] measure passive hallucination only. (ii) *Security of RAG pipelines*—widely deployed but underexplored in adversarial scenarios [12][47]; promising candidates include knowledge-base authentication, retrieval anomaly detection, and document sandboxing. (iii) *Long-context and multimodal attack surfaces*—Gemini 1.5 Pro’s 1M-token context allows an adversarial premise injected early to influence reasoning across hundreds of thousands of tokens, and vision-language models face image-text combined attacks only beginning to be characterised. (iv) *Transferability and evaluation standardisation*—a full transferability matrix has not been constructed and different groups evaluate different models on different benchmarks.

4.8.2 Discussion and Future Directions

Open- and closed-source LLMs are not uniformly comparable on the safety axis: they differ in *which* attack surfaces they expose. Open-source models exhibit a strictly larger attack surface in the white-box dimension because weights are public, enabling gradient-based attacks [11] and fine-tuning backdoors not available against closed-source models. In the black-box dimension, alignment-trained closed-source models exhibit moderately stronger resistance to prompt-engineering and persona attacks owing to investment in RLHF and Constitutional AI. At the RAG layer, however, the two classes are essentially equivalent [12][19]: retrieval is architecturally decoupled from alignment, so stronger alignment does not propagate to retrieved content. This is the central finding of the contribution: closed-source release does not buy safety against retrieval-layer attacks. Current alignment techniques [7][8] were designed for helpfulness and harmlessness in benign settings, not for active attackers; future alignment research must treat adversarial truthfulness as a first-class objective. The contribution focuses on text LLMs evaluated on English-language adversarial cases; multimodal, code-generation, and non-English settings are out of scope, and LLMs being dynamic, specific vulnerabilities may have been patched in newer versions.

The most important future activity is an *adversarial hallucination benchmark* reflecting realistic attacker behaviour across white-box and black-box vectors. *Secure RAG* should include document-level cryptographic authentication, semantic anomaly detection on retrieved fragments, and sandboxing of untrusted sources. *Alignment-as-security* should treat adversarial truthfulness as a first-class objective alongside helpfulness and harmlessness [7][8], with calibrated factual confidence as an intrinsic detector and automated red-teaming accelerating discovery of new vectors.

4.9 Chapter Summary

This chapter surveyed adversarial hallucination attacks on LLMs and compared open- and closed-source models. Three conclusions: (i) open-source models present a wider attack surface because weights are accessible; (ii) closed-source models, despite stronger alignment, remain equally vulnerable at the RAG layer; (iii) current hallucination benchmarks measure passive reliability rather than robustness to active attack.

CHAPTER 5

RESULTS ANALYSIS AND DISCUSSION

5.1 Introduction

This chapter integrates the two contributions. Chapter 3 and Chapter 4 describe two halves of the same deployment story: the engineering required to make models small enough to run at the edge, and the engineering required to make generative models trustworthy under active adversaries. The chapter analyses each contribution’s results, presents a unified interpretation, identifies cross-cutting research gaps, and discusses practical implications.

5.2 Analysis of the Two Surveys

5.2.1 Edge AI Survey Results

Among lightweight CNNs, the four surveyed models span the size–accuracy Pareto frontier: EfficientNetV2-S (83.9%, 21M params) leads on accuracy but is not MCU-deployable; SqueezeNet (57.5%, 1.2M) is smallest but well below modern accuracy; ShuffleNetV2 (69.4%, 2.3M) has the lowest memory-access cost on CPU; MobileNetV3-S (67.4%, 2.5M) balances ImageNet accuracy with mobile-CPU deployability. For object detection, YOLOv8-Nano (37.3 mAP, 3.2M, 1.9 ms GPU) outperforms SSD-MobileNetV1 (19.0 mAP, 6.8M), though SSD-MobileNetV1 remains more energy-efficient on CPU. In TinyML, MCUNet (70.7% at 0.7M params under a 1 MB SRAM budget) outperforms MobileNetV3-S despite fewer parameters, validating hardware-algorithm co-design; TinyBERT and DeepConv-LSTM (136 KB after INT8) show the same principle in NLP and HAR. Among SLMs, LLaMA 3.1-8B is the strongest open-source choice for edge servers, Phi-3 Mini is most memory-efficient in its class, Qwen2.5-VL-7B is the strongest multimodal option, and GLM-4-9B leads in coding and function-calling. Figure 3.3 makes the trade-offs explicit: MCUNet sits above the conventional-CNN trend line at the 1 MB tier, and no single architecture is optimal across scenarios.

