# The Service Provider Path to Drive the Home Edge AI Solution to Scaling Home Inference Requirements

A technical paper prepared for presentation at SCTE TechExpo25

**Charles Cheevers**
Chief Technology Officer
Vantiva
Charles.Cheevers@vantiva.com

# Table of Contents

**Title** **Page Number**

## List of Tables

**Title** **Page Number**

# 1. Introduction

The rise of Cloud based Generative AI solutions has created a burden and opportunity for the Service Provider. The burden is carrying these new low latency tokens that drive the next word answers of Large Language models. The opportunity is to find a Service Provider Value insertion in this new AI driven future. To find the value equation and to drive a cost to return on investment the following elements are typically considered in the use of any new technology – and AI is no different

- Save Money – typically looking at areas like Generative AI based Customer Support and other automated workflows
- Make Money – new AI driven services or value additions into the AI workflows and ecosystems including the additional customers that value this new feature/service.
- Solving some inertia or issue that a consumer values – in this case one of the main areas of opportunity is going to be solve the potential huge issue of consumer privacy.

Service Providers are in a unique strategic position – they own or manage the Customer Premises Equipment (CPE). This opens opportunities for differentiated, AI-enabled offerings that go beyond basic connectivity – all while addressing key concerns around privacy, latency, and cost. There seems to be a desire to move as much offload to the client edge for AI processing which

- Defers the dependency on building out the DataCenter Infrastructure to higher and higher token counts for AI services.
- Distributes the Power and Water cooling problems to the client edge and in particular the home.
- Increases the ability to use greener power sources (Home Solar etc) to charge SmartPhones and power Home based devices using AI services
- Keeps consumer data like images and audio local to the CPE devices to process to less sensitive formats like text.

The concept of a Hybrid – Home Edge and Cloud – seems to offer the best path to lowest power (or at least decentralized power use) and highest privacy. It also offers the Service Provider the most opportunity to insert in the AI value path for minimizing Opex, translating Capex investment into return and enforcing privacy as a key element.

As a quick example of the energy saving drivers to support this architecture – recent academic research provides powerful quantitative validation for the benefits of the hybrid model. A January 2025 paper by Siavash Alamouti, "Quantifying Energy and Cost Benefits of Hybrid Edge Cloud," presents a detailed mathematical model and simulation. The study assumes that AI workloads often follow a Pareto distribution, where a small number of tasks (e.g., 20%) are resource-intensive, while the majority (e.g., 80%) are lightweight and suitable for local processing.

Based on this model, the findings are striking. For a typical smart device with traditional workloads, a Hybrid Edge Cloud (HEC) model that processes 80% of tasks locally achieves **energy savings of approximately 65%** compared to a fully centralized cloud model. For more intensive "agentic" workloads, such as those from AI agents or autonomous systems generating 20 GB of data per day, the HEC model still achieves **energy savings of up to 75%**. This translates into a reduction of approximately **10,000 kWh per device, per year**. Other research corroborates this, showing that the reduced communication latency achieved by using edge servers brings significant benefits in terms of the energy consumption of client devices.

The following table synthesizes data from across the research to provide a comparative estimate of the annual carbon footprint per user, illustrating the scale of these savings.

**Table 1- Home Edge AI Offload Carbon Emissions Saving**

| Footprint Component | All-Cloud Model (gCO2e/year) | Home-Offload Model (80% offload) (gCO2e/year) | % Reduction |
|---|---|---|---|
| **Operational Carbon (Compute)** | 1,080,720 | 288,192 | 73.3% |
| **Operational Carbon (Network)** | 609,840 | 121,968 | 80.0% |
| **Embodied Carbon (Annualized)** | 15,000 | 5,525 | 63.2% |
| **Total Annual Carbon Footprint** | **1,705,560** | **415,685** | **75.6%** |
| Model assumptions: Based on Alamouti's agentic workload (7,300 GB/year). Cloud compute assumes 1.5 kWh/GB [41], PUE of 1.58, and average US data center grid intensity of 548 gCO2e/kWh. Network assumes 0.7 kWh/GB and global network carbon intensity. Home-Offload compute assumes 0.5 kWh/GB local processing on a US average residential grid. Embodied carbon is annualized over a 4-year lifecycle, comparing a share of a cloud server vs. a CPE device with 90% refurbishment savings. | | | |

This quantitative model, while based on a set of assumptions, demonstrates the dramatic potential of the home-offload architecture. The total annual carbon footprint per user is reduced by over 75%. The largest savings come from slashing the energy-intensive network transmission and shifting the bulk of computation from inefficient, high-carbon data centers to more efficient local processing.

A hybrid “hierarchical AI” architecture spanning cloud, network edge, and home devices can yield optimal performance and cost efficiency, with the Service Provider acting as the orchestrator of AI processing across these layers. Several factors make Home Edge AI not just an opportunity but a necessity for Service Providers:

## 1.1. Privacy and Trust

Consumers are increasingly cautious about transmitting personal audio, video, and sensor data to cloud servers for AI processing. Home-edge AI solutions retain sensitive data on-premises, under the control of both customers and service providers, thereby significantly minimizing risks associated with breaches or misuse. For instance, an AI-enabled gateway can analyze camera feeds or voice commands locally, ensuring that raw personal data is not uploaded to external servers and addressing regulatory and consumer expectations regarding data sovereignty. Localized processing fosters trust and inherently aligns with privacy regulations.

This domain presents substantial opportunities for Service Providers to investigate their potential roles as 'Trusted Providers' of AI services or intermediaries between AI memory systems and consumers. Although some consumers may be prioritizing simplicity, convenience, and cost over privacy and security, advancements in AI are likely to heighten awareness of intimate learning by these technologies. Service Providers may consider developing methods to anonymize user interactions with AI services or, at the very least, offer tools to monitor and audit AI usage within homes. For example, gateway devices could observe packet flows and service access to cameras, microphones, and other sensors integrated into AI systems, providing feedback on any anomalies or unexpected behavior.

## 1.2. Latency and Reliability

Running AI inference at the network edge reduces response times by avoiding cloud latency, making it ideal for real-time applications like voice assistants, gaming, AR/VR, and security. Local processing maintains core smart home functions during internet outages and provides faster, more reliable results than cloud-only solutions. Simple edge applications use optimized CNNs and RNNs for speech and vision tasks, such as predictive text on smartphones and local vision models for human activity, improving speed and privacy by limiting cloud data transfers.

## 1.3. Bandwidth and Cloud Cost Savings

Running AI tasks on home devices lowers the need to send large amounts of data to the cloud, saving bandwidth and reducing ongoing cloud costs. Investing in smarter home equipment shifts some expenses from monthly cloud fees to upfront device purchases. For instance, a capable home AI gateway may cost more initially but can process thousands of inferences locally at minimal extra cost, unlike cloud services that charge per use. Edge processing also spreads energy consumption across multiple small devices, some powered by renewables like home solar, decreasing dependence on power-intensive data centers and supporting greener operations.

## 1.4. New Revenue and Service Differentiation

Integrating AI at the home edge enables Service Providers to participate more directly in the AI value chain, an opportunity not available when intelligence is confined solely to the cloud. Instead of surrendering all AI-driven, value-added services to over-the-top (OTT) providers and hyperscalers, Service Providers can utilize their existing infrastructure to deliver distinctive services that are closely aligned with their network operations and customer premises equipment (CPE). By adopting NPU-based AI within gateways and set-top boxes, Service Providers elevate their offerings beyond standard bandwidth to encompass advanced, intelligent services. These may include AI-driven Wi-Fi optimization, proactive customer support, and comprehensive smart home automation that leverages connectivity—services where broadband providers hold a distinct advantage.

## 1.5. A Large Language Model Home AI Server

The concept of Home Edge Architecture involves adding enough local AI performance—through a new Home AI Server or a high-inference sidecar for routers—to handle tasks within the home. As cloud AI inference costs decline with better processors, generative AI access is now divided into at least three tiers (with premium models expected for top performers). The current classification includes:

- Free Tier: No ads, limited prompts and model access, mainly for UI and data training.
- Plus Tier: $20-$30/month, some limitations remain, user learning is ambiguous, restricted video generation.
- Pro Tier: $200/month, top models and unlimited text prompts, but video generation remains limited.
- Enterprise/Collaboration Tier: Focuses on security and sandbox workspaces to keep company data separate from training data.

By mid-2025, consumers use **10,000 to 50,000 AI tokens** daily due to widespread integration in digital tasks such as searches, summaries, emails, and image creation. By 2030, AI will be embedded in homes and workplaces, making usage continuous and raising average daily token consumption to **500,000 to 2 million tokens**, driven by increased interaction and dedicated devices.

## 1.6. The Rise of Multimodality

As interactions incorporate voice, image, and video, token costs will rise due to increased computational demands. Future AI systems will proactively manage tasks and information, further boosting daily token usage. AI's integration into professional sectors like software development and healthcare will drive high token consumption by handling large datasets and complex operations. By 2035, AI will be seamlessly embedded in daily life, making a separate "digital life" obsolete, with average daily token use projected at **5 million to 20 million tokens per consumer**.

Key drivers of exponential growth include:

- **Hyper-Personalized Environments:** AI will tailor homes, vehicles, and devices to individual needs through constant learning and data processing, generating a steady flow of token transactions.
- **Internet of Intelligent Things (IoIT):** IoT will advance to IoIT, where everyday objects feature decentralized AI agents that communicate and collaborate, expanding personal token footprints.
- **AI-Powered Entertainment and Creation:** Entertainment will shift to AI-generated, personalized content. Both professionals and hobbyists will use advanced AI tools for creative tasks, each activity adding to the token count.

This paper examines the opportunities for Home Edge AI and Hybrid AI architectures in areas like energy savings and service provider engagement amid rising demand for AI at home. Home Edge AI offers faster, private user experiences and creates new revenue streams for providers, while hyperscalers benefit from offloading tasks to the edge. Potential business models could allow anyone investing in processing infrastructure—consumers included—to offer compute resources for AI workloads. For years, ideas have circulated about utilizing millions of CPE devices to perform compute tasks during idle periods; now, with enhanced NPUs and more DRAM, running ML and inference at the edge is increasingly feasible. Although this paper does not detail distributed CPE clusters, it acknowledges their potential. Additionally, powering STB inference engines may require rethinking Code of Conduct and Sleep/Idle requirements for CPEs, especially as they offset cloud energy costs by staying active for new services.

Let us begin by stating the problem Service Providers face in an AI landscape dominated by hyperscalers, then examine the current CPE device capabilities and emerging hardware, map AI model types to these edge devices (with emphasis on what can run on modest hardware like 2.5 TOPS NPUs and 2GB RAM), survey a rich matrix of use cases and the value they create, and finally provide actionable recommendations – including an evolutionary roadmap and a hybrid architecture – for deploying AI at the home edge in a cost-effective, strategic manner. Service Providers sit at the threshold of a new era where every gateway and set-top could become a mini AI cloud, and those who move first to architect and deploy these solutions stand to gain a significant competitive advantage.

## 2. Problem Statement and What We Suggest Solving

Over the past 18 months, Service Provider organizations have been actively searching for effective AI use cases, particularly those enabled by advances in Generative AI. Every department is exploring how to integrate AI services and workflows into both their operations and the products offered to residential customers—broadband, video, IoT, and mobile smart devices. This paper will focus specifically on residential applications of AI, excluding enterprise/small business verticals and network infrastructure opportunities, though some overlap exists with telemetry-based AI decisions at data-rich endpoints such as CPE devices. We will briefly address network AI benefits for individual homes, clusters, and broader headend-level problem resolution.

Machine Learning has long been used by service providers to analyse consumer behaviour, Wi-Fi performance, and predict network capacity. Previously, large datasets went mostly unused due to high development costs for analytics software. However, new AI tools can now extract valuable insights from structured and unstructured data, making analysis faster and less wasteful compared to storing entire datasets in datalakes that are frequently purged.

This paper explores how traditional Service Provider CPE devices can be enhanced with AI features, recommending updated hardware, architecture, and approaches to increase AI inference capabilities in the home. We will review current use cases, focusing on smaller AI models like CNNs and RNNs as likely initial solutions at the edge, while also considering the potential of transformer-based Small to Medium Language models before resorting to large-scale cloud LLMs. Integrating generative AI presents unique challenges regarding investment and ROI, a topic expected to remain relevant for the next five years. Both connectivity/video silicon vendors (e.g., Broadcom, Qualcomm, Mediatek) and data center AI providers (e.g., Nvidia, AMD) are actively developing silicon solutions for home applications, which currently fall into several main categories.

- Lowest cost: Uses traditional BB Gateway or STB Host Processors with basic CPU, GPU, and low-power NPUs (1.5–10 TOPS), limited to DDR4 memory and modest token rates.
- Medium cost: Features newer silicon with enhanced AI processing via onboard or external AI processors (20–50 TOPS) and at least 8GB LP-DDR5 DRAM.
- Highest cost: Designed for retail, SMB, or heavy AI users, these Home AI servers (Nvidia, AMD; $1,000–$3,000+) handle large open-source SLMs (e.g., LLaMA, Mistral, DeepSeek, HuggingFace) and scale to cloud-level models. While costly for most consumers, projected household AI usage may justify the investment over time. Service providers may consider offering such devices to meet growing in-home AI demand.

Continuous advances in AI processors and more efficient LLMs—including specialized derivatives and potentially smaller models for hierarchical uses show that the Edge presents a significant opportunity for Service Providers. The current focus is on initial steps and incremental strategies to add value in an evolving AI-driven landscape.

## 2.1 Service Provider Value in the AI Inference Chain

Today, hyperscalers like Google, Amazon, Microsoft, and Meta dominate the AI ecosystem by offering cloud-based APIs and services that use Service Provider networks mainly for data transport. While hyperscalers profit from AI-related revenues, Service Providers mostly bear data transport costs without equivalent returns. To gain more value, Service Providers should leverage on-premises devices and edge infrastructure to run AI services directly. As AI use cases expand in broadband, video, IoT, and 5G, Service Providers can capture new opportunities by moving inference closer to users. The following sections outline these potential value growth areas.

### *2.1.1 Voice and Natural Language Services and Rise in Multimodal Interfaces to Access Realtime AI*

Voice assistants like Amazon Alexa and Google Assistant commonly depend on cloud processing, but advancements such as ChatGPT's multimodal interfaces and Google's Project Astra are driving a shift toward more local AI capabilities. With the rise of wearable devices like smart glasses, video and audio AI processing is expanding on both mobile and home platforms. Service providers can leverage existing broadband gateways, set-top boxes with cameras and microphones, and IoT devices to offer multimodal services that enhance privacy and reduce latency by keeping data internal. Applications include moving speech-to-text, video, and audio analysis closer to the edge, sending only essential information to the cloud rather than complete raw data. For instance, security cameras can process inferences locally, transmitting only necessary information unless an incident occurs.

### *2.1.2 Broadband Network Optimization*

AI enhances broadband services by enabling smart QoS through local pattern recognition on routers, allowing instant traffic classification and prioritization of activities like streaming or gaming. Anomaly detection models can predict network issues by monitoring latency and device behavior, while AI in gateways can detect security threats from unusual IoT traffic without sending data to the cloud. These capabilities lower support costs and boost customer satisfaction through automatic troubleshooting and proactive maintenance.

### *2.1.3 Video and Entertainment Services*

Cable and IPTV providers can use AI in set-top boxes (STBs) or smart TVs to improve video experience. Modern STB chips with NPUs enable real-time computer vision tasks like person detection, gesture recognition, and eye tracking. Features such as AI video upscaling, scene recognition, and noise reduction are now possible on-device, offering better picture quality and interactive services. Edge AI also helps personalize content by analyzing viewing habits locally, maintaining privacy. For example, an STB could host an “AI sports co-commentator,” providing live stats and predictions without cloud delay.

### *2.1.4 Smart Home IoT and Security*

Many Service Providers now include home security, automation, or IoT hub services. Placing AI inference in the home gateway enhances these features, allowing security cameras to process video locally for object and facial recognition without sending data to the cloud, preserving privacy. AI can also aggregate sensor data to detect unusual patterns or routines, benefiting applications like elderly care by identifying deviations that may signal distress. Some gateways even use Wi-Fi signals to recognize activities such as breathing or falling. These advanced features can be bundled into premium safety packages, increasing revenue and customer loyalty for Service Providers.

### 2.1.5 Customer Support – Merging Network and CPE Telemetry to AI Driven Remediations

AI presents significant opportunities to redesign Customer Support, transitioning from simple chatbot expert systems to advanced Generative AI solutions. With technologies like Retrieval Augmented Generation and access to network telemetry data, AI can now triage issues using LLMs with TR181 and CPE logs. Enhanced consumer interactions, empathetic voice responses, and realistic avatars further improve support, reflecting new standards in video conferencing since 2020.

After identifying key principles to highlight the Home Edge device's role in advancing AI services, it's important to consider Service Providers' business dynamics and ROI needs. There are three main categories influencing investment in new architectures to keep Service Providers relevant as AI expands into homes and mobile devices.

- Saving Money with AI Services
- Making Money with AI Services
- Retaining customers and growing NPS and stickiness via the use and adoption of AI Services

If an AI-assisted application or service does not deliver ROI in these three categories, it is unlikely to be adopted. This serves as a useful benchmark for deciding whether to continue using current programmatic data processing solutions.

Here's a brief overview of the three key areas and the main ideas that could drive AI investment towards ROI.

### 2.1.6 Saving Money with AI Services; The Primary Use Cases

Using generative AI and multi-modal interfaces to replace Tier 1 and Tier 2 human customer support, triage issues, and automate problem resolution could enable a fully automated AI-driven support workflow. While implementation is complex and touches existing Backoffice systems, this approach offers a promising path for complete automation.

- AI chatbots enabling new multimodal interactions with voice, video, or avatars
- AI chatbots using detailed consumer device and telemetry data for real-time home insights
- AI chatbots converting TR181 and Syslog/RDKlog/logcat files into actionable support information
- Local SLM optimized for basic first-level support

It is possible to integrate Generative AI into Edge devices and even low-spec hardware like gateways or set-top boxes. These SLMs can then offer additional capabilities.

- Summarizing TR181 and device log data locally may reduce costs and detect anomalies faster than traditional 15-minute ACS polling, by only reporting changes and skipping redundant data. However, current small quantized LlaMA models often lack sufficient accuracy for key Wi-Fi and customer support metrics. Developing dedicated sub-1Bn models could improve consistency, but further research is needed to gauge their effectiveness in interpreting CPE telemetry.
- Compact, highly quantized generative AI SLMs could facilitate customer interactions through text, voice, or video, but present models like LlaMA 3.2 (8Bn and lower) struggle with accurate Wi-Fi troubleshooting due to their limited precision. More development is necessary to achieve reliable performance and clear ROI.

#### 2.1.6.1 Leveraging CNN/RNN Models in the CPE Edge Devices to Minimize use of Those Models in the Cloud for AI Workflows.

### Key Areas Include:

- Speech and audio inference, including speech-to-text, transcription, translation, and dubbing.
- Using household noises for security and related applications.
- Enhanced audio privacy: Limit external sharing of locally generated files and avoid cloud storage of source audio whenever possible.
- Video inference.
- Edge processing of camera feeds: Actions and analysis occur locally to reduce cloud compute and storage costs.
- Improved video privacy: Minimize raw video uploads; most analysis (e.g., person detection, movement models, emotion recognition, counting people) happen locally, with only outcomes sent to the cloud.

##### 2.1.6.1.1 Security Applications

Traditional security applications rely on both CPE agents and cloud-based learning models to detect anomalies and patterns. With AI-powered NPU-enabled CPE devices, more security tasks and advanced pattern detection can now be performed locally. This Edge-based process offers improved security while reducing cloud computing costs.

##### 2.1.6.1.2 Device and Service Fingerprinting

Training and running inference locally on device and service usage patterns can be enhanced by deploying AI models to NPUs for packet flow analysis on CPE devices. Accurate device and service fingerprinting is increasingly valuable for AI-driven customer support and troubleshooting, especially when using conversational interfaces that distinguish devices by type and model rather than just MAC address. It's also important for diagnosing issues, such as explaining why YouTube performs poorly on a specific device with weak connectivity and frequent reconnections. For effective AI-based support, identifying devices and services by name rather than technical identifiers is crucial for clear human interaction and problem resolution.

##### 2.1.6.1.3 Making Money with AI Service Provider Hybrid Services

The main use cases for this area are not easily defined by a single killer application. Currently, there are several promising verticals and broader ideas that require a wide perspective on the evolution of consumer services over the next five years.

- Video-based services present significant opportunities for AI enhancement.
- Vision AI models using cameras, including optimized CNNs, can run at the edge or be integrated into AI-enabled devices like NPU STBs and broadband gateways. These models, suitable for edge applications, now support facial recognition, object detection, and tracking.
- Entertainment video on TV: Analysing user patterns across movies, audio, and gaming enables local content recommendations based on household inputs such as users and schedule.
- Shoppable TV solutions have advanced by leveraging video CNNs to identify items (e.g., clothing) for matching with retail inventory, enabling scenarios like voice-driven shopping via AI speech and video object detection.

- Advertising AI augment services: Generative AI enables personalised, detailed product expertise and engagement through avatars, tailored recommendations, and compatibility checks within the home environment.
- Telemetry, monitoring, and energy management services can also benefit from AI augmentation.

Local CNN/RNN models on CPE devices can help correlate IoT-controlled devices, especially high-power ones like HVAC, dishwashers, and laundry machines. Future systems could integrate multiple data sources, such as presence detection, to further optimize energy management and enable collaboration with utilities. Additionally, using a TV or set-top box with an avatar-based Energy Assistant from your utility may complement smartphone apps in the future.

- Presence Monitoring AI services
    - The world is moving toward widespread sensor-based technology, with standards like 6G and Wi-Fi 8/9 enabling large-scale RF sensing. Homes now feature Wi-Fi, BLE beacon sensing, UWB, and possibly 60GHz radar or FMCW solutions. Local models process these data sources for location, presence, and motion inference, boosting Edge AI in smart homes.
    - New AI Gateway verticals for the home
    - AI-assisted elderly care and telemedicine
    - AI for monitoring elderly wellness
    - Security and trust in elder care homes
    - Privacy protection in AI-powered homes

The Service Provider can introduce a transparent two-factor system to serve as a trusted intermediary between smart home applications and concerns about data misuse. By partnering with a Humanoid Robot provider, they could offer enhanced privacy and security, ensuring robot telemetry is independently reported to the consumer rather than the robot supplier.

Others opportunities that will emerge as we enter a more AI led servcies for consumers can potentially include

- Introduce new multimodal interfaces for hands-free interaction with Generative AI SLM at home.
- Service Providers can bundle partnered LLMs for customers, similar to video streaming bundles, integrating with home and mobile environments to deliver lower latency and enhanced transparency.
- Implement private information vaults for homes and users to support LLM learning, with anonymization options for privacy when prompting.
- Enable easy in-room voice and video access to LLM assistants.
- OpenAI's acquisition of Love.io signals always-available multimodal devices for personal assistant access across home environments, utilizing STB/TV hardware.
- Use TV/STB as a multimodal platform for upselling services and deploying AI-generated avatars.
- Leverage TV screens for innovative advertising via avatar-based, human-like sales interactions, enhanced by generative AI for personalized experiences—such as purchasing a car from a knowledgeable avatar.
- Partner with frontier LLM companies to offer secure, private AI services.
- Collaborations between Service Providers and LLM vendors enable tailored, private home solutions, optimize network speeds, and utilize both cloud and edge infrastructure efficiently. Telemetry and home data can be used to further customize offerings for various household types and needs.

Service Providers currently see AI value mainly captured by third parties in the cloud. To reclaim this, SPs can integrate home-edge AI with Cloud AI partnerships, which requires addressing device limitations and capabilities. The next section details these device factors.

# 3. Current Device Landscape and Emerging AI Capabilities

AI inference requires significant computing power, but hardware for these tasks is quickly becoming part of customer equipment. This section reviews Service Provider hardware, including gateways, Wi-Fi APs, set-top boxes, and emerging AI servers—evaluating CPU (DMIPS), NPU (TOPS), and DRAM specs. Devices range from basic units for lightweight models to high-end systems comparable to PCs. Benchmarking helps match AI workloads to suitable device classes.

## 3.1. Broadband Gateways & Wi-Fi Routers

Modems and routers deliver internet access and manage home Wi-Fi, typically using basic CPUs and hardware accelerators for packet routing, with minimal AI capability. However, modern gateway SoCs now include dedicated NPUs for AI models. Recent broadband gateways and routers offer over 30K DMIPS performance, supporting packet processing well beyond 10Gbps speeds and can run simple CNNs and non-Realtime AI models.

**Table 2 - Factors Driving the Need for AI in the Gateway**

| Criteria | Good for CPU | Needs NPU / GPU |
|---|---|---|
| Model Size | Small (<10M params) | Large (>10M params) |
| Compute Type | Light compute, SIMD | Heavy matrix mult, convolutions |
| Use Case | Tabular, simple classification, KWS | Vision, LLMs, real-time video AI |
| Latency tolerance | High | Low |
| Memory BW requirements | Low to moderate | High |
| Power efficiency | Acceptable at low throughput | High throughput with low power per inference |
| Examples | TinyML, LightGBM, DistilBERT, KWS | YOLOv5, ViT, Whisper Large, LLaMA, SD |

With NPU enhancements in the gateway and router, the edge can now support models larger than 500MB and high compute-memory ratios. Early gateways and routers offer about 2.5 TOPS of AI inference, and with over 500MB of available memory (preferably high-speed LP-DDR5), they're capable of enabling important physical and packet layer inference applications.

Some of the early model applications being considered or developed are

- Packet-level inference can support security, device/service fingerprinting, QoS management, and related applications.
- PHY-level inference uses FFT or CSI data from Wi-Fi systems to troubleshoot technologies like DOCSIS® QAM or enhance Wi-Fi features such as motion detection and adaptive antenna gain.

AI edge applications are advancing quickly, with packet and PHY level inference already used by security and fingerprinting firms via CPE agents and cloud systems. Now, companies can leverage local NPU resources to enhance their solutions.

While compute power is important, memory size and model concurrency remain significant challenges, especially for real-time, token-rate limited inference. Broadband gateways have gradually increased

DRAM from 512MB in older devices to 1GB, and now up to 4GB in high-end Wi-Fi 7 Gateways, as they support more applications and AI models. However, current devices still use DDR4 DRAM and haven't yet adopted faster DDR5, which could further reduce inference latency.

**Table 3 - The Performance Factors of DDR5 vs DDR4 DRAM**

| Parameter | DDR4-based AI SoC | LPDDR5-based AI SoC |
|---|---|---|
| Bus Speed | 2133–3200 MT/s (max theoretical bandwidth ~25.6 GB/s) | 4266–6400 MT/s (max bandwidth ~51.2 GB/s) |
| Access Latency (ns) | ~60–75 ns typical | ~40–55 ns typical |
| Power Efficiency | ~1.2V, higher active and idle power | ~0.5–0.6V, lower active and idle power |
| Inference Times | Slightly higher due to lower bandwidth (depends on model size; typical ~1.2–1.4x slower on memory-bound ops) | Faster on memory-heavy layers (batchnorm, large CNNs, transformer KV caching) |
| Tokens/sec (LLaMA 7B on 2–10 TOPS) | ~5–10 tokens/sec depending on model and quantization | ~8–15 tokens/sec (up to 1.5–2x faster in some KV-heavy workloads) |
| Sustained Bandwidth (real-world) | ~10–15 GB/s in AI workloads (copy + compute overlap limits) | ~20–35 GB/s effective in well-optimized AI pipelines |
| Memory Capacity Typical | 2–8 GB | 4–16 GB |
| Use Case Examples | Entry AI CPE, IoT gateways, low-end NPU devices | Premium AI smartphones, advanced AI gateways |

Several new solutions have been proposed to increase inference TOPS and DRAM for running larger models, such as SLM and Transformer models, at home on Edge devices.

The CoPilot laptop market is driving Silicon platforms with at least 40TOPS for AI inference, making these devices potential candidates for affordable Home AI solutions. Gateway and Router designs are being evaluated for onboard high-inference, high-DRAM options versus sidecar or standalone AI servers, with ongoing discussions about cost-effectiveness. Device costs range from $50 to $500+ for running large models, while attention centers on the value and added benefits of sub-40TOPS and 8GB LP-DDR5 systems compared to standard Gateway platforms at 2.5TOPS.

**Table 4 - AI Applications for Different TOPS and DRAM Levels**

| Category | 2.5TOPS, 4GB DDR4 | 25TOPS, 8GB DDR5 | 40TOPS, 16GB DDR5 | 1200TOPS, 512GB DDR5 |
|---|---|---|---|---|
| Basic AI Inference | Small CNN/RNN, lightweight keyword spotting, anomaly detection on telemetry | Medium CNNs, lightweight transformer (e.g., MobileBERT) | Multiple concurrent models, vision + NLP for home | Full GPT-3.5 class LLMs, advanced multimodal |
| Voice Assistants | Wake-word, local command recognition | Local NLU, short sentence parsing | Full local assistant, continuous conversation | Personalized large context conversations |
| Wi-Fi Optimization (TR-181 models) | Basic band steering, anomaly detection | Dynamic QoS tuning with predictive models | Adaptive multi-device load balancing | Holistic traffic shaping with predictive LLM-based models |
| Security & Privacy | Basic intrusion detection, device fingerprinting | Behavior anomaly detection per device | AI-driven DPI & dynamic isolation | Full zero-trust AI orchestration locally |

| Category | 2.5TOPS, 4GB DDR4 | 25TOPS, 8GB DDR5 | 40TOPS, 16GB DDR5 | 1200TOPS, 512GB DDR5 |
|---|---|---|---|---|
| BLE/Thread Device Management | Device presence detection, basic sensor fusion | Simple gesture recognition, local environment context | Predictive device usage & automation triggers | Full environment modeling with semantic context |
| Video Analytics (Local) | Motion detection, people counting | Basic face detection & license plate blurring | Full object detection, fall detection, intruder detection | Multi-stream analytics, re-identification, advanced AR overlays |
| Smart Home Control | Rules-based triggers | AI learned automations | Context-aware automations | Personalized environment orchestration |
| Energy Optimization | Basic monitoring | Predictive usage reduction | Optimization with appliance pattern recognition | Grid-aware, demand-response local AI control |
| Healthcare Monitoring | None or threshold-based alerts | Simple motion analysis | Fall detection, respiration monitoring | Advanced health vitals extraction via vision, multimodal |
| Data Models Feasibility (TR-181 + extended) | Limited real-time inference, periodic processing | Near-real-time adaptation to device models | Continuous learning from telemetry | Full reinforcement learning on local data |
| Model Size Practicality | Up to 50MB models | Up to 500MB models | Up to 2GB models | Up to 200GB+ models |
| Concurrent Streams | 1-2 low complexity | 4-6 moderate | 8-12 high complexity | 50+ ultra-high complexity |
| Value-Add Features for Service Provider | Entry-level AI service differentiation, basic home monitoring upsell | Tier-2 smart home services, advanced parental controls, premium security | Home health services, advanced AI assistants, smart energy | Premium AI home concierge, AI-powered eldercare, advanced security and AR services |

Double clicking on the model types supported

- **2.5TOPS, 4GB DDR4:**
  TinyML, MobileNetV2, SqueezeNet, ESPnet-KWS
  TensorFlow Lite / EdgeTPU models under 50MB
  Simple ARIMA for predictions
- **25TOPS, 8GB DDR5:**
  MobileBERT, DistilBERT for NLU
  YOLOv5n, EfficientDet-Lite
  TFLite or ONNX quantized models
  Predictive anomaly detection models
- **40TOPS, 16GB DDR5:**
  BERT base / RoBERTa base
  YOLOv5m, EfficientDet-D2
  Small speech-to-text (Whisper small)
  Multi-modal fusion (audio + video)
- **1200TOPS, 512GB DDR5:**
  LlaMA-3.x / Mixtral / advanced RAG with local vector stores
  LLaVA-1.6 / MM1 multimodal models
  Whisper large, advanced STT + TTS
  Large ViT, Stable Diffusion locally for AR/VR
  Reinforcement learning models for adaptive control

A core part of any SP's Gateway Strategy is maximizing use of existing inventory and maintaining feature parity between old and new Gateways. However, adding AI models at the edge, which requires more memory and specialized processors like NPUs, makes this challenging. Gateway processing and memory demands have evolved significantly from over five years ago to the current Wi-Fi 6/6E and upcoming Wi-Fi 7/8 devices.