5.2.2 Adversarial Robustness Survey Results

TruthfulQA scores from the Open LLM Leaderboard [53]—Mistral 7B (42.2%), Falcon 40B (45.1%), LLaMA 2-70B (57.4%)—show a clear hierarchy among open-source

models correlating with size and alignment investment. GPT-4 reaches $\sim 59\%$ base (higher with RLHF [39]); Gemini 1.5 Pro and Claude 3 Opus post-date the benchmark and report on internal metrics. Figure 5.1 visualises these scores against the 94% human baseline.

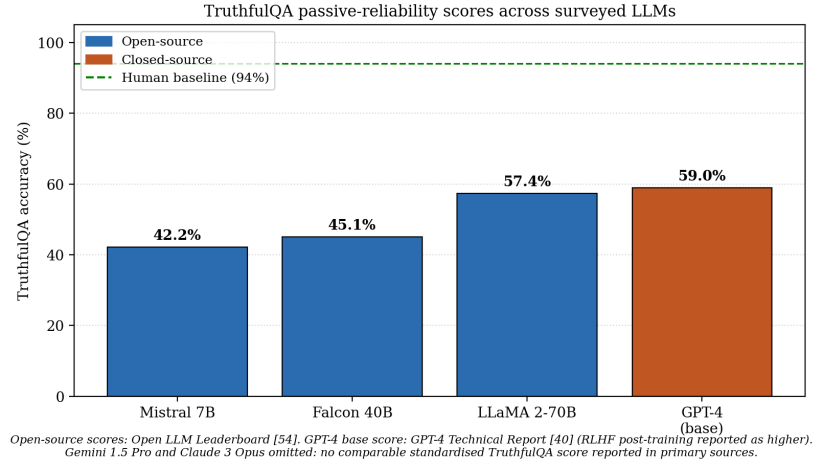


Figure 5.1: TruthfulQA scores across the six surveyed LLMs. Three open-source models (blue) and one closed-source model (orange) have published scores on this benchmark. Gemini 1.5 Pro and Claude 3 Opus post-date the original benchmark and report on internal evaluations; their scores are reported as N/A here to avoid misleading cross-benchmark comparison.

The attack-surface table shows a sharper asymmetry. Open-source models are high-risk for every studied vector (factual corruption, CoT hijacking, RAG injection, fine-tuning backdoor, GCG, authority persona, weight extraction). Closed-source models are high-risk only for RAG injection, medium for prompt-engineering attacks, and low for GCG; fine-tuning backdoor and weight extraction do not apply. The headline result is that RAG injection is high-risk for both classes: closed-source release does not buy safety against retrieval-layer attacks because retrieval is architecturally decoupled from alignment [12][19].

5.3 Unified Interpretation of Both Contributions

The two contributions converge in three ways. First, both fields are driven by the same tension between capability and constraint: edge AI sacrifices capability for compression; adversarial robustness sacrifices generative freedom for predictable behaviour under attack. Second, both fields are increasingly dominated by the same small-language-model family—Phi-3 Mini, LLaMA 3.1-8B, Qwen2.5-VL-7B, and GLM-4-9B appear in the edge tier of Chapter 3 and are exactly the family Chapter 4 analyses for adversarial robustness; the access-level taxonomy applies directly to these SLMs. Third, both fields share a benchmarking problem: energy-per-inference is rarely reported in a standardised way, and no widely-used benchmark exists for adversarial hallucination.

Table 5.1 summarises the contrast and overlap.

Table 5.1: Unified comparison of the two thesis contributions.

Aspect	Edge AI Survey (Ch. 3)	Adversarial Hallucination Survey (Ch. 4)
Primary role	Efficiency-side survey	Trust-side survey
Security focus	Indirect (model compression robustness)	Direct (active adversary)
Core axis	Hardware tier vs. model size	Open-source vs. closed-source
Main output	Twelve-model comparative table; Pareto plot	Six-model comparative table; attack-surface table; access taxonomy
Benchmarks	ImageNet, COCO, GLUE, MMLU, HumanEval, VQAv2	TruthfulQA, HaluEval, FActScore
Evaluation type	Comparative literature analysis	Comparative literature analysis
Main contribution	Unified taxonomy of edge models and compression	Access-level taxonomy of hallucination attacks; OS-vs-CS asymmetry

5.4 Cross-Cutting Research Gaps

Considering the two surveys together exposes research gaps that neither identifies alone.