- *Basic ISP-Supplied Router (early 2020s):* ~5k DMIPS dual-core ARM CPU, 256–512 MB DDR3, no NPU. Minimal AI—only simple tasks like anomaly or keyword detection; mainly used for networking.
- *Modern Mid-Range Gateway (2023):* ~12k DMIPS quad Cortex-A53 @1.5GHz, possibly a 0.5–1 TOPS DSP, ~1 GB DDR4. Handles basic AI such as Wi-Fi motion sensing or ML QoS, but heavier computation relies on the cloud.
- *High-End Wi-Fi 7 Gateway (2024/2025):* Quad Cortex-A73 + A53 (~20k+ DMIPS), up to 2.5TOPS NPU, 2–4 GB DDR4. Focused on local packet/PHY inference and smart home monitoring AI models.
- *Specialized IoT/Edge Gateway:* Niche gateways use SoCs for edge computing (quad A53 ~16k DMIPS + 2.3 TOPS NPU, 2 GB RAM) and can process vision or audio (face/wake-word recognition) locally, common in smart home hubs.

As we move forward, the AI capabilities of the GW edge will continue to advance, driven by demands for higher TOPS, increased memory, and ongoing debates on device integration versus separate solutions for premium services.

We will briefly introduce the AI frameworks proposed for SP CPE edge devices, as well as model creation optimized for smaller edge runtimes and specific silicon environments. General AI models from platforms like HuggingFace.co are not immediately compatible with ARM, Mali GPU, or NPU silicon found in Broadband GW and other SP CPE devices. Silicon providers must supply SDK support, and operators need to enable full access to CPU, GPU, and NPU functions for the models. Currently, TensorFlow Lite is the common framework used across CPE devices, and conversion tools are available to port ONNX and other frameworks to TFLite.

## 3.2. Set-Top Boxes (STBs) and Video Hubs

Modern STBs have evolved beyond video decoding and UI, now featuring AI for voice search, recommendations, and interactive camera-based functions. Built with powerful SoCs—often more advanced than gateways—they include CPUs, GPUs, and NPUs delivering 1.5–9 TOPS. For instance, current STB silicon can have six-core CPUs (~75k DMIPS), a GPU, and a 5 TOPS INT8 NPU, plus up to 8 GB RAM for DVRs. These specs enable real-time CNN tasks like person detection, object recognition, or speech-to-text processing directly on the device.

Mid-range operator-issued STBs, like streaming boxes or DVRs, may lack NPUs but have capable CPUs and GPUs. A quad Cortex-A55 at 1.5 GHz provides enough power to run small neural networks using CPU (with NEON) or GPU (via OpenCL). These devices often handle voice processing for remote control through compact models, either locally or by sending audio to the cloud. With NPU integration becoming standard, mid-level STBs now promote AI features.

As we consider the STB's place in the AI hierarchy, it's worth highlighting its potential to be even more significant than the gateway or router, serving as a "Room AI" device. The STB can open new opportunities for both service providers and consumers. With the rise of Generative AI focused on multimodal interfaces—text, voice, video inputs and outputs—the STB combined with TV, camera, and

FFV meets the requirements for these interactions, positioning it as an effective platform for personal AI assistants.

- The STB serves as the Service Provider's demarcation point, handling both local and cloud AI processing.
- It supports far-field voice recognition, capturing human voices and ambient sounds in the room.
- Its camera interface allows for local and cloud-based video input processing.
- HDMI connection enables the TV to act as the primary screen for video and images.
- The TV also provides audio output and delivers multimodal sound experiences.
- STB's Wi-Fi and BLE capabilities support sensing movement, device location, and add another layer of interaction.
- Through Matter Hub integration, the STB gathers data from IoT devices for enhanced AI inference and actionable knowledge.
- Strategically placed in the home, the STB can function as a central Home AI Hub.

This paper strongly recommends that service providers reconsider relying solely on STBs, highlighting the potential for dedicated in-room devices with human interfaces to access Generative AI. Currently, smartphones provide immediate access to these services, but a fixed device would allow all family members to connect easily and benefit from a large screen. Additionally, camera-based solutions in rooms could offer advantages such as enhanced security, support for aging in place, and improved accessibility for disabled individuals.

To assess the value of the STB and evaluate the cost/ROI of shifting its role from video decoder to in-room AI hub, let's focus on the past five-plus years of STB development. The current and future generations of STB offer a cost-effective entry point for Room AI Hub.

Looking at the STB hardware tiers – 5 year+ in the last 5 years and shipping today :

- *Basic 4K Set-Top (2021):* Quad ARM Cortex-A53 @1.2 GHz (~10k DMIPS), Mali GPU, no NPU, 1-2 GB RAM. Limited AI, mostly sending voice commands to the cloud; handles simple client-side tasks and minimal sensor processing.
- *Advanced AI Set-Top (2024):* New SOCs with up to 4× Cortex-A73 + 2× A53 (~75k DMIPS), 9 TOPS NPU, Mali-G52 GPU, 4–8 GB RAM. Supports local speech-to-text, person detection, chatbots, and fitness coaching via USB webcam, all without cloud reliance. These STBs offer improved video quality and on-device AI features. Upgrading to 8 GB RAM is advised for running larger models.

### 3.3. A Brief Look at What is Happening in Retail

The role of non-Service Provider Retail Home AI is still unclear due to its early stage, but some initial developments are emerging for edge architecture in homes and small businesses. AI sidecar architectures—adapted from mobile, automotive, and security sectors—are now being integrated into CPE broadband, STB, and IoT/camera devices. While cloud inference providers shift focus to edge inferencing, companies like Nvidia and AMD are introducing solutions with performance ranging from 25TOPS to over 1000TOPS, including Nvidia's established Jetson line and the newer DGX Spark, to support local AI processing with larger memory capabilities.

These solutions support independent AI processing in the home, with the goal of enabling large-scale use of humanoid robots. By offloading AI tasks to local robot processing, costs can be reduced, and constant Internet connectivity is not required. A HomeAI server could further optimize compute resources and

reduce battery usage and heat in robots. Advanced vision models will be essential for robots to handle complex household tasks. Over the next five years, these strategies could encourage partnerships between service providers and facilitate the integration of humanoid robots into homes.

- Low latency and prioritized packets/tokens for robot cloud offloading.
- Robots are expected to support both Wi-Fi (6GHz) and 5G/6G, with preference for Wi-Fi and cellular as backup.
- Enable private, offline-capable AI compute and model execution at home.
- Service providers should be able to monitor robot audio/visual data, enforce local data policies, and offer audit or kill switch options.
- Provide value-added monitoring of both Wi-Fi and 5G/6G traffic for robot connectivity.

The key issue is whether retail or service providers will address Home AI processing for shared services like robots. Since significant investments and partnerships are required to run models such as LLMs on shared home compute resources, service provider strategies remain in development for the next five years. Instead of large-scale cloud-based Home AI servers, the current trend is to explore broadband NPU-assisted AI services beyond 2.5 TOPS, which may offer the optimal value for service providers as they expand into emerging Home AI platforms. Further analysis will examine how current models align with service provider involvement.

**Table 5 - Retail Level Home AI Devices/Servers/PC**

| Platform | Jetson AGX Orin 64 GB | DGX Spark | AMD Ryzen AI Max / Strix Halo |
|---|---|---|---|
| AI Performance | 275 TOPS (INT8) | 1 PFLOPS FP4 / 1,000 TOPS | Up to 60 TOPS (Strix Halo APU) |
| GPU | NVIDIA Ampere; 2048 CUDA + 64 Tensor cores | NVIDIA Blackwell GPU; 5th-gen Tensor, 4th-gen RT | RDNA 3.5 GPU with up to 40 CUs |
| CPU | 12-core Arm Cortex-A78AE @2.2 GHz | 20-core Arm (10x Cortex-X925 + 10x Cortex-A725) | Up to 16 Zen 5 cores (Strix Halo); Ryzen AI Max+395 integrated |
| Memory | 64 GB LPDDR5 (256-bit, 204 GB/s) | 128 GB LPDDR5x unified, 256-bit, 273 GB/s | Up to 128 GB unified memory (112 GB GPU alloc.) |
| Storage | 64 GB eMMC 5.1 | 1–4 TB NVMe M.2 SSD (self-encrypted) | Platform-dependent (typically NVMe on desktop/server) |
| Memory Bandwidth | 204 GB/s | 273 GB/s | e.g., LPDDR5X-8000 256-bit (speculative) |
| Form Factor / Power | Module (100 × 87 mm), 15–60 W | Desktop-box (~150 × 150 × 50 mm), power-optimized | APU within desktops/laptops; 55–130 W TDP |
| Unique Features | Dual NVDLA v2, video encode/decode pipelines | Unified GPU/CPU memory, ConnectX-7 Smart NIC, Wi-Fi7 | Hybrid OGA support, huge unified memory, integrated NPU for LLMs |

The NVIDIA Jetson Orin Nano offers around 8 GB RAM and up to 40 TOPS of AI performance in a compact 15 W form factor. Its capabilities make it suitable for offloading vision and speech tasks from cloud to home environments, serving as a model for service providers offering advanced features like multi-camera AI security or local voice assistants. With its 6-core ARM CPU, the Orin Nano can efficiently handle medium-sized models such as EfficientDet or Flan-T5 small, outperforming typical gateway CPUs by supporting concurrent AI data streams. In contrast, lower-spec BB GW and STB devices with embedded NPUs often lack the memory and processing power for simultaneous AI model execution, leading to frequent context switching and slower inference—limitations that diminish as TOPS and DRAM increase.

A mid-range home AI server can offer around 275 TOPS and 64GB RAM, like NVIDIA's Jetson AGX Orin, which packs data-center-grade AI into a compact device. It supports large models and multiple HD video streams. While the $2000 price is high, it matches the cost of a quality PC. Enthusiasts have built rigs with over 1000 TOPS and 128GB RAM to run advanced models locally. This trend may point toward more affordable and commercially available home AI servers soon, possibly through leasing or shared compute models for consumers.

Service Providers have traditionally trailed high-end retail Wi-Fi routers in features and hardware, often using smaller memory sizes and older processors. This approach could also apply to Home AI Server architecture, where providers invest in edge AI hardware and offer services to consumers, emphasizing privacy, lower latency, support, and strong partnerships with LLM and model vendors. In the future, we might see bundled LLM packages for households rather than just individual users.

**Table 6 - Models by Precision to Parameter to Memory Size**

| Precision | Bits per Parameter | Bytes per Parameter | Max Parameters in 64 GB | Model Size (Parameters) |
|---|---|---|---|---|
| FP32 | 32 | 4 | 17 billion | 17B |
| FP16 | 16 | 2 | 34 billion | 34B |
| INT8 | 8 | 1 | 68 billion | 68B |
| INT4 | 4 | 0.5 | 136 billion | 136B |

To summarize this landscape, Table 7 provides a snapshot of representative CPE and edge device classes with their key AI-relevant specs and example capabilities:

**Table 7 - SP CPE Device Types Reference to AI Capabilities**

| Device Type | CPU Performance | NPU (TOPS) | Memory (DRAM) | Example AI Capabilities |
|---|---|---|---|---|
| Basic Broadband Gateway | ~5k DMIPS (dual-core ARM) | 0 (CPU-only ML) | 256–512 MB | Minimal AI (e.g. simple anomaly detection on CPU) |
| High-End Wi-Fi 7 Gateway | ~20–30k DMIPS (quad A7x) | ~1-2.5 TOPS | 2–4 GB | On-device voice, Wi-Fi motion sensing, single-camera security inference |
| Standard 4K Set-Top Box | ~10–15k DMIPS (quad A55) | 0 (or tiny DSP) | 2 GB | Mostly cloud-assisted AI; local voice wake-word; basic recommendations |

| Device Type | CPU Performance | NPU (TOPS) | Memory (DRAM) | Example AI Capabilities |
|---|---|---|---|---|
| AI-Enhanced Set-Top Box | ~75k DMIPS (4×A73+2×A53) | ~2.5-5 TOPS (INT8 NPU) | 4 GB | Edge AI video analytics, local ASR (speech), image recognition, AR features |
| Home AI Sidecar (SP potential first step) | ~40k DMIPS (6×A78) | ~2-50 TOPS (GPU/NPU) | 8+ GB | Runs medium models (e.g. object detection, small language model) in real-time |
| Home AI Server (SP tracking retail version -1) | ~100–200k DMIPS (12-core) | ~275 TOPS (GPU) | 32–64 GB | Runs large models (multi-stream vision, 7B+ LLMs) locally; a quasi- “edge cloud” |

*Service Provider CPE and edge devices (circa 2024) show a range of hardware specs: aggregate CPU DMIPS for all cores, peak INT8 TOPS for NPU/AI (if present). Devices range from basic units without AI acceleration to ad*vanced models matching cloud server AI throughput.

Current-generation CPE devices are rapidly advancing to support effective on-premises AI inference. Mid-level gateways and STBs now offer several TOPS, allowing tasks like face recognition and voice keyword detection without needing the cloud. High-tier devices with 20–40 TOPS are emerging, and for intensive AI workloads, specialized sidecars exceeding 100 TOPS are available.

Thermal and power limits mean typical broadband gateways must be fanless, compact, and use less than 40 W. High-power accelerators aren't practical in these devices, so CPE NPUs focus on efficiency, often using designs from smartphones. ISPs may offer dedicated AI hubs for users needing more power. As semiconductor technology advances, performance per watt is expected to rise dramatically, making widespread deployment of powerful AI in mainstream devices feasible within five years.

With increased use of AI compute in home devices—such as set-top boxes running non-realtime inference, proxy tasks, or training—overall power consumption can rise. If a set-top box processes multimodal generative AI or acts as a gateway proxy, standby and sleep profiles may be affected, leading to longer periods of energy draw. To address this, designers should prioritize efficient CPU, GPU, and especially NPU architectures, keeping NPU and DRAM as the primary powered elements for optimal efficiency.

**Implication for Service Providers:** Home Edge AI hardware is ready; the next step is to match AI models to device tiers based on performance. The following section examines voice, vision, and language inference models and shows how they align with different device capabilities, helping providers deploy the right workloads to the right device for optimal results.

## 4. Home Edge AI Model Analysis (Mapping AI to Device Tiers)

AI models vary widely; a small neural network for a microcontroller is very different from a large cloud-based transformer. For practical home AI deployment, it's helpful to define device capability tiers and link them to suitable models and use cases.

Tier 1: Base Level (Legacy/Entry-Level CPE) – Devices made before 2024 with up to 2GB DRAM, like basic gateways and older STBs, have limited processing power and usually lack NPU acceleration. These devices can only run extremely lightweight or optimized models, often at slower speeds. For example, a Tier 1 router (~1 GB RAM, no NPU) could handle simple tasks like anomaly detection or keyword spotting, but performance and effort versus reward must be carefully considered.

- Optimising GW or STB code may be necessary to free up memory for small CNN models to run on device CPUs.
- Quantising models to 4-bit or 1-bit often causes excessive false positives and negatives, making them impractical.

The current approach is to limit and gradually increase AI model deployment on NPU-enabled CPE edge devices like GW and STB, based on available memory and processing power. Backporting AI models to older devices is not being considered now, which requires maintaining separate firmware versions for AI and non-AI devices. Some features, such as packet inference security and fingerprinting, can run on both device types, offering additional benefits and cost savings for AI-enabled units.

**Tier 2: Emerging AI-Capable CPE –** Devices with embedded NPUs offering ~2.5–8 TOPS and 2–4 GB DRAM are capable edge devices, like high-end STBs or new gateways. These generally have strong CPUs (~20k–50k DMIPS) and can run many advanced models, though not the largest ones in real time. For instance, a Tier 2 gateway with 2.5–5 TOPS and 4 GB RAM can process models up to 100 million parameters (with quantization), supporting tasks like on-device PHY inference or camera object detection. These commonly run CNNs, as transformer models require more resources. There is growing interest in developing compact language models for immediate, on-device support, including setup, offline debugging, and basic troubleshooting.

Tier 3: High-End Home AI (Sidecars/Servers) – Devices offering roughly 25 to 1000+ TOPS and 8–512 GB RAM range from powerful sidecar boxes to advanced home AI servers. These systems deliver near-cloud-level computing locally, featuring high-performance CPUs (100k+ DMIPS, potentially multi-core x86 or many-core ARM). Only exceptionally large models exceed their capabilities; for example, a Tier 3 server with 256 GB RAM and 1000 TOPS can run a 30B-parameter GPT-style model at home, supporting many Service Provider features on the edge instead of in the cloud. High-end options (275–1000+ TOPS) could evolve into future home AI solutions over five years. The distinction between sidecars and dedicated servers may impact cost and performance, and future designs may integrate these components directly into boards or SOCs.

After defining these tiers, we evaluate which AI models a Service Provider might use at home and determine the suitable tiers based on requirements like throughput or real-time performance. The focus is on speech recognition, natural language processing (NLP), and computer vision—key for devices such as voice assistants, support bots, and security cameras. We'll also specify if each model is compute-bound or memory-bound, clarifying whether TOPS or RAM limits device performance.

## 4.1. Examples of Models by Domain and Their Edge Feasibility

### *4.1.1. Speech Recognition & Audio Processing*

Modern speech-to-text models like OpenAI Whisper range from small (39M parameters) to large (1.5B). On low-end devices (Tier 1, ~1 GB RAM, no NPU), only Whisper tiny can run close to real-time for simple tasks; larger versions need more capable hardware (Tier 2+). Very large models require powerful devices or cloud resources. Simple audio models like wake-word detectors are extremely small and suit even microcontrollers. Tier 1 devices can handle basic audio tasks, while Tier 2 supports more robust transcription and classification. Tier 3 can manage full ASR, TTS, or music generation models.

**Table 8 - Public Available AI Speech to Text Models**

| Model Name | Developer | Langs (English / Multilingual) | Model Sizes (Quantized) | Use Case (Tiny → Full) | Notes |
|---|---|---|---|---|---|
| **Whisper** | OpenAI | English / Multi (~100) | Tiny (~39MB), Base (~74MB), Small (~244MB), Medium (~769MB), Large (~1550MB) | Base+ for mobile/edge, Medium+ for cloud | Very high accuracy, slow on CPU |
| **Coqui STT** (ex-Mozilla) | Coqui | English / Multi (limited) | ~50MB – 200MB | Edge to mid-range | Fast, light models, open license |
| **Vosk** | Alpha Cephei | English / Multi (20+) | ~50MB (tiny) to ~1.5GB | Edge to cloud | Offline, C++-based backend, very efficient |
| **Whisper.cpp** | OpenAI Community | English / Multi | Same as Whisper, quantized: Tiny.int8 ~20MB | Tiny to mid | Whisper port to C++, optimized for edge |
| **ESPnet** | JSALT, others | English / Multi | Varies (200MB–1GB) | Academic to cloud | More academic, less edge-focused |
| **Nemo (NeMo)** | NVIDIA | English / Multi | 150MB–>2GB+ | Cloud-focused | Needs GPU, cutting-edge accuracy |
| **Kaldi** | Johns Hopkins | English / Multi | 100MB–>500MB | Research / production | Complex to set up but powerful |

**Table 9- Public Available Wake Word Detection Models**

| Model Name | Developer | Langs (English / Multi) | Model Size | Use Case (Tiny → Full) | Notes |
|---|---|---|---|---|---|
| Porcupine | Picovoice | English / Multi (15+) | ~20–100KB | Ultra tiny edge | Real-time, on-device, <1MB RAM usage |
| Snowboy *(discontinued)* | KITT.AI | English / Multi | ~50–200KB | Legacy tiny edge | Still used in some offline projects |
| WakeNet | ESP-Speech (Tencent) | English / Multi | 200KB–1MB | Mobile & MCU | Very efficient CNN-based |
| Mycroft Precise | Mycroft AI | English / Multi | ~1MB–10MB | Edge-focused | Fully open-source |

| Model Name | Developer | Langs (English / Multi) | Model Size | Use Case (Tiny → Full) | Notes |
|---|---|---|---|---|---|
| Picovoice Leopard *(full ASR + WW)* | Picovoice | English / Multi | ~10MB+ (ASR) | Edge/midrange | ASR + Wakeword combined stack |
| Sensory TrulyHandsfree | Sensory | English / Multi (30+) | ~200KB – 2MB | Commercial embedded | High-accuracy WW, commercial license |
| Rhasspy Wake Word | Rhasspy | English / Multi | 0.5–3MB | Open source, edge | Works with several backends |

**Table 10- Typical Tiers of Device Type for Model**

| Size Tier | Typical RAM Use | STT Models | WW Models | Notes |
|---|---|---|---|---|
| Tiny (MCU/low-end edge) | <1MB | Whisper.cpp tiny-int8, Vosk small | Porcupine, Snowboy, WakeNet | Keyword spotting, simple STT |
| Small (Mobile CPU/NPU) | 1–100MB | Whisper base, Coqui STT, Vosk | Precise, Picovoice | More accurate STT with wakeword |
| Mid (Edge w/ NPU, GPU) | 100MB–700MB | Whisper small, Whisper.cpp small | Leopard, Sensory | Good accuracy, offline ops |
| Large (Cloud) | 700MB–2GB+ | Whisper large, NeMo, ESPnet | Often combined with STT | High accuracy, GPU/TPU inference |

**Table 11- Language Support for Each Model**

| Model | Languages |
|---|---|
| Whisper | 100+ |
| Vosk | ~20 |
| Coqui STT | Mainly English |
| Nemo | Many, model dependent |
| ESPnet | Dozens |
| Porcupine | ~15 |
| WakeNet | Multilingual |
| Precise | Mostly English |
| Snowboy | 10–15 |
| Sensory | 30+ |

### *4.1.2. Natural Language Processing (NLP) and Generative AI*

Language models such as small transformers or RNN-based models with 50M–100M parameters can run on Tier 1 devices if optimized, like Google's Flan-T5 Small (80M parameters), which fits into ~80MB with 8-bit quantization. While speeds may vary, high-end gateway CPUs or NPUs can handle these models. Tier 2 devices (4–8GB RAM) support medium models up to 1B parameters using 4- or 8-bit quantization, enabling deployment of Small Language Models (SLMs) for local generative tasks. SLMs between 100M–300M parameters are ideal for edge, supporting fast, private tasks like dialogue or simple

Q&A without relying on the cloud. Tier 3 devices (16–64GB RAM) can run multi-billion parameter models for advanced on-premises AI, though at higher cost. Further considerations for SLM deployment at the edge versus the cloud will be discussed in Section 5.

The use of Generative AI at the edge is probably one of the most discussed elements of this new push for AI use. It affects a number of things

- Employing hierarchical AI from client (source) to cloud (sink) leverages each processing node, reducing reliance on the cloud, saving latency and cost, and improving the distribution of power needs beyond centralized data centers.
- In this context:
    - Home clients like smartphones and laptops increasingly handle AI tasks themselves—such as Copilot laptops running PHI SLM for screen vision or iPhones prioritizing local AI functions or using Apple's Private Compute Cloud when necessary.
    - Devices with limited memory or processing power may rely on proxies like gateways, AI-enabled set-top boxes, home AI servers, or direct cloud calls.
    - AI-enabled gateways and set-top boxes are categorized by capability tiers.

The debate over running Generative AI models on Edge devices versus relying on the cloud—due to concerns about latency, cost, and privacy—is central to Edge AI. While massive cloud-based models with hundreds of billions or even a trillion parameters can automate many tasks, shifting these functions to the Edge remains challenging.

The main issues are one of memory and compute to retain the accuracy of a Large Language model as its reduced in size and quantized to lower precision to fit the smaller memory and compute environment.

Currently, 8-bit models with 1 billion parameters typically require 1GB DRAM, so devices like GW and STB with 4GB DRAM are limited in model size and capabilities, leading to increased hallucinations compared to large cloud models. These edge models generally lack advanced data parsing, embedding generation, vector database integration, and RAG support. Additionally, their context window is usually restricted to around 2KB, making it challenging to process larger datasets.

**Table 12 - The General Fit of Quantized Language Models to the Memory Size Available.**

| Free DRAM | Model Name | Params | Quant | Est RAM Req | Use Case | Notes |
|---|---|---|---|---|---|---|
| 8 GB | LLaMA 2 7B | 7B | 4-bit | ~6.5–7.5 GB | Chat, summarization | Fast on 4-bit, limited parallel ctx |
| | Phi-2 | 2.7B | 4-bit | ~5–6 GB | Reasoning, code | Efficient, high performance for size |
| | Mistral 7B | 7B | 4-bit | ~7.5 GB | General purpose | Fast attention; fits at limit |
| | Qwen 1.5 4B | 4B | 4-bit | ~5–6 GB | Multi-lingual | Compact & high accuracy |
| | TinyLLaMA 1.1B | 1.1B | 8-bit | ~2.2 GB | Lightweight assistant | Extremely fast |
| 4 GB | Phi-2 | 2.7B | 4-bit | ~3.5–4 GB | Reasoning, education | Ideal balance of size/performance |

| Free DRAM | Model Name | Params | Quant | Est RAM Req | Use Case | Notes |
|---|---|---|---|---|---|---|
| | TinyLLaMA 1.1B | 1.1B | 8-bit | ~2.2 GB | Basic chat, summarization | Very fast, low latency |
| | GPT2-medium | 345M | 8-bit | ~1.5–2 GB | Text generation | Old but functional |
| | DistilGPT2 | 82M | 8-bit | ~1.2 GB | Lightweight tasks | Very fast, low memory usage |
| 2 GB | TinyLLaMA 1.1B | 1.1B | 8-bit | ~2.2 GB | May fit if context <2k | Tight fit, possible with swap/cache |
| | GPT2-small | 124M | 8-bit | ~0.8 GB | Prompt response | Very fast and old |
| | MobileBERT (encoder) | 25M | 8-bit | ~0.6 GB | QA, embeddings | Good for non-gen tasks |
| | DistilBERT | 66M | 8-bit | ~1.0 GB | Classification, QA | Encoder only |
| 1 GB | ALBERT-Tiny/XXS | <20M | 8-bit | ~0.3–0.5 GB | Classify/Embed/Intent | Great for low-power use |
| | GPT2-tiny | 42M | 8-bit | ~0.7 GB | Text generation | Very simple output |
| | DistilGPT2 (tiny) | 33M | 8-bit | ~0.7 GB | Completion, chat | Mini inference tasks |
| | CodeT5-small | ~60M | 8-bit | ~0.8 GB | Code summarization | Transformer for code |

Silicon vendors use compression methods to fit twice the model parameters during quantization. Larger models can also be stored in Flash and loaded into DRAM as needed, a process known as model paging, weight swapping, or demand loading. This approach increases memory capacity at the cost of speed and latency. Rather than loading the entire model into DRAM, the system:

- **Divides the model into segments** (typically layers or tensor blocks).
- Loads only the **needed segments** for the current inference step.
- **Evicts unused weights** to free up DRAM.
- **Streams in** the next required segments from **Flash/NAND storage**.

This is like **virtual memory** for AI models.

**Table 13- Performance Factors in AI Processing**

| Factor | Description |
|---|---|
| Memory Efficiency | Can use models larger than DRAM capacity |
| Lower Quantization | Can keep FP16 or 8-bit weights, improving accuracy |
| Latency Increase | Flash is much slower than DRAM (especially random read from NAND) |
| Throughput Loss | Real-time or multi-threaded inference becomes difficult |
| Power Impact | Frequent Flash access increases energy use |
| Complexity | Requires careful caching logic, layer prefetching, and data layout control |

For Broadband and STB current AI enabled devices – this has not yet typically been implemented as we try and figure out the use cases and the realtime and non-realtime applications.

It's still uncertain whether a local SLM function in the Home Edge is ideal. Unlike large cloud LLMs with vast general knowledge, edge and local models should target specific tasks, considering cost, privacy, and latency—especially since limited hardware struggles with high token throughput compared to powerful cloud clusters. From a service provider's perspective, SLM functions should generate value by saving money or attracting customers, focusing on practical tasks. The challenge is optimizing generative AI SLMs for a narrow set of tasks rather than attempting broad capability, which is limited by fewer parameters and quantization.

Key SLM use cases have emerged with transformer-based Generative AI apps. A major application is deploying Small Language Models on Home Edge devices to quickly address customer support questions across areas like billing, contracts, service quality, education, and equipment. Each area requires tailored strategies for effective closed-loop Edge AI SLM solutions.

Some areas for innovation include

- Clarify bill descriptions and distinguish between bills, contracts, and actual service parameters.
- Enhance Gen AI-driven phone setup apps to guide users through setup steps and track completion before connection.
- Address Wi-Fi issues, from retrieving passwords to identifying the affected service or device via TR181 data analysis.
- Develop a complaint handling solution using a quantized SLM, noting the risk of hallucination if unmanaged.
- Discuss UI design for local SLM interaction.
- Enable smartphone-to-router messaging with escalation to IVR, AI chatbot, or support staff.
- Integrate voice input/output via STT/TTS and avatar-based solutions on smartphones or TVs.

Any solution should also have the ability to escalate to the next level solution for resolution

- Attempt task with local SLM. If unsuccessful, escalate to Cloud LLM; if that fails, consult a human.
  - Additional escalation options include:
    - Using an AI Sidecar at home for advanced tasks,
    - Leveraging a smartphone as a proxy SLM through gateway or set-top box,
    - Utilizing service provider network LLMs in data centers,
    - Considering third-party LLM platforms such as OpenAI, Google, X, Anthropic, Perplexity, and others.