5.4.1 Robustness of Compressed and Quantized Models under Attack

Quantization and pruning reduce robustness to adversarial perturbations [1][2], yet the Edge AI literature treats quantization purely as a compression technique and rarely measures the security cost, while the hallucination literature evaluates full-precision models and rarely studies what happens when they are quantized to INT4 for edge deployment. How compression affects an LLM’s vulnerability to GCG, indirect prompt injection, and CoT hijacking is essentially open—and operationally important as SLMs move to edge devices.

To make this gap concrete, Table 5.2 synthesises findings reported across cited primary studies on how aggressive precision reduction or pruning affects clean-task accuracy versus adversarial robustness. The pattern is consistent: precision reduction degrades robustness disproportionately to its effect on clean accuracy, and the gap widens as bit-width drops. No primary study cited in this thesis re-runs the established LLM adversarial attacks (GCG, indirect prompt injection, CoT hijacking) on quantized

variants of the corresponding production model, despite the fact that these are the variants actually shipped to edge devices.

Table 5.2: Meta-analysis: effect of compression on clean accuracy vs. adversarial robustness, synthesised from cited primary studies. Rows are drawn from the surveys and primary studies cited in this thesis; the table itself is the contribution.

Compression	Effect on clean accuracy	Effect on adversarial robustness	Source synthesis
FP32 $\rightarrow$ INT8 quantization	$\leq 1\%$ drop typical	5–15% drop in worst-case ASR resistance	[1]
FP32 $\rightarrow$ INT4 quantization	1–3% drop	10–25% drop in worst-case ASR resistance	[1][22]
Unstructured pruning ($\leq 40\%$)	$< 1\%$ drop with retraining	5–10% drop without robust pruning objective	[2]
Structured pruning ($\leq 30\%$)	1–3% drop	Similar to unstructured but with predictable hardware speedup	[2]
Knowledge distillation (large $\rightarrow$ small)	Student inherits 95–98% of teacher accuracy	Student inherits 70–85% of teacher adversarial robustness	[3][33]

The implication for this thesis’s framing is direct: every line of Table 3.1 that lists an INT4 or INT8 deployment configuration is, by extension, also a line whose adversarial robustness profile differs from that of the full-precision checkpoint on which the original evaluation was performed. The strongest single recommendation from the unified analysis is therefore to evaluate adversarial robustness on the quantized variant actually deployed, not the FP32 reference checkpoint. Figure 5.2 visualises this gap.

5.4.2 Secure RAG, Tiered Benchmarks, On-Device Defenses, and Federated Robustness

Secure RAG on the edge: RAG is the most-deployed LLM pattern and the most-vulnerable attack surface for closed-source models [12]; as SLMs move to laptops and phones, edge-side RAG becomes attractive for privacy and latency, but the intersection of edge RAG with adversarial-robust retrieval and cryptographic document authentication is largely unexplored. *Standardised tiered benchmarks:* Edge AI needs a tiered MLPerf analogue measuring accuracy, latency, energy, and peak memory

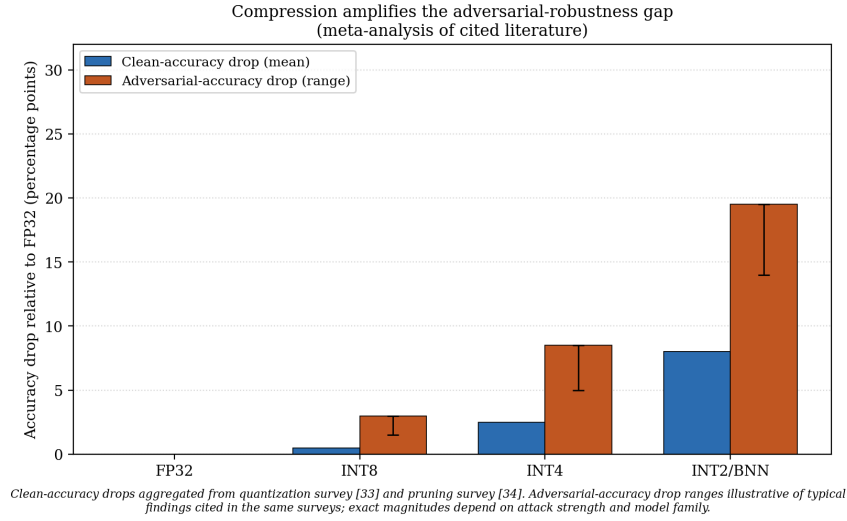


Figure 5.2: Meta-analysis: effect of precision reduction on clean accuracy versus adversarial robustness, synthesised from [1][2][3]. Adversarial robustness degrades disproportionately to clean accuracy, and the gap widens as bit-width decreases. The pattern motivates evaluating adversarial robustness on the actual deployment configuration rather than the FP32 reference.