**Table 14- Representative Top 10 Typical Customer Call Types to Support Line**

| # | Reason customers contact support | Evidence and examples |
|---|---|---|
| 1 | Billing disputes and unexpected charges | Billing complaints are the most common telecom support issue. The Canadian CCTS reported over 17,000 billing complaints in 2023–24—a 52% rise—including unexpected charges, price hikes, and missing credits or refunds. Billing accounts for 45% of all issues across wireless, internet, and TV services. Similarly, Ofcom data from the UK shows billing, pricing, and charges make up about 19% of broadband and 21% of mobile complaints, often due to incorrect plan fees, roaming/data charges, or equipment costs. |

| # | Reason customers contact support | Evidence and examples |
|---|---|---|
| 2 | Slow speeds and poor service performance | Customers frequently report slow internet, buffering, and call drops. A 2024 blog highlighted that speed and connectivity issues are top reasons for contacting support. The CCTS notes ongoing complaints about service interruptions, below-expected internet speeds, and dropped calls, with 43% of service-quality issues linked to wireless and 36% to internet services. These problems contribute significantly to quality of service complaints in CCTS data. |
| 3 | Faults, service provisioning and installation issues | Many calls stem from network faults, installation delays, and provisioning issues. Ofcom reports that these account for 37% of broadband and 25% of mobile complaints. Customers frequently face delays or missed deadlines for installations and cancellations—CCTS categorizes these as service delivery issues, often related to repairs or unkept appointments. Major providers regularly receive similar complaints during service activations or repairs. |
| 4 | Dissatisfaction with complaint handling | Many Telecoms complaints stem from unresolved issues. Ofcom reports that "complaints handling" causes 32% of broadband, 29% of mobile, and 43% of pay-TV complaints, often due to repeated calls, long wait times, or poor customer service. |
| 5 | Unclear contract terms or disclosure issues | Customers frequently call to dispute or clarify contract terms and promotions. The CCTS reports a 35% rise in complaints about unclear contracts, often due to missing details. Disclosure issues are the second-most common complaint, including confusion over promotion end dates, discounts, contract length, and automatic price increases. |
| 6 | Equipment setup, faults and repairs | Problems with modems, routers, set-top boxes or mobile devices cause many support calls. The CCTS lists repair issues and appointments and installation, activation or re-activation charges among the top issues across service types. Industry guidance urges ISPs to provide troubleshooting guides and self-service tools because customers frequently contact support about equipment setup and connectivity issues. |
| 7 | Service changes (upgrades/downgrades) | Customers often contact support to upgrade, downgrade, or adjust their services, such as adding TV channels, changing broadband speed, or modifying data plans. Service Provider agents help with these changes and plan selection, while CCTS data show contract changes are a frequent concern. |
| 8 | Cancellation and switching providers | In competitive markets, customers often try to cancel services or switch providers but encounter obstacles. The CCTS reports a 47% rise in complaints about cancellation difficulties and more issues with service transfers and termination fees. Table 6.5 highlights these as top |

| # | Reason customers contact support | Evidence and examples |
|---|---|---|
| | | concerns for local phone services, and regulators note consumers commonly face delays and extra charges when switching. |
| 9 | Account information and product inquiries | Some calls are for information on plans, coverage, or promotions. Studies show customers often contact telecom providers to clarify options, ask about coverage, upgrades, bundles, or when planning a move or new device. |
| 10 | Credit or refund not received / financial adjustments | Many billing calls relate to missing credits or refunds, misapplied payments, or refund requests after outages. The CCTS's leading billing issues are credits or refunds not received (3,670 cases) and misapplied payments. Providers like AT&T and Vodafone encounter similar concerns when promotional credits or rebates are delayed. |

To create local software and SLM applications that engage with consumers on these 10 topics, there remain significant gaps compared to a Cloud LMM pipeline. For instance, a local SLM in a gateway cannot process a consumer's bill from a PDF, since it lacks direct upload capability. Document uploads to language models require helper applications, so even this basic PDF use case needs additional support services.

**End-to-End Workflow: PDF Upload and Q&A in LLM Apps**

- **File Upload**
  - User uploads a PDF through the interface.
  - The file is stored temporarily in a secure backend environment (e.g., cloud storage or local memory).
- **Parsing and Text Extraction**
  - A PDF parser (like pdfplumber, PyMuPDF, or Adobe PDF Services) extracts text, metadata, and structure (e.g., headings, tables).
  - Pages or sections are typically split into manageable text chunks (e.g., 500-1000 tokens per chunk), preserving context with overlap.
- **Text Chunking & Preprocessing**
  - Text chunks are cleaned and normalized:
  - Remove headers/footers
  - Fix Unicode or encoding issues
  - Maintain semantic structure (title, section, paragraphs)
- **Embedding Generation**
  - Each text chunk is passed through a pretrained embedding model (e.g., OpenAI's text-embedding-3-small, Sentence-BERT, or Cohere).
  - These embeddings convert text into high-dimensional vector representations that capture semantic meaning.
- **Vector Storage**
  - Embeddings and corresponding text chunks are stored in a vector database (e.g., FAISS, Pinecone, Weaviate, or Qdrant).
  - Each chunk is indexed for fast similarity search.
- **Retrieval (when a question is asked)**

  - User submits a question.
  - The question is embedded using the same embedding model.
  - A similarity search (cosine or dot product) is performed against the vector DB to retrieve the most semantically relevant chunks (top-k matches).
- **RAG (Retrieval-Augmented Generation)**
  - The retrieved text chunks are passed as context to the LLM prompt.
  - The LLM (e.g., GPT-4, Claude, Mistral) generates an answer grounded in the retrieved context.
- **Response & Traceability**
  - The generated answer is returned to the user, often with:
  - References to page numbers or sections
  - Optionally, citations or links to the original chunks

A Home Edge device will not have these helper resources available – or will have to be developed to the workflow… (Performance/Memory/Capacity will be an issue. The Limitations are summed up below

**Table 15- Limitations to Doing SLM Locally on Edge Device**

| Limitation | Impact |
|---|---|
| Low RAM (<8GB) | Insufficient memory to load embedding models or store multiple document chunks for semantic search. |
| Low Compute/TOPS | Slower inference and embedding generation; unsuitable for real-time RAG with large models. |
| No fast disk/SSD | Embedding search requires fast access to vector data; may not be feasible without NVMe/SSD. |
| No persistent DB | Edge systems often lack embedded vector DBs optimized for retrieval tasks. |
| No large LLM | Even distilled models (e.g., Mistral-7B) require >16GB RAM and GPU/TPU for fast performance. |

On a constrained edge device, this entire workflow becomes impractical without heavy optimization:

- Lightweight models such as TinyBERT or MobileBERT would be required.
- Vector search functionality must either be compressed or transferred to another system.
- Due to limited context memory—commonly less than 2KB on typical edge devices—answer accuracy may suffer as adequate contextual information cannot be retained.

So, when trying to get a Small Model fine tuned for specific Generative AI tasks there are typically 4 approaches that are being promoted

#### *4.1.2.1. Techniques to Try and Finetune SLM for Service Provider Specific Tasks*

- **Model Compression:** Reduces large LLM size while maintaining performance.
- **Pruning:** Removes nonessential weights or neurons (e.g., near-zero values, inactive heads) with minimal accuracy loss.

- **Knowledge Distillation:** Trains a smaller model ("student") to mimic a larger “teacher” model’s output, transferring capabilities efficiently.
- **Layer Fusion & Weight Sharing:** Merges layers or shares weights to cut redundancy, resulting in compact models that perform well on specific tasks.
  - **Quantization:** Lowers bit precision for weights/activations (e.g., 32-bit to 8/4/2-bit), decreasing model size and speeding up inference.
- **PTQ (Post-training Quantization):** Applies quantization after training for quick size reduction with little accuracy drop.
- **QAT (Quantization-Aware Training):** Optimizes the model during training for low-precision deployment, ideal for resource-limited devices.
- **Efficient Attention & Sparse Architectures:** Uses approximations like Linformer, Performer, Longformer, or sparse attention to lower memory and compute requirements, especially important for SLMs.
  - **Mixture of Experts:** Routes inputs through select sub-models, cutting computation but boosting capacity.
  - **Low-Rank Adaptation (LoRA):** Adds small trainable matrices for efficient fine-tuning on new tasks.
- **Hybrid Edge + Cloud Execution:** Combines lightweight on-device SLMs for simple tasks with cloud-based LLMs for complex queries, improving latency, cost, privacy, and real-time performance.

By combining these techniques, developers can:

- Compress and distill large models into compact, efficient SLMs,
- Fine-tune them on domain-specific data,
- Deploy them flexibly across heterogeneous hardware,
- And still preserve generative capabilities for targeted use cases.
- Training the model for specialised expertise requires investment and tailored datasets to adjust its weights and biases for deeper capabilities in specific areas, rather than broad generalisation.

**Table 16- Toolkits for Optimizing Small Language Models**

| Toolkit | Purpose | Common Use Case |
|---|---|---|
| Hugging Face Transformers | General-purpose NLP toolkit for training/fine-tuning LLMs and SLMs | Pretraining, fine-tuning, LoRA, distillation |
| PEFT (Hugging Face) | Lightweight fine-tuning (e.g., LoRA, prefix tuning) | Domain-specific SLM fine-tuning with low resource cost |
| LoRA (Low-Rank Adaptation) | Adapter-based fine-tuning via rank decomposition | Parameter-efficient tuning of base models on task-specific data |
| BitsAndBytes | 8-bit and 4-bit quantization + memory-efficient fine-tuning | Quantized inference & training of LLMs on limited hardware |
| Optimum (Hugging Face) | Model optimization and quantization bridge with ONNX/TensorRT | Convert/fine-tune models to run faster on edge/cloud HW |
| ONNX Runtime | Cross-platform inference engine with quantization and acceleration | Export and deploy fine-tuned models on various devices |
| SparseML | Sparse training, pruning, and quantization | Structured pruning for smaller, faster models |

| Toolkit | Purpose | Common Use Case |
|---|---|---|
| Intel Neural Compressor | Quantization-aware training and post-training quantization tools | Compressing models for Intel CPUs/accelerators |
| TensorFlow Lite | Mobile/embedded deployment and quantization tools for TensorFlow models | Running fine-tuned models on phones, microcontrollers |
| PyTorch Mobile | Mobile inference for PyTorch models | Deploy SLMs on Android/iOS apps |
| OpenVINO | Intel toolkit for optimizing and running models at the edge | Running optimized models on edge devices and IoT platforms |
| DeepSpeed | Scalable and efficient model training/fine-tuning (Microsoft) | Fine-tuning large-scale transformer models efficiently |
| FSDP (Fully Sharded Data Parallel) | PyTorch technique for memory-efficient large model training | Fine-tuning large models on limited multi-GPU setups |

Here is a basic process for example to get a small language model to be more of an expert in RSSI signal strength at a very small Parameter size 1Bn and highly quantized..

To fine-tune a small 8-bit 1B-parameter LLaMA model for accurately understanding technical concepts like RSSI, MCS, and Wi-Fi frequency ranges, more is needed than parameter tuning alone. Here's why it may underperform and how to fine-tune it effectively:

**Table 17- Typical Limitations to Fine Tuning Success**

| Limitation | Description |
|---|---|
| Underparameterization | 1B params = too small to encode broad general + niche technical knowledge. |
| Quantization error | 8-bit models can lose semantic precision during inference. |
| No domain pretraining | The base model likely hasn't seen much about Wi-Fi, RSSI, or MCS during training. |
| Contextual misunderstanding | Concepts like "-98dBm is weaker than -50dBm" require numerical reasoning and domain logic. |

**An example of what is required to fine tune a base SLM to be more proficient at understanding Wi-Fi questions**

- **Curate a High-Quality Domain Dataset (Golden Dataset Preferred)**
- Using following approaches and sources
- Manually labeled examples (Q&A or instructional format)
- Extracted from RFCs, textbooks, support documentation
- Includes reasoning patterns, not just facts

Example data entries:

- Q: Is -98 dBm a stronger or weaker signal than -50 dBm?
- A: -98 dBm is a much weaker signal. In RSSI, higher values (closer to 0) indicate stronger signals.
- Q: What MCS index corresponds to the highest data rate in 802.11ac?

- A: MCS index 9 with 160 MHz channel width and 4 spatial streams gives the highest data rate.

And then aim for :

- At least **5k–50k** diverse Q&A examples for a good initial fine-tune
- Optionally: structured examples like markdown charts, RF heatmaps (if the model supports images), formulas

**Use a Fine-Tuning Method Fit for SLMs**

- **LoRA** or **QLoRA**: lightweight, efficient, and works well even with quantized models
- **PEFT via Hugging Face**: great support for adapters and efficient training
- Use **gradient checkpointing**, **grouped attention**, or **bitsandbytes** if needed for memory constraints

**Table 18- Frameworks and Tools for Finetuning**

| Task | Recommended Toolkits |
|---|---|
| Fine-tuning setup | Hugging Face Transformers + PEFT + LoRA |
| Quantized training | QLoRA (BitsAndBytes 4/8-bit) |
| Dataset formatting | LangChain, OpenAI Datasets, Textbooks |

**Evaluate with Targeted Test Sets**

- Create unit tests to ask things like:
  - "Which RSSI value is better: -60 or -80?"
  - "What is MCS and how does it affect throughput?"
- Use these to measure **accuracy improvement** vs. the base model

**Consider Prompt Engineering or External Retrieval**

- If fine-tuning isn't feasible or effective:
  - Use **RAG** (Retrieval-Augmented Generation): combine the model with an indexed Wi-Fi knowledge base
- Structure prompts to guide reasoning:
  - "Based on typical Wi-Fi RSSI interpretation, is -98 dBm stronger than -50 dBm?"

In conclusion, attaining the target accuracy with SLM requires focused investment, advancements in computational power and DRAM within CPE devices, the implementation of local helper functions for data processing, and broader context support. Although efforts are ongoing to increase the capabilities of smaller models, consumer adoption will depend on sustained, demonstrable improvements over time.

**Table 19- Simple Steps in Finetuning Process**

| Step | Description |
|---|---|
| Collect Data | Curated Q&A about RSSI, MCS, bands, protocols, SNR, PHY modes |
| Fine-tune Efficiently | Use LoRA or QLoRA with a PEFT-compatible framework |
| Evaluate | Custom test prompts, compare to expected answers |
| Optional RAG | Add external retrieval from technical docs for better coverage |
| Iterate | Monitor failure cases and retrain on new examples |

### *4.1.3. Computer Vision (CV)*

This category includes image classification, object detection (e.g., YOLO, SSD), facial recognition, and pose estimation. While CNNs dominate these tasks, newer models use transformers or CNN/Transformer hybrids like vision transformers. Efficient CNNs such as MobileNet and EfficientNet are ideal for low-power edge devices, outperforming heavier models like ResNet. Lightweight object detectors like YOLOv4-tiny and SSD MobileNet can run at several FPS on 2–5 TOPS hardware, making them suitable even for older devices with no NPU at low resolutions. Adding a basic NPU enables real-time detection at moderate resolutions. Face recognition uses lightweight CNNs (e.g., MobileFaceNet) for small databases on Tier 2 CPEs. Pose estimation is more demanding; compact models like MoveNet may work on Tier 2, but advanced versions require Tier 3 CPEs or cloud resources.

Video analytics that process every frame (such as monitoring several security cameras) are typically compute-bound and rely on TOPS for scaling. Tier 2 CPE may be insufficient for multi-camera setups, as running multiple YOLO detectors can quickly exceed 25 TOPS. For these scenarios, Tier 3 or cloud offload is preferable. Hardware requirements for specific use cases are summarized in Table 2 (Section 5).

When assessing model feasibility, it's crucial to know if workloads are compute-bound or memory-bound. Compute-bound tasks, like most image CNNs, depend mainly on processing power (TOPS/DMIPS); more TOPS improves speed. Memory-bound models, such as large language models, require sufficient RAM—if the model doesn't fit, it won't run regardless of extra compute. More RAM allows bigger models, while more TOPS only helps compute-bound workloads.

A small LLM avatar with a 7B parameter model may only need a few TOPS for inference, but its 4–8GB size makes memory the limiting factor on Tier 2 devices. In contrast, HD video object detection pipelines can fit in 1GB, but without enough TOPS, real-time processing at 30 fps isn't possible, making them compute-bound.