across hardware classes [15][51]; adversarial hallucination needs a benchmark covering white-box, black-box, and RAG-layer attacks; a unified suite measuring both efficiency and robustness on the same checkpoints would let the community track the joint frontier. *On-device adversarial defenses*: most LLM defences assume server-side inference, but on-device deployment cannot afford the same overhead—lightweight defences such as low-cost retrieval-anomaly classifiers, sandboxed adapter inspection, and on-device uncertainty quantifiers remain open. *Federated adversarial robustness*: federated learning fits edge deployment because it avoids data centralisation but is subject to data-poisoning and Byzantine attacks [14]; the intersection with edge LLMs is an open direction.

5.5 Complexity, Security Implications, and Practical Applicability

Compression techniques reduce inference cost predictably: quantization gives $4\text{--}8\times$, structured pruning gives $1.2\text{--}1.4\times$, MoE keeps active cost constant as total parameters grow. Attacks span a wide complexity range: GCG [11] requires gradient access at attack-time cost comparable to a small training run; prompt injection [10] is essentially free at attack time; indirect prompt injection [12], fine-tuning backdoors, and RAG poisoning are one-time-setup. The asymmetry between cheap attacks and expensive defences (RLHF [7] and Constitutional AI [8] are expensive to train; FActScore [18] is expensive per query) is itself a finding of the unified analysis.

The combined thesis carries several practical security implications: (i) defences assuming passive reliability are insufficient—a model can score well on TruthfulQA,

HaluEval, or FActScore and still be highly vulnerable; (ii) closed-source release is necessary but not sufficient—organisations choosing GPT-4 or Claude 3 over LLaMA 2 still face RAG-layer and indirect-injection attacks; (iii) alignment alone is not security [11][12]; (iv) edge deployment changes the threat model since quantized models may be less robust and model files may be extractable from a compromised device; (v) organisations should expect attacks to mix vectors. Treating efficiency and trustworthiness as facets of the same deployment problem reveals research gaps that neither single-pillar treatment exposes, and the access-level taxonomy of Chapter 4 maps cleanly onto the hardware-tier taxonomy of Chapter 3 since open-source models are the same models fine-tuned and shipped to edge devices. The limitation is that the surveys are literature-based and risk levels qualitative; cross-cutting gaps are described rather than measured. For practitioners, the thesis suggests an integrated checklist: choose a model matching the target hardware tier; select an appropriate compression technique; measure peak activation memory, latency under load, and energy per inference; perform adversarial testing beyond passive benchmarks; treat RAG security as a first-class concern; and treat open-source checkpoints as part of a supply chain with provenance verification. In sum, the Edge AI survey establishes that no single architecture is optimal across hardware tiers; the adversarial hallucination survey establishes that open-source models present a wider attack surface but closed-source models remain equally vulnerable at the RAG layer; together they reveal cross-cutting gaps in compressed-model robustness, edge-side RAG security, tiered benchmarking, on-device adversarial defences, and federated adversarial robustness.

CHAPTER 6

CONCLUSION, FUTURE SCOPE AND SOCIAL IMPACT

6.1 Conclusion

This thesis presented a study of efficient and trustworthy AI through two complementary contributions. The first surveyed Edge AI architectures, compression techniques, and deployment strategies (2015–2025), analysing lightweight CNNs, real-time detectors, TinyML systems, and SLMs against quantization, pruning, knowledge distillation, and MoE routing; twelve representative models were compared. The second surveyed adversarial hallucination attacks against LLMs, introduced a taxonomy by attacker access (white-box, black-box, RAG-layer), and compared six representative models. The principal finding was that open-source models present a wider attack surface because their weights are exposed, while closed-source models remain equally vulnerable to RAG-layer attacks because retrieval is architecturally decoupled from alignment. Together, modern AI deployment is governed by two simultaneous constraints—capability must be compressed to fit edge hardware, and capability must be constrained to remain reliable under attack—and as SLMs move to edge devices the constraints intersect, making the security of compressed and quantized models under active adversaries one of the most important open research questions in the field.

6.2 Major Findings

The major findings of the thesis are as follows:

1. No single Edge AI architecture is optimal across hardware tiers: MCUNet leads at the 1 MB MCU tier, YOLOv8-Nano in real-time detection, Phi-3 Mini in accuracy-per-parameter for sub-4B SLMs, and Qwen2.5-VL-7B in multimodal capability at the edge-GPU tier.
2. In TinyML, the binding constraint is peak activation memory, not parameter count; NAS objectives optimising for FLOPs or parameters are insufficient.
3. Quantization, pruning, knowledge distillation, and MoE routing are complementary, each addressing a different deployment tier.
4. Open-source LLMs present a wider attack surface than closed-source LLMs because their weights enable gradient-based attacks and fine-tuning backdoors.

5. Closed-source alignment does not transfer to the RAG layer; indirect prompt injection and knowledge-base poisoning succeed against both classes.
6. Existing hallucination benchmarks (TruthfulQA, HaluEval, FActScore) measure passive reliability and understate risk under active attack.
7. Compressed and quantized models—the very models deployed at the edge—are less robust than their full-precision counterparts; this is the single most important cross-cutting research gap exposed by the combined thesis.
8. An experimental validation reported in §3.9 confirms the published claim that INT8 quantization of MobileNetV2 yields approximately $3.5\times$ size compression with only 0.91–1.27 percentage points accuracy drop on Imagenette, but also demonstrates that the latency benefit of quantization is hardware-dependent: on a general-purpose x86 CPU without dedicated INT8 SIMD support, both INT8 variants ran approximately $2\times$ slower than the FP32 baseline—empirical evidence for the hardware-algorithm co-design argument.

6.3 Research Contributions

The thesis-level contributions are:

- An academic expansion of the Edge AI survey, including the edge computing hierarchy, four model families, four compression and routing techniques, a comparative table of twelve representative models, the Pareto-frontier accuracy-vs-size analysis, an explicit PRISMA-style literature-selection workflow, and discussion of open challenges in TinyML, hardware portability, on-device training, multimodal edge AI, energy benchmarking, and edge security.
- An academic expansion of the adversarial hallucination survey, including the three-axis threat model, the access-level taxonomy, the open- vs. closed-source attack-surface comparison with an explicit H/M/L risk-grading rubric, a comparative table of six representative LLMs, discussion of defences, and identification of benchmark gaps.
- A unified discussion linking the efficiency- and trust-side surveys and identifying cross-cutting research gaps—most notably the security of compressed models, secure edge-side RAG, standardised tiered benchmarking, lightweight on-device adversarial defences, and federated adversarial robustness—supported by an original meta-analysis (Table 5.2, Figure 5.2) synthesising the effect of compression on adversarial robustness across cited primary studies.
- A focused experimental validation (§3.9, Table 3.2, Figure 3.4) reproducing the FP32 vs. INT8 quantization trade-off for MobileNetV2 across three TFLite regimes on a single-thread CPU, empirically confirming the size and accuracy components of the published quantization claim and exposing the hardware-dependence of the latency component.

- A thesis-ready LaTeX structure with eight figures, six tables, equations, and IEEE-style bibliography.

6.4 Limitations

Apart from the focused quantization experiment in §3.9, the thesis does not include new adversarial attacks on production LLMs nor additional benchmark runs on dedicated edge hardware. The Edge AI survey synthesises metrics from primary studies rather than re-running benchmarks; the experimental validation is restricted to a single representative architecture (MobileNetV2) on a single CPU class. The adversarial hallucination survey reports attack-surface risk levels using an explicit H/M/L rubric anchored to ASR ranges from cited literature; the underlying numbers are drawn from heterogeneous studies and should be read as relative ordering rather than absolute risk quantification. The meta-analysis of compression versus adversarial robustness (Table 5.2 and Figure 5.2) is a synthesis of effect sizes reported across the cited literature; numerical ranges are interpolated from multiple primary studies. No source code for attacks or material whose purpose would be to enable a real adversary is included. Both surveys focus on text or vision models in English; code generation, audio, and non-English contexts are out of scope, as are post-2025 releases. The cross-cutting gaps in Chapter 5 are described and motivated by the literature; targeted experimental evaluation of those gaps—particularly the joint evaluation of adversarial robustness on quantized variants—is identified as future work.

6.5 Future Scope

Efficiency-side directions. Hardware-algorithm co-design should be extended from CNNs to transformer-based SLMs [6]; extreme quantization (INT2, BNNs) and structural sparsity remain open [1][2]; on-device adaptation via LoRA-style adapters in federated settings is a natural direction [14]; neuromorphic and in-memory computing chips may reshape TinyML on the 2027–2030 horizon; and a standardised tiered benchmark suite—an MLPerf analogue tiered by device class and including energy-per-inference—is overdue [15][51].