Its important to know that there are typically 2 video sources where Computer Vision AI models can be applied

- Camera-based video input for understanding the scene, including body pose and facial expressions, often used in security or room monitoring.
- Inline video streams to STB decoder enable AI inference directly within secure video pipelines for various applications.
- Shoppable TV: instantly purchase visible products.
- Sports tracking features.

- Enhanced video and image quality, now available as a direct inline feature.
- Object detection for app functions (e.g., auto record or pause when specific objects appear).

Below is a practical, edge-focused catalog of **popular vision models** that run well on **2.5, 10, and ~50 TOPS** NPUs within **2–8 GB DRAM**. They are grouped by task from simplest (person detection) to deeper scene understanding (segmentation, tracking, action, open-vocabulary). Throughput numbers are **single-stream, batch=1, INT8** estimates with a conservative **~35–50% effective utilization** and include typical pre/post overhead. Use them as *planning ranges*; real results vary by compiler, kernel fusion, and I/O.

**Table 20- Edge Vision Model Chooser**

| Task | Model (popular edge variant) | Res. (px) | Ops/frame (≈) | Est. RAM (≈) | What it's good for | FPS @ 2.5 / 10 / 50 TOPS |
|---|---|---|---|---|---|---|
| Person-only detection | NanoDet-Plus (person class) | 416 | 1–2 GF | 150–300 MB | Very fast single-class person detector; tiny anchors | 60 / 120+ / 200+ |
| | PP-PicoDet-L (person class) | 416 | 1–2 GF | 200–350 MB | Stable latency; PaddleLite backends are strong | 50 / 100+ / 180+ |
| | YOLOv5-n (pruned) (person class) | 512 | 2–3 GF | 250–400 MB | Good balance of recall/precision | 35 / 90 / 160 |
| General object detection | YOLOv8-n | 640 | 3–5 GF | 300–500 MB | Best all-around tiny detector; modern heads | 25 / 70 / 140 |
| | YOLOv7-tiny | 640 | 5–7 GF | 350–600 MB | Mature kernels on many NPUs | 18 / 50 / 110 |
| | EfficientDet-D0 | 512 | 2–4 GF | 300–500 MB | Efficient + TFLite friendly | 30 / 75 / 150 |
| | PP-YOLOE-s | 640 | 8–12 GF | 450–700 MB | Strong accuracy in small size | 12 / 35 / 85 |
| | RT-DETR-R18 (quant.) | 640 | 25–35 GF | 0.8–1.2 GB | One-stage DETR; solid latency at 10–50 TOPS | 4–6 / 15–20 / 45–55 |
| Instance segmentation (objects "extracted") | YOLOv8-n-seg | 640 | 6–9 GF | 450–700 MB | Fast mask heads; great for "cut-out" objects | 12–18 / 35–50 / 90–110 |
| | Mask R-CNN (MobileNetV2) | 640 | 25–40 GF | 1.0–1.6 GB | Classic 2-stage; heavier but accurate masks | 3–5 / 12–18 / 40–50 |
| | YOLACT-R50 (pruned/INT8) | 550 | 15–25 GF | 0.8–1.3 GB | Simple fast instance seg; robust | 6–8 / 20–28 / 55–70 |
| Pose (keypoints) | MoveNet Lightning | 256 | 0.5–0.8 GF | 120–220 MB | Super low-latency single-person | 120+ / 200+ / 300+ |
| | MoveNet Thunder | 256 | 1–2 GF | 180–300 MB | Higher accuracy; still very fast | 70 / 120+ / 250+ |
| | YOLOv8-n-pose | 640 | 6–8 GF | 450–700 MB | Multi-person pose via one-stage | 12–18 / 35–50 / 90–110 |
| Multi-object tracking (MOT) | ByteTrack (+ detector) | — | +0.1 GF | +<100 MB | Tracker is cheap; throughput ≈ detector FPS | ≈ detector FPS |
| | DeepSORT (+ small ReID, e.g., OSNet-x0.25) | 128 crops | +0.2–0.5 GF | +150–250 MB | Better ID stability; small extra cost | ≈ detector FPS – 5–10% |
| Re-ID / attributes | OSNet-x0.25 (INT8) | 128 crop | 0.2–0.3 GF | 100–200 MB | Person ID across cameras; attr heads easy | Per-ROI: 1–2 ms @10 TOPS |
| Semantic segmentation | DeepLabV3+ (MobileNetV3) | 512 | 8–12 GF | 500–800 MB | Room/zones, road/sidewalk, etc. | 10–15 / 30–40 / 80–95 |
| | BiSeNetV2 | 512 | 5–8 GF | 400–650 MB | Very fast light semantic seg | 15–25 / 45–60 / 110–130 |
| | Fast-SCNN | 512 | 3–5 GF | 350–550 MB | Extreme speed; lower accuracy | 25–35 / 70–90 / 150+ |

| Task | Model (popular edge variant) | Res. (px) | Ops/frame (≈) | Est. RAM (≈) | What it's good for | FPS @ 2.5 / 10 / 50 TOPS |
|---|---|---|---|---|---|---|
| Depth estimation | MiDaS small / DPT-small | 256–384 | 3–6 GF | 350–600 MB | Monocular depth for AR/analytics | 18–30 / 50–80 / 120–150 |
| | Depth-Anything-Small | 384 | 5–8 GF | 400–700 MB | Sharper edges than MiDaS small | 15–25 / 45–60 / 110–130 |
| OCR (scene text) | PP-OCRv3 (mobile) | 640 long side | 3–5 GF | 350–600 MB | DB text det + CRNN recog; pipelines well | 20–30 / 60–80 / 140+ |
| Action recognition | TSM-MobileNetV2 (8–16f @224) | 224 | 10–18 GF | 500–800 MB | Temporal activities (fall, fight) | 8–12 / 25–35 / 70–90 |
| | X3D-S / M (quant.) | 224 | 12–25 GF | 600 MB–1.2 GB | Accurate 3D CNN; still edge-friendly | 6–10 / 20–30 / 60–80 |
| Open-vocabulary & "deeper" understanding | MobileCLIP-S (image encoder only) | 224–336 | 1–3 GF | 250–450 MB | Text-image embeddings for labels/search | 40–80 / 120+ / 200+ |
| | GroundingDINO-Tiny (ROI-only) | 640 (ROI) | 30–50 GF | 1.0–1.8 GB | Text-prompted detection on events/ROIs | (event-driven) 2–4 / 8–12 / 25–35 |
| | FastSAM / MobileSAM (ROI-only) | 512 (ROI) | 8–15 GF | 500–900 MB | Fast generic "cut out" of prompted object | 10–18 / 30–45 / 80–100 |

***Why do FPS ranges vary?*** *NPU utilization, I/O formats (NV12 vs RGB), and compiler/kernel maturity (TensorRT, QNN, TIM-VX, TFLite-delegates, OpenVINO) all contribute. The numbers assume INT8 on the NPU and m*inimal CPU impact from pre/post processing.

Summarizing TOPS/DRAM performance and size for Vision and video AI at the Home Edge yields the following application and performance categories, which can serve as a guideline for developers.

**2.5 TOPS, 2–4 GB DRAM**

- *Sweet spot:* NanoDet-Plus / PP-PicoDet (person-only) at **1080p 20–30 FPS**, or YOLOv8-n at **720p 25–35 FPS** (1080p ~10–18 FPS).
- *Segmentation:* YOLOv8-n-seg at **720p ~12–18 FPS**.
- *Pose & tracking:* MoveNet Lightning/Thunder at **>100 FPS** on crops; ByteTrack with detector at detector FPS.
- *Pipelines:* Run detector at 10–15 Hz and track at 30 Hz to save compute; trigger segmentation/pose on ROIs only.

**10 TOPS, 4–8 GB DRAM**

- *Detectors:* YOLOv8-n at **1080p 30–45 FPS**, YOLOv8-s / PP-YOLOE-s at **1080p 20–35 FPS**.
- *Segmentation:* YOLOv8-n-seg **1080p 30–40 FPS**.
- *DETR-style:* RT-DETR-R18 **1080p ~15–20 FPS** (better long-tail categories).
- *Add-ons:* ReID (OSNet-x0.25) at negligible overhead; MiDaS-small/Depth-Anything-S at **30–60 FPS** 384p; PP-OCRv3 **60–80 FPS** on text crops.
- *Use-case:* Multi-task 1×1080p@30 with detection+tracking+pose on events, or **2×1080p@15–20 FPS** detection streams.

**~50 TOPS, 8 GB DRAM**

- *Detectors:* YOLOv8-s **1080p 60–90 FPS** (or 4× 1080p@30 across streams).
- *Segmentation:* YOLOv8-n-seg **90–110 FPS**; Mask R-CNN MobileNet **40–50 FPS** at 640.
- *"Deeper understanding" add-ons:* Event-driven GroundingDINO-Tiny + FastSAM on ROIs; MobileCLIP for embeddings/semantic search.

- *Use-case:* **3–4 streams of 1080p@30** detection+tracking, with on-event instance masks, OCR, depth, and action heads without dropping frames.

Defining use cases for vision models, especially camera-based home edge solutions—reveals key capabilities, mainly in camera inference. Expanding the concept of using a set-top box as a multimodal AI interface, we can envision creating a virtual assistant presence in the living room to enhance user experience.

Rather than interacting with artificial intelligence through a plastic or handheld device,

- The television serves as an analogue for a human being.
- A far-field microphone installed on the set-top box connected to the television functions as the 'ears.'
- The screen is utilised to visually represent the human, drawing upon familiarity from video conferencing engagements.
- An avatar is employed to directly interact with individuals in the room.
- The speaker on the television or set-top box provides the 'voice' for the avatar.
- The camera acts as the eyes of the avatar.
- Accordingly, the camera assumes the role of the avatar's vision and must determine the following:
    - Is there a person present?
    - What is their identity?
    - Are multiple individuals present?
    - What emotions are being displayed?
    - What is the orientation of their body and limbs?
    - Are they preparing to speak?
    - Are they gesturing?

The Vision models described here – all allow the above questions to be answered to provide context to the AI Human Avatar engagement – trying to simulate what a Human would be doing – for example

- If the vision model detects you're not looking at the TV, Avatar, or Camera, the Avatar listens but doesn't respond.
- Direct eye contact with the TV, Avatar, or Camera tells the Avatar to engage in conversation.
- The system evaluates emotion and body language for more natural interaction.

So, in order to create a solution like this on different levels of STB performance using Computer Vision models… Here are some recommendations.

**Recommended *Home-Edge* pipelines (examples)**

- **Basic person detection (doorway / privacy-first) – 2.5 TOPS, 2–4 GB**
    - Detector: *NanoDet-Plus (person-only) 416p @ 30–60 FPS*.
    - Tracker: *ByteTrack* (cheap).
    - Optional: *MoveNet Lightning* on ROIs for posture cues.
    - Tips: Run detector at 10–15 Hz, tracker at 30 Hz. Encode only motion/IDs, not raw frames, for privacy.
- **General security + packages/pets – 10 TOPS, 4–8 GB**
    - Detector: *YOLOv8-n 640p* (or *-s* for accuracy) @ 1080p 30–40 FPS.
    - OCR on ROIs (labels, license plates): *PP-OCRv3*.
    - ReID: *OSNet-x0.25* to avoid double-alerts.

    - Optional: *YOLOv8-n-seg* for clean object cut-outs when alerts fire.
- **Home automation with zones/semantics – 10–50 TOPS**
    - Semantic seg: *BiSeNetV2 (512p)* for room/zone masks @ 30–60 FPS.
    - Detector: *YOLOv8-n/s* for dynamic objects.
    - Logic: "If person in 'kitchen-zone' after 10 pm then …".
- **Deeper understanding / on-demand queries – 50 TOPS**
    - Core detector: *RT-DETR-R18* (accuracy on uncommon classes).
    - Text-prompted refine: *GroundingDINO-Tiny* on event ROIs ("find the red toolbox").
    - Cut-out: *FastSAM* on the grounded box to extract the object.
    - Embeddings: *MobileCLIP-S* to tag/store features for later retrieval ("show me when a stroller appeared").
- **Fall detection / actions – 10–50 TOPS**
    - Pose: *YOLOv8-n-pose* or *MoveNet Thunder*.
    - Temporal head: *TSM-MobileNetV2* (8–16 frames).
    - Detector frequency can be **5–10 Hz** with tracker bridging frames.

**Implementation notes & rules of thumb**

- **Quantize early.** INT8 post-training quantization (with calibration sets) typically delivers **2–4×** throughput over FP16 with minimal accuracy loss for the above models. Use per-channel weight and per-tensor activation schemes if your NPU supports it.
- **Keep inputs NV12/YUV.** Avoid repeated color-space conversions; many camera pipelines deliver NV12. Convert only once if the NPU requires RGB.
- **ROI cascade = free FPS.** Detect at lower res (e.g., 640/720p), then run heavier heads (pose/seg/OVD) *only on crops or on a downsampled pyramid*.
- **Track to save compute.** Run detector at 10–15 Hz; run the tracker every frame to maintain IDs at 30 Hz; refresh detector on occlusions.
- **Memory budgeting.** With INT8 weights, tiny/small models typically keep **<1 GB peak** (incl. activations). Heavier two-stage/transformer models can hit **1–2 GB** peaks at 640–960. On **2 GB devices**, stick to nano/tiny + ROI cascades; **4–8 GB** comfortably runs small/seg/pose stacks.
- **Streams math (rough):** For *YOLOv8-n @1080p* on a mature 10 TOPS INT8 pipeline you can expect about **30–40 FPS** for a single stream, or **2×1080p@15–20**. At ~50 TOPS, **3–4×1080p@30** with room for pose/seg on events.

### *4.1.4. Speech Processing AI Models*

Another area especially with the rise of FFV microphones for multi modal inputs as well as the ability to infer what is happening with sound and noise.

Looking at **AI models and approaches for various voice/audio processing tasks** that are capable of running within **embedded edge compute environments** ranging from **2.5 TOPS to 50 TOPS** and with **2GB to 8GB DRAM** – these are the models that can be leveraged.