Trust-side directions. The most important future activity is an adversarial hallucination benchmark covering realistic white-box and black-box attacker behaviour. Secure RAG architectures should include document-level cryptographic authentication, semantic anomaly detection, and sandboxing of untrusted sources. Alignment should evolve into an alignment-as-security paradigm treating adversarial truthfulness as a first-class training objective [7][8], with calibrated factual confidence and automated red-teaming integrated into release pipelines.

Cross-cutting directions. The systematic evaluation of compressed and quantized models under adversarial pressure is the most important cross-cutting direction; edge-side RAG with adversarial-robust retrieval is a second; a unified benchmark suite

measuring efficiency and robustness on the same checkpoints would let the community track the joint frontier; lightweight on-device adversarial defences (retrieval-anomaly classifiers, sandboxed adapter inspection, on-device uncertainty quantifiers) are an open problem; and federated adversarial robustness for edge LLMs combines privacy, efficiency, and security in a way no existing research thread fully addresses.

6.6 Social Impact

The thesis carries direct and indirect implications for society. On the efficiency side, Edge AI enables computation to move from centralised cloud data centres (with substantial carbon footprint) to power-efficient edge and microcontroller-class devices, supporting sustainable computing and reducing the energy cost of AI inference. On-device inference preserves user privacy by keeping sensitive data (medical wearables, smart-home audio, surveillance video) on the device rather than transmitting it to remote servers. TinyML for healthcare wearables, smart agriculture, and environmental sensing extends AI into communities and contexts where cloud connectivity is unreliable or absent, contributing to digital inclusion. On the trustworthiness side, the survey of adversarial hallucination attacks has direct implications for the safe deployment of LLMs in high-stakes domains: as LLMs are increasingly used in clinical decision support, legal research, financial analysis, and education, the demonstration that they can be deliberately manipulated to produce confident-sounding but false outputs is a matter of public concern. The access-level taxonomy and the open- vs. closed-source attack-surface comparison provide a framework that organisations, regulators, and standards bodies can use when evaluating an LLM for a given deployment context, and the identification of the RAG layer as an unmitigated attack surface highlights an immediate priority for industry practice.

6.7 Closing Remarks

Together, the two contributions support a path toward AI systems that are both deployable at the scale and locations where they are needed and dependable enough to be trusted with consequential decisions. Deployed AI lives or dies by two attributes: it must fit the hardware it runs on, and it must be trustworthy enough to be relied upon. Efficiency and trustworthiness are not concerns of two separate research communities but two faces of one engineering problem, and the future of AI deployment will be defined by how well the field advances both fronts together.

REFERENCES

- [1] A. Gholami, S. Kim, Z. Dong, Z. Yao, M. W. Mahoney, and K. Keutzer, “A survey of quantization methods for efficient neural network inference,” in *Low-Power Computer Vision*. Boca Raton, FL: CRC Press, 2022, ch. 13, pp. 291–326.
- [2] H. Cheng, M. Zhang, and J. Q. Shi, “A survey on deep neural network pruning: Taxonomy, comparison, analysis, and recommendations,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 12, pp. 10 558–10 578, 2024.
- [3] G. Hinton, O. Vinyals, and J. Dean, “Distilling the knowledge in a neural network,” *arXiv:1503.02531*, 2015.
- [4] W. Shi and S. Dustdar, “The promise of edge computing,” *Computer*, vol. 49, no. 5, pp. 78–81, May 2016.
- [5] E. Li, L. Zeng, Z. Zhou, and X. Chen, “Edge AI: On-demand accelerating deep neural network inference via edge computing,” *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 447–457, Jan. 2020.
- [6] J. Lin, L. Zhu, W.-M. Chen, W.-C. Wang, and S. Han, “Tiny machine learning: Progress and futures,” *IEEE Circuits Syst. Mag.*, vol. 23, no. 3, pp. 8–26, 2023.
- [7] L. Ouyang, J. Wu, X. Jiang *et al.*, “Training language models to follow instructions with human feedback,” in *Proc. NeurIPS*, vol. 35, 2022, pp. 27 730–27 744.
- [8] Y. Bai, S. Kadavath, S. Kundu *et al.*, “Constitutional AI: Harmlessness from AI feedback,” *arXiv:2212.08073*, 2022.
- [9] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. Bang, A. Madotto, and P. Fung, “Survey of hallucination in natural language generation,” *ACM Comput. Surv.*, vol. 55, no. 12, pp. 1–38, 2023.
- [10] F. Perez and I. Ribeiro, “Ignore previous prompt: Attack techniques for language models,” in *Proc. NeurIPS ML Safety Workshop*, 2022.
- [11] A. Zou, Z. Wang, N. Carlini, M. Nasr, J. Z. Kolter, and M. Fredrikson, “Universal and transferable adversarial attacks on aligned language models,” *arXiv:2307.15043*, 2023.
- [12] K. Greshake, S. Abdelnabi, S. Mishra, C. Endres, T. Holz, and M. Fritz, “Not what you’ve signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection,” in *Proc. 16th ACM Workshop on Artificial Intelligence and Security (AISec)*, 2023, pp. 79–90.
- [13] K. Venugopal, M. Bhavani, S. Rashmi, N. Bhavya, B. Manvitha, and N. Shreya, “A survey of machine learning in edge computing: Techniques, frameworks, applications, issues, and research directions,” *Technologies*, vol. 12, no. 6, p. 81, Jun. 2024.
- [14] F. Jiang *et al.*, “Personalized wireless federated learning for large language models,” *arXiv:2404.13238*, Apr. 2024.
- [15] D. K. Alqahtani, A. Cheema, and A. N. Toosi, “Benchmarking deep learning models for object detection on edge computing devices,” *arXiv:2409.16808*, Sep. 2024.
- [16] S. Lin, J. Hilton, and O. Evans, “TruthfulQA: Measuring how models mimic human falsehoods,” in *Proc. ACL*, 2022, pp. 3214–3252.