**Table 21- AI Models for Voice and Audio Processing (Edge-Deployable)**

| Task | Example Models / Architectures | Min TOPS Required | DRAM Range | Notes / Deployment Feasibility |
|---|---|---|---|---|
| Speech-to-Text (ASR) | - Whisper Tiny/Medium-DeepSpeech (Mozilla)- | 2.5–25 TOPS | 2–6 GB | Whisper Tiny can run around 2.5–5 TOPS. Larger models |

| Task | Example Models / Architectures | Min TOPS Required | DRAM Range | Notes / Deployment Feasibility |
|---|---|---|---|---|
| | Wav2Vec2 (Quantized)- NVIDIA Riva (optimized pipelines) | | | like Base or Medium require 10–25 TOPS. Quantized or distilled versions help. |
| Text-to-Speech (TTS) | - FastSpeech2- ESPnet TTS- Coqui TTS- Google Tacotron 2 (quantized)- NVIDIA Riva TTS | 2.5–10 TOPS | 2–4 GB | FastSpeech2 is real-time on edge. Requires high-quality vocoder (HiFi-GAN) for natural output. |
| Noise Suppression | - RNNoise (Xilinx)- DeepFilterNet- Microsoft Deep Noise Suppression- NVIDIA Riva Denoiser | 1–5 TOPS | 1–2 GB | RNNoise is ultra-lightweight, DeepFilterNet works in real-time. Easily fits in sub-10 TOPS. |
| Speaker Diarization | - ECAPA-TDNN (via SpeechBrain)- ResNet x-vector- pyannote-audio (simplified) | 10–25 TOPS | 4–6 GB | Partial support in low-memory if inference steps are pipelined. Speaker embedding can be optimized. |
| Speech Enhancement (Volume, Clarity) | - SEGAN- MetricGAN- DeepFilterNet2 | 2.5–10 TOPS | 2–4 GB | Improves SNR and clarity. SEGAN variants can be run in real-time at edge. |
| Hearing Assistance (Frequency Shaping) | - Wav2Letter + Filter Bank- Neural Bandwidth Extension- DeepSpeech + Auditory Enhancement | 2.5–10 TOPS | 2–4 GB | Hearing aid-class tasks with optimized filters or low-power DNNs. |
| Audio Event Detection("What is happening") | - YAMNet (MobileNet)- PANNs (CNN14)- VGGish | 2.5–10 TOPS | 2–4 GB | YAMNet is lightweight and classifies ~500 audio events. Well-suited for edge boards. |
| Speaker Separation (e.g., Voice vs. Background) | - Conv-TasNet- Demucs- DCCRNet | 10–25 TOPS | 4–6 GB | Streaming versions exist for on-device separation; Demucs is demanding but can be pruned. |
| Wake Word Detection | - Porcupine by Picovoice- Snowboy- EdgeImpulse keyword spotting- TinyML CNN models | 0.5–2 TOPS | <1 GB | Designed for ultra-low-power chips; typically run on MCU-class devices. |

**Table 22- Hardware Guidance by TOPS & DRAM Envelope**

| Category | Suitable Models |
|---|---|
| 2.5 TOPS / 2GB RAM | RNNoise, FastSpeech2 (light), Whisper Tiny, YAMNet, Porcupine |
| 10 TOPS / 4GB RAM | DeepSpeech, FastSpeech2, Demucs (quantized), DeepFilterNet2 |
| 25 TOPS / 6GB RAM | Whisper Base, VGGish, ECAPA-TDNN, Conv-TasNet |
| 50 TOPS / 8GB RAM | Whisper Medium, Coqui TTS full stack, Speaker Diarization full |

Service Providers should adapt to evolving multimodal inputs in Generative AI and seek ways to integrate these innovations into their operations. While the Set Top Box remains common in homes, video services are declining, especially among U.S. providers, as consumers shift toward retail devices like streaming boxes, Smart TVs, computers, phones, and tablets for video viewing.

Generative AI user interfaces are rapidly evolving, with significant investment in Voice, Video, and Sound UI for multimodal interaction. Emerging smart glasses offer continuous visual and audio input, unlike smartphones which are restricted by their design and placement. OpenAI's acquisition of Love.io hints at a keyboard-free device using audio and video for everyday AI interactions. By mid-2026, more details are expected, but service providers can already utilise set-top boxes—centrally located in homes—as multimodal, environment-aware devices that provide access to AI services.

- Far Field Microphone in the device – low cost addition
- The TV screen as the multimodal output for Video
- The TV speaker as the multimodal output for Audio (The STB can contain a speaker to be independent of the TV status)

Hands-free interaction with local SLMs or partnered cloud LLMs is now possible and cost-effective. Integrating a camera with privacy features into the living room allows for a comprehensive AI input/output solution managed by the service provider. The set-top box can handle functions like Speech-to-Text and Text-to-Audio locally, keeping user data private and reducing cloud dependence. Local processing of camera apps improves privacy and lowers costs, while using local models for body pose, emotion, and facial recognition means only minimal, anonymized data is sent externally.

With current Azure and GCP pricing, processing an average of 15 seconds of speech daily suggests local STB could be cost-effective. Privacy also remains a key consideration in this choice.

**Table 23- Potential Cost Savings of Edge AI Speech to Text**

| Scenario | Cloud Cost / Month | Local STB Cost | Savings / Month |
|---|---|---|---|
| 5 Requests/Day | $0.23–$0.90 | ~$0 (one-time setup) | $0.23–$0.90 |
| 10 Requests/Day | $0.45–$1.80 | ~$0 | $0.45–$1.80 |
| 30 Requests/Day | $1.35–$5.40 | ~$0 | $1.35–$5.40 |
| 50 Requests/Day | $2.25–$9.00 | ~$0 | $2.25–$9.00 |

### *4.1.5. Physical Layer AI Inference*

A primary use case for Edge AI among service providers involves leveraging Physical Layer AI models to address network quality challenges and optimize radio and antenna configurations. These models are applicable across various access technologies, including **DOCSIS RF**, **PON**, **5G FWA**, and **Wi-Fi RF**, provided that digitized signal telemetry—such as FFT, iFFT, IQ samples, or derived RF metadata—is available.

**Table 24- AI Models for PHY Analysis and Optimization at the Edge**

| Application Area | Model Types / Architectures | Typical Input Data | Description / Function | Edge Feasibility (TOPS/Memory) |
|---|---|---|---|---|
| PHY Signal Quality Classification | 1D-CNN- RNN / LSTM- SVM (shallow models)- Transformer-lite | Time/freq series (e.g., SNR, CINR, BER, RSSI, latency, FFT data) | Classify signal quality (e.g., "good", "degraded", "error-prone") for DOCSIS/PON/5G/Wi-Fi | 2.5–10 TOPS, 1–2GB DRAM |
| Noise / Interference Detection | 2D-CNNs on spectrograms- Autoencoders (anomaly detection)- GAN-based denoising (lite)- Isolation Forest (for structured RF stats) | FFTs, spectrogram heatmaps, constellation maps, IQ streams | Detect ingress noise, Wi-Fi/CB interference, LTE co-channel interference, thermal noise, impulse bursts | 5–25 TOPS, 2–4GB DRAM |
| AI-Augmented Transmit Power & Beam Adaptation | Reinforcement Learning (DQN, PPO)- Q-learning- CNN-based signal predictor | RSSI history, link-layer ACK/NACK, spatial signal distribution | Learn dynamic output power / beam direction adjustments to maintain link stability and reduce power | 10–25 TOPS, 4–8GB DRAM |
| Antenna/Beam Steering Optimization | Graph Neural Networks (GNNs) for mesh- Transformer (temporal RF states)- Bayesian optimization models- LSTM for historical feedback loops | CSI (Channel State Info), Angle of Arrival (AoA), beam index reports, device density, mobility pattern | Optimize antenna beam direction, switching, handover paths (esp. for MIMO, mesh Wi-Fi, or 5G FWA) | 10–50 TOPS, 4–8GB DRAM |

| Use Case | Sample Architecture | Framework / Format | Model Summary | Edge Deployment Notes |
|---|---|---|---|---|
| PHY Signal Quality Estimation | 1D CNN with residual blocksInput: Time-series (SNR, CINR, RSSI, BER) | PyTorch → TorchScriptONNX (for TensorRT or EdgeTPU) | 3 conv layers → 2 dense layers (ReLU) → Softmax (3 classes: Good/Degraded/Poor) | <2M parameters, ~1.5MB footprint. Runs in <5ms on 2.5–10 TOPS NPUs. |
| | LSTM or GRU (Temporal Quality) | PyTorch / TVM / TFLite | 2 LSTM layers (32–64 units) → Dense → Softmax | Low-frequency polling (every few seconds). Needs 0.5–1 sec sequence buffer. |
| Noise / Interference Detection | 2D CNN (e.g., MobileNetV2 or ResNet18-lite) on spectrogram slices | TensorFlow LiteONNXTVM | Input: 128x128 FFT images (spectrograms)Binary classifier or anomaly score | ~5–8MB models. Can run at 10 FPS locally with <10 TOPS hardware. |

| Use Case | Sample Architecture | Framework / Format | Model Summary | Edge Deployment Notes |
|---|---|---|---|---|
| | Autoencoder with 3 conv/deconv blocks | PyTorch or TFLite | Input: FFT-based imageOutput: reconstruction loss | Unsupervised anomaly detection – train with known "clean" RF snapshots. |
| AI-Driven Transmit Power / Beam Control | DQN (Deep Q-Network) | PyTorch + TorchScriptONNX | Input: RSSI history, ACK rate, mobility patternAction: power level / beam index | Low frame rate RL (e.g., 1 per 500ms). 2–3 dense layers sufficient. |
| | Shallow MLP for Q-learning | TensorFlow Lite or ONNX | 3 dense layers (64–128–64) with ReLUOutput: Q values for 4–6 actions | Easy to deploy with local reward feedback (e.g., throughput improvement). |
| Antenna / Beam Steering Optimization | Graph Neural Network (GAT) for multi-node mesh | PyG → TorchScript | Nodes: CPEs/APs; Edges: SNR/RSSIPredict: route or beam pattern | For multi-node topologies with dynamic signal graph. Heavier runtime. |
| | Transformer Lite for temporal CSI/AoA | Hugging Face TransformersONNX | Encode temporal sequence of CSI → Dense heads (regression or classification) | Quantized small transformer (12–24 heads, <6MB) can run on 20+ TOPS NPUs. |
| | Bayesian Optimization + DNN surrogate | scikit-optimize + PyTorch | Use lightweight neural net to emulate signal environment; search optimal beam/power configs | Ideal for one-time optimization (e.g., install time or per device move). |

**Time-Series Models** like **LSTM** and **1D-CNNs** are ideal for learning temporal correlation in signal metrics (SNR drift, burst noise, etc.).

- **Spectral-based approaches** (e.g., 2D-CNN on spectrograms or constellation maps) excel at identifying **impulse noise**, **interference**, or **modulation defects** in RF environments.
- **Autoencoders or Isolation Forests** can work in **unsupervised** settings for **anomaly detection** when labeled RF training data is sparse.
- **Reinforcement learning** is particularly useful for **adaptive power/beamforming**, where outcomes (packet success, link throughput) are used as rewards.
- **Graph-based models** are emerging in **mesh environments** where node relationships (topology + RF states) influence routing and beam dynamics.

**Table 25- Edge-Ready Considerations**

| Model Constraint | Recommendation |
|---|---|
| Compute | Use quantized models (INT8, FP16) on NPUs (2.5–50 TOPS range) |
| Memory | Limit history buffer sizes for LSTM/Transformer inputs |
| Privacy | Avoid transmitting raw IQ or FFT data—do all analysis locally |
| Latency Sensitivity | Use lightweight models with <5ms inference where real-time tuning is needed |

### *4.1.6. Layer 2/3 Packet inference*

One of the key areas that the use of AI is being considered in the GW/Router edge is to do packet inspection for the following use cases

- Security and Anomaly detection on packet flows
- Device ID and Service Fingeprinting – trying to identify what services are running on what devices to help with QoS mechanisms

If we look at **Layer 2 and Layer 3** and the data we have to use in AI inference we have the following general areas

- **Layer 2 (L2)**: Includes Ethernet headers — MAC addresses, VLAN tags, frame types.
- **Layer 3 (L3)**: Involves IP headers — source/destination IPs, DSCP, TTL, IP options, and fragmentation fields.

These layers do not include application payload, but offer rich metadata enabling inference via metadata analysis, statistical modeling, and machine learning.

#### *4.1.6.1. Security and Anomaly Detection*

Where we try and detect scanning, spoofing, volumetric attacks, route changes, misconfigurations – we can use features like

- L2: MAC spoofing, VLAN anomalies.
- L3: Unusual TTLs, IP fragmentation, source/destination distribution, flow symmetry.
- Many Layer 2 devices have VPN or tunneled content so considering that as well as non-Tunneled flows from a performance perspective :

**Table 26- L2/L3 AI Model Uses**

| Compute Level | Feasibility (Non-Tunnel) | Feasibility (Tunneling) |
|---|---|---|
| 2.5 TOPS / 2GB | Basic threshold/rule-based IDS | Limited — unable to analyze tunnel headers deeply |
| 10 TOPS / 4GB | Lightweight statistical ML models | Can start extracting tunnel header (e.g., GRE, IPSec outer headers) |
| 25 TOPS / 8GB | Flow correlation, NetFlow clustering | Good tunnel decapsulation; model can correlate across outer+inner headers |
| 50 TOPS / 8GB | Near-real-time detection, unsupervised learning (autoencoders) | Full tunneled traffic mapping; accurate behavior anomaly classification |

Multiple public models exist for Security and IDS development. Leading companies in these fields are expected to deploy trained models on edge devices. Edge Architectures built for silicon will enable fast packet access to NPUs, meeting advanced application needs.

**Table 27- Sample Public Security and IDS Models and Performance**

| Model | Params | Quantization | Use Case | Suitable For |
|---|---|---|---|---|
| AutoEncoder (custom PyTorch/Keras) | <10M | FP32 / INT8 | Unsupervised anomaly detection | 2.5–10 TOPS |
| GluonTS DeepAR / LSTM | ~20M | FP32/INT8 | Time-series anomaly, DDoS | 10–25 TOPS |
| Facebook Prophet | <1M | FP32 | Seasonality-based detection | 2.5 TOPS |
| HuggingFace TimeGPT (Nixtla) | 100M+ | FP32 / INT8 | Real-time traffic anomalies | 25–50 TOPS |
| OpenTelemetry + LightGBM | ~30M | INT8 | Feature-based anomaly detection | 10–25 TOPS |

#### *4.1.6.2. Device Identification*

We use local inference in GW, cloud systems, or a hybrid model to fingerprint endpoints by their L2/L3 behavior (such as IoT device type or OS). This relies on packet feature analysis.

- L2: MAC OUI, ARP behavior, VLAN tagging.
- L3: IP TTLs, DHCP behavior, traffic patterns over time.

And then if we look at the performance needed for tunnelled and non-tunneled flows – the following is the feasible.

**Table 28- Device ID Capabilities by CPE Performance**

| Compute Level | Feasibility (Non-Tunnel) | Feasibility (Tunneling) |
|---|---|---|
| 2.5 TOPS / 2GB | Simple rule-based (MAC OUI tables) | Not feasible — tunnel hides endpoint identity |
| 10 TOPS / 4GB | Time-series patterns (heuristics + rules) | Limited unless outer header shows endpoint info |
| 25 TOPS / 8GB | Classifiers (random forest, XGBoost) on L2+L3 | Outer+inner IP analysis if tunnel headers are visible |
| 50 TOPS / 8GB | Deep learning on flow fingerprints | Can fingerprint devices even within tunnels with decapsulation |

Some base models to leverage to train and deploy an edge based Device fingerprinting solution.

**Table 29- Public Model Samples for Device Fingerprinting**

| Model | Params | Quantization | Use Case | Suitable For |
|---|---|---|---|---|
| Random Forest / XGBoost | <100k–1M | INT8 | MAC/IP fingerprinting | 2.5–10 TOPS |
| TabNet (by Google) | ~10M | INT8 / INT4 | End-to-end tabular fingerprinting | 10–25 TOPS |
| ResNet1D (time-series variant) | ~5–10M | FP32/INT8 | MAC-based sequence modeling | 10–25 TOPS |
| Tiny Transformer (HuggingFace) | ~30M | INT8 | Behavioral identity classification | 25–50 TOPS |

Note: Identifying Device ID and Service ID requires a hybrid method, using numerous data points from many homes and training with large datasets. After training the model with GW and lab data, it is deployed on the Edge to infer device or service types based on specific packet flow markers.

#### *4.1.6.3. Service Fingerprinting (Identifying Type of Application)*

Determining priority and QoS for multiple home applications is essential, especially for identifying work-from-home video conferences and ensuring they receive higher priority. While L4S tagging is increasingly used for QoS, service tagging offers insights into usage patterns without needing app-specific prioritization. The trained Edge model aims to infer app type (video, VoIP, P2P, browsing) without DPI, using typical dataset features.

- Flow durations, packet inter-arrival times, DSCP values, TTL variance.