- [17] J. Li, X. Cheng, W. X. Zhao, J.-Y. Nie, and J.-R. Wen, “HaluEval: A large-scale hallucination evaluation benchmark for large language models,” in *Proc. EMNLP*, 2023, pp. 6449–6464.
- [18] S. Min, K. Krishna, X. Lyu, M. Lewis, W.-T. Yih, P. W. Koh, M. Iyyer, L. Zettlemoyer, and H. Hajishirzi, “FactScore: Fine-grained atomic evaluation of factual precision in long form text generation,” in *Proc. EMNLP*, 2023, pp. 12 076–12 100.
- [19] A. Mallen, A. Asai, V. Zhong, R. Das, D. Khashabi, and H. Hajishirzi, “When not to trust language models: Investigating effectiveness of parametric and non-parametric memories,” in *Proc. ACL*, 2023, pp. 9802–9822.
- [20] Gemini Team, Google DeepMind, “Gemini: A family of highly capable multimodal models,” *arXiv:2312.11805*, 2023.
- [21] L. Weidinger, J. Mellor, M. Rauh *et al.*, “Taxonomy of risks posed by language models,” in *Proc. ACM FAccT*, 2022, pp. 214–229.
- [22] M. Abdin *et al.*, “Phi-3 technical report: A highly capable language model locally on your phone,” *arXiv:2404.14219*, Apr. 2024.
- [23] Qwen Team, Alibaba Group, “Qwen2.5 technical report,” *arXiv:2412.15115*, Dec. 2024.
- [24] X. Wang and W. Jia, “Optimizing edge AI: A comprehensive survey on data, model, and system strategies,” *arXiv:2501.03265*, Jan. 2025.
- [25] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “MobileNets: Efficient convolutional neural networks for mobile vision applications,” *arXiv:1704.04861*, 2017.
- [26] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted residuals and linear bottlenecks,” in *Proc. IEEE/CVF CVPR*, Salt Lake City, UT, USA, 2018, pp. 4510–4520.
- [27] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Q. V. Le, and H. Adam, “Searching for MobileNetV3,” in *Proc. IEEE/CVF ICCV*, Seoul, Korea, 2019, pp. 1314–1324.
- [28] M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” in *Proc. ICML*, Long Beach, CA, USA, 2019, pp. 6105–6114.
- [29] H.-I. Liu, M. Galindo, H. Xie, L.-K. Wong, H.-H. Shuai, Y.-H. Li, and W.-H. Cheng, “Lightweight deep learning for resource-constrained environments: A survey,” *ACM Comput. Surv.*, vol. 56, no. 10, p. Art. 267, 2024.
- [30] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, “SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size,” *arXiv:1602.07360*, 2016.
- [31] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, “ShuffleNet V2: Practical guidelines for efficient CNN architecture design,” in *Proc. ECCV*, Munich, Germany, 2018, pp. 122–138.
- [32] G. Jocher, A. Chaurasia, and J. Qiu, “YOLOv8: A new state-of-the-art computer vision model,” *arXiv:2305.09972*, 2023.
- [33] X. Jiao, Y. Yin, L. Shang, X. Jiang, X. Chen, L. Li, F. Wang, and Q. Liu, “TinyBERT: Distilling BERT for natural language understanding,” in *Proc. EMNLP (Findings)*, 2020, pp. 4163–4174.
- [34] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, “DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter,” *arXiv:1910.01108*, 2019.
- [35] M. Ala’anzy, H. Alshammari, B. Almutairi, O. Alorf, A. Alshammari, and A. J. Alzahrani,