The performance capabilities of different NPU/DRAM setups for non-tunneled and tunneled packets are show below

**Table 30- Performance Requirements of AI Models on Non-Tunneled and Tunneled Packets**

| Compute Level | Feasibility (Non-Tunnel) | Feasibility (Tunneling) |
|---|---|---|
| 2.5 TOPS / 2GB | Binary classification (bulk vs real-time) | No visibility inside tunnel, poor inference |
| 10 TOPS / 4GB | Flow feature extraction + basic classifiers | Some app-level tunnel inference possible (e.g., QoS hints) |
| 25 TOPS / 8GB | Multi-class classification; correlation across flows | With decapsulation, infer tunneled app type |
| 50 TOPS / 8GB | Deep temporal modeling (e.g., LSTM on flow stats) | Can fingerprint multiplexed services in VPNs or SD-WAN tunnels |

The following suggest some base models that can be trained to deploy to the edge AI environment to be able to indentify service types

**Table 31- Sample Public AI Models That Can be Used in Service Fingerprint Applications**

| Model | Params | Quantization | Use Case | Suitable For |
|---|---|---|---|---|
| NetML (Flow-based CNN) | ~4M | INT8 | Multiclass service prediction | 10–25 TOPS |
| NetScope (from Cisco) | ~30M | FP32 | Application behavior fingerprinting | 25–50 TOPS |
| K-Means + Decision Tree | <1M | NA | Lightweight protocol class | 2.5–10 TOPS |
| HuggingFace BERT for time-series (TS-BERT) | 100M | INT8 | Complex flow type prediction | 25–50 TOPS |

#### *4.1.6.4. Quality of Service (QoS) Inference*

How can perceived service quality—such as latency, jitter, and congestion—be inferred? Is it possible to use features like DSCP markings, TTL drops, L3 retransmissions, and inferred out-of-order delivery from flow data?

**Table 32- Potential for QoS AI Inference for Tunneled and Non-Tunneled Packets**

| Compute Level | Feasibility (Non-Tunnel) | Feasibility (Tunneling) |
|---|---|---|
| 2.5 TOPS / 2GB | Heuristic rule-based QoS deduction | Tunnel QoS markings hard to interpret |
| 10 TOPS / 4GB | Statistical time-series models on packet flows | Outer headers help infer tunnel path quality |
| 25 TOPS / 8GB | Flow-based latency/jitter estimation; congestion windows | Able to extract inner DSCP if decapsulation is light |
| 50 TOPS / 8GB | Dynamic QoS classification; feedback models | Supports full tunnel inspection + latency maps |

Some base models that could be used to train on these quality and QoS indicators include.

**Table 33- Sample Public AI Models That Can Be Used in QoS Inference**

| Model | Params | Quantization | Use Case | Suitable For |
|---|---|---|---|---|
| ARIMA / Holt-Winters | ~1M | NA | Time-series QoS estimation | 2.5–10 TOPS |
| GRU-D (for missing QoS data) | ~5–8M | FP32 / INT8 | Estimating QoS in incomplete flows | 10–25 TOPS |
| Conv-TCN (Temporal Convolution Net) | ~10M | INT8 | Real-time latency estimation | 25 TOPS |
| MLFlow + Tabular Classifier (e.g., CatBoost) | ~20M | INT8 | Composite QoS score classification | 10–25 TOPS |

Before we leave the L2/L3 packet inference potential there are a number of other areas that can be leveraged from trained models. These include:

**Table 34- Other Potential Areas for AI Packet Inference**

| Inference | Description | Feasibility Notes |
|---|---|---|
| Traffic Directionality | L2 MACs and IP source/destination trends reveal client/server roles | Low compute requirement |
| Broadcast/Multicast Analysis | Useful for detecting misbehaving devices | OK at all levels |
| Flow Symmetry / Behavioral Profiling | Used in anomaly detection and device classification | Needs 10+ TOPS to be effective |
| VPN or Tunnel Type Detection | Identify GRE/IPSec/MPLS overlays | Requires 25+ TOPS for pattern recognition |
| MAC-IP Binding Violations | Spoofing/mobility tracking | Viable with 10+ TOPS |

| Inference | Description | Feasibility Notes |
|---|---|---|
| Service Reachability Mapping | Graph of accessible services over time | Heavy memory + compute needed; 25+ TOPS optimal |

### 4.1.7. Sensor AI Inference

This area includes the desire to be able to use the sensing potential ability of RF and other elements of a device. It typically manifests itself as use cases like

- Presence and Location
- Wi-Fi sensing for movement
- Wi-Fi location
- BLE sensing for location
- FMCW – Frequency Modulation Continuous Wavelength
    - Can be used to detect breathing and heart rate
- UWB – Ultra Wide Band
    - Can be used to detect location in room

The accuracy of location, motion, and presence applications varies by technology. Wi-Fi is the most common in homes with gateways, routers, or STBs, while BLE is also prevalent. FMCW can be integrated inexpensively and excels at device-free monitoring of individuals, such as elderly patients in beds or chairs. UWB is increasingly being incorporated into Wi-Fi devices.

**Table 35- Sensing Solutions Inference Accuracy and Complexity**

| Technology | Inference Types | Accuracy | Hardware | AI Complexity | Privacy |
|---|---|---|---|---|---|
| Wi-Fi Sensing | Motion, presence | ~1–3 m | Standard AP/clients | Low–Medium | High |
| Wi-Fi Location | Indoor position | ~1 m | Multi-AP / CSI | Medium | High |
| BLE Sensing | Proximity | ~1–3 m | Beacons/phones | Low | High |
| FMCW Radar | Breathing, gestures | ~0.1 m | Radar SoCs | Medium–High | Very High |
| UWB | In-room location | ~10 cm | UWB chipsets | Low–Medium | Very High |

There is also a role for AI with ‘Sensor Fusion’ (e.g., combining UWB + Wi-Fi + FMCW) enables more reliable and robust context awareness.

The AI inference models for this Sensor Fusion typicall mean having :

- CNNs for range-Doppler maps (FMCW)
- RNNs/LSTMs for temporal activity inference
- GNNs for device-location-behavior modeling

Deployment targets commonly include edge devices such as routers, smart hubs, and mobile devices. Workload requirements typically range from 2.5 to 50 TOPS, depending on the complexity of the application.

A comprehensive summary of use cases by sensor type and performance requirements for AI TOPS/DRAM is presented in the following table, serving as an effective checklist.

**Table 36- Example Sensing Application Based on CPE AI Capability/Performance**

| Application Type | Sensor Type | 2.5 TOPS / 2 GB DRAM | 10 TOPS / 4 GB DRAM | 25 TOPS / 8 GB DRAM | 50 TOPS / 8 GB DRAM |
|---|---|---|---|---|---|
| Presence Detection | Wi-Fi Sensing | Basic RSSI thresholding | CSI pattern detection | Flow-based movement classification | Multi-room motion tracking |
| | BLE | RSSI-based proximity | Adaptive thresholding | Movement trend learning | Room-level presence graphs |
| | FMCW | Not viable | Basic motion detection | Multi-target presence | Range-angle Doppler mapping |
| | UWB | Not viable | Entry/exit detection | Tag-based zone presence | Multi-object tracking |
| In-Room Localization | Wi-Fi | Not viable | ToF triangulation | CSI-based fingerprinting | Real-time location mapping |
| | BLE | Beacon zoning | Room-level localization | Improved accuracy with ML | Indoor routing |
| | FMCW | Not viable | Not viable | Object localization (single) | Micro-positioning (<20cm) |
| | UWB | Not viable | Tag tracking | Sub-meter precision | 3D in-room positioning |
| Vital Signs Monitoring | FMCW | Not viable | Coarse breathing detection | Breathing + heart rate | Real-time waveforms + multiple people |
| | Wi-Fi | Not viable | Not viable | Breathing pattern from CSI (experimental) | Low-resolution HR/breathing |
| Gesture Recognition | FMCW | Not viable | Not viable | Simple gestures (swipe, tap) | Complex gestures (point, wave, zoom) |
| | UWB | Not viable | Not viable | Entry-level gesture detection (AoA) | High-resolution hand tracking |
| Fall Detection / Human State | FMCW | Not viable | Not viable | Fall detection + posture | Posture + gait analysis |
| | UWB | Not viable | Not viable | Simple activity zones | Activity heatmaps |
| Device-Free People Counting | FMCW | Not viable | Binary occupancy | People count (1–3) | Dense occupancy maps |
| | Wi-Fi | Room occupancy estimate | People count (1–2) | Trend classification | Flow-based occupancy modeling |
| Secure Access / Proximity Unlock | UWB | Not viable | Static unlock (car door, lock) | Smart access with entry direction | Time-of-flight + angle authentication |
| | BLE | RSSI unlock | Adaptive geofencing | Activity-aware unlock | Spoof-resistant logic |
| AR/VR Spatial Anchoring | UWB | Not viable | Not viable | Tag-based zone mapping | Real-time AR object anchoring |
| | FMCW | Not viable | Not viable | Wall/object outline mapping | Real-world SLAM fusion |
| Elder Care / Health Sensing | FMCW | Not viable | Basic breathing alerts | Heart rate trends | Multimodal state tracking |

| Application Type | Sensor Type | 2.5 TOPS / 2 GB DRAM | 10 TOPS / 4 GB DRAM | 25 TOPS / 8 GB DRAM | 50 TOPS / 8 GB DRAM |
|---|---|---|---|---|---|
| | BLE/Wi-Fi | Presence only | Activity trend flags | Passive monitoring | Behavior modeling |
| Sensor Fusion (Hybrid) | Wi-Fi + BLE | Rule-based | RSSI + flow analysis | Light ML fusion | Temporal behavior modeling |
| | FMCW + UWB | Not viable | Not viable | Range-Doppler + ToF sync | Cross-modal tracking (motion + location) |

If we look at what base models that can be considered to use for these applications

**Table 37- AI Model and Type and Specification by Sensing Application**

| Use Case | Sensor | Starting AI Model | Model Type | Params (Approx) | Deployment Notes |
|---|---|---|---|---|---|
| Presence Detection | Wi-Fi / BLE | Logistic Regression / XGBoost | Tabular ML | <1M | Lightweight, suitable for 2.5 TOPS, quantize to INT8 |
| | FMCW | 1D CNN | CNN | ~1–3M | Preprocess range-Doppler input, quantize to INT8 |
| | UWB | Random Forest | Tree-based | ~500K | Simple RSSI/ToF classification, fast inference |
| In-Room Localization | Wi-Fi (CSI) | Tiny ResNet or TCN | CNN / Temporal CNN | ~2–5M | Trained on CSI features, use Edge TPU/NPU |
| | BLE | KNN or SVM | Non-DL | <500K | Fast localization; quantized or embedded form |
| | UWB | DNN + AoA/ToF Features | MLP | 3–10M | Requires fused antenna input; INT8 recommended |
| Vital Sign Monitoring | FMCW | 1D CNN + RNN | CNN + LSTM | ~5–10M | Respiratory signal tracking, quantize to INT8/bfloat16 |
| | Wi-Fi (experimental) | BiLSTM or Transformer | Sequence model | ~20M+ | Needs stable CSI, best at 25+ TOPS |
| Gesture Recognition | FMCW | ResNet-18 (1D/2D) | CNN | 10–15M | For Doppler-range map input; prune + quantize |
| | UWB | TCN or TinyYOLO-style CNN | Temporal CNN | ~5M | Use for coarse AoA/ToF-based gestures |
| Fall Detection | FMCW | GRU or BiLSTM | RNN | ~3–6M | Works on Doppler envelope or filtered signal |

| Use Case | Sensor | Starting AI Model | Model Type | Params (Approx) | Deployment Notes |
|---|---|---|---|---|---|
| | UWB | DNN classifier | MLP | ~2–5M | Fast response with short signal windows |
| People Counting | FMCW | YOLO-Lite or MobileNet | CNN | ~3–10M | For radar heatmaps or clustered reflections |
| | Wi-Fi | LSTM + Flow Embedding | RNN | ~8–12M | Flow-level activity requires more context |
| Proximity Unlock / Access | UWB | SVM / Decision Tree | Classical ML | <1M | Very low latency, use post-ToF processing |
| | BLE | Simple MLP | DNN | ~1M | Embed into MCU-class devices |
| AR/VR Anchoring | UWB | Graph Neural Network (GNN-lite) | GNN | ~15M+ | Works with spatial graphs, INT8 or bfloat16 |
| | FMCW | Sparse 3D CNN | CNN | ~20–30M | If full-range map used; needs 25+ TOPS |
| Elder Care / Health | FMCW | Temporal Autoencoder | CNN + RNN | ~6–10M | Breathing/HR trend detection, INT8 possible |
| | BLE/Wi-Fi | LSTM + anomaly classifier | RNN + Classifier | ~4–8M | Long sequence analysis (fall, no-motion) |
| Sensor Fusion (Hybrid) | Wi-Fi + BLE | Ensemble of classifiers | Tree-based / RF | <5M | Combine multiple modalities at low compute |
| | FMCW + UWB | Late-fusion LSTM | RNN + MLP | ~15–25M | Complex, suited for 25+ TOPS devices |

The choice of the solution above is filtered by the TOPS/DRAM available so the following considerations are in place

**Table 38- Sensing Models by CPE Performance**

| Platform | Suitable Models |
|---|---|
| **2.5 TOPS / 2 GB** | Tree models, SVM, simple MLP, pruned 1D CNN |
| **10 TOPS / 4 GB** | Tiny CNNs, RNNs, quantized gesture models |
| **25 TOPS / 8 GB** | 2D CNNs (YOLO-lite), LSTMs, attention-based networks |
| **50 TOPS / 8 GB** | Transformer-style models, GNNs, multimodal fusion models |

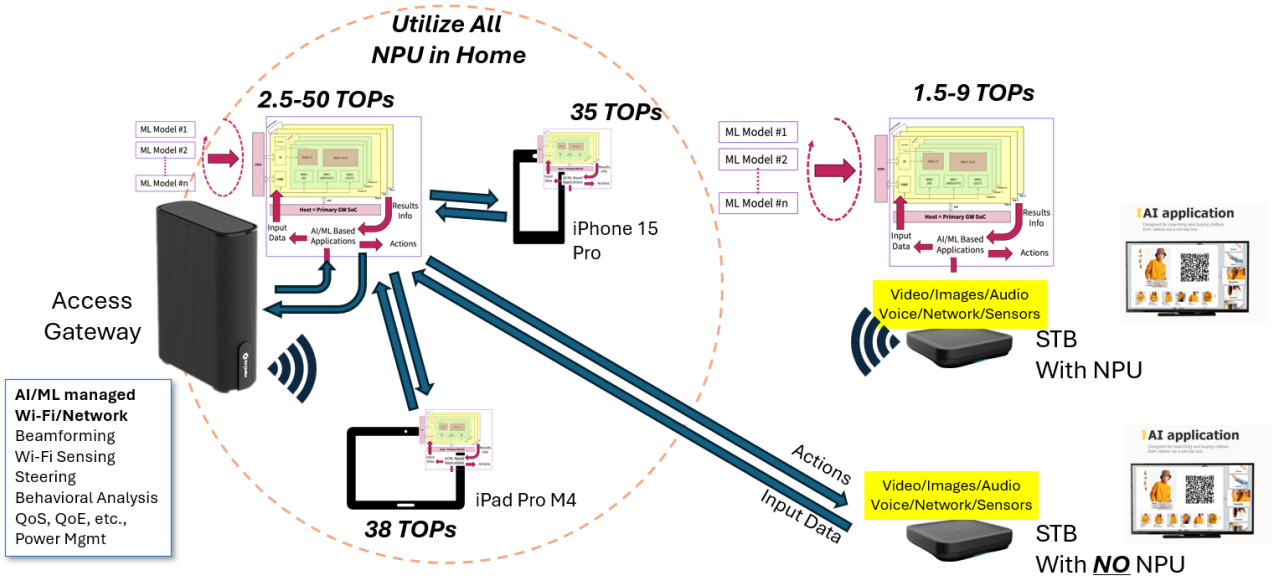
## 4.2. Multiple AI Inference Devices – Aggregating Their