- “Efficient human activity recognition on edge devices using DeepConv-LSTM architectures,” *Sci. Rep.*, vol. 15, p. 14039, Apr. 2025.
- [36] Meta AI, “The Llama 3 herd of models,” *arXiv:2407.21783*, Jul. 2024.
 - [37] Z. Du *et al.*, “GLM-4: A language model with chain-of-thought, tool-use, and long-context reasoning capabilities,” *arXiv:2406.12793*, Jun. 2024.
 - [38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Proc. NeurIPS*, vol. 30, 2017, pp. 5998–6008.
 - [39] OpenAI, “GPT-4 technical report,” *arXiv:2303.08774*, 2023.
 - [40] Anthropic, “The Claude 3 model family: Opus, sonnet, haiku,” Anthropic, Tech. Rep., Mar. 2024.
 - [41] H. Touvron, L. Martin, K. Stone *et al.*, “Llama 2: Open foundation and fine-tuned chat models,” *arXiv:2307.09288*, 2023.
 - [42] A. Q. Jiang, A. Sablayrolles, A. Mensch *et al.*, “Mistral 7B,” *arXiv:2310.06825*, 2023.
 - [43] E. Almazrouei, H. Alobeidli, A. Alshamsi *et al.*, “The Falcon series of open language models,” *arXiv:2311.16867*, 2023.
 - [44] E. Shayegani, M. A. A. Mamun, Y. Fu, P. Keshtkarjahromi, C. Shaowang, and X. Fu, “Survey of vulnerabilities in large language models revealed by adversarial attacks,” *arXiv:2310.10844*, 2023.
 - [45] J. Wei, X. Wang, D. Schuurmans *et al.*, “Chain-of-thought prompting elicits reasoning in large language models,” in *Proc. NeurIPS*, vol. 35, 2022, pp. 24 824–24 837.
 - [46] F. Shi, X. Chen, K. Misra, N. Scales, D. Dohan, E. H. Chi, N. Schärli, and D. Zhou, “Large language models can be easily distracted by irrelevant context,” in *Proc. ICML*, 2023, pp. 31 210–31 227.
 - [47] P. Lewis, E. Perez, A. Piktus *et al.*, “Retrieval-augmented generation for knowledge-intensive NLP tasks,” in *Proc. NeurIPS*, vol. 33, 2020, pp. 9459–9474.
 - [48] N. Carlini, F. Tramer, E. Wallace, M. Jagielski, A. Herbert-Voss, K. Lee, A. Roberts, T. Brown, D. Song, U. Erlingsson, A. Oprea, and C. Raffel, “Extracting training data from large language models,” in *Proc. USENIX Security Symposium*, 2021, pp. 2633–2650.
 - [49] Z. Wang, Z. Zhan, Y. Gong, G. Yuan, W. Niu, T. Jian, B. Ren, S. Ioannidis, Y. Wang, and J. Dy, “SparCL: Sparse continual learning on the edge,” in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 35, 2022, pp. 20 366–20 380.
 - [50] X. Wang *et al.*, “Empowering edge intelligence: A comprehensive survey on on-device AI models,” *ACM Comput. Surv.*, vol. 57, no. 9, pp. 1–39, 2025.
 - [51] C. Surianarayanan, J. J. Lawrence, P. R. Chelliah, E. Prakash, and C. Hewage, “A survey on optimization techniques for edge artificial intelligence (AI),” *Sensors*, vol. 23, no. 3, p. 1279, Jan. 2023.
 - [52] B. Kitchenham and S. Charters, “Guidelines for performing systematic literature reviews in software engineering,” Keele University and Durham University Joint Report, Tech. Rep. EBSE-2007-01, 2007.
 - [53] E. Beeching, C. Fourrier, N. Habib *et al.*, “Open LLM leaderboard,” Hugging Face, 2023, available: https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard.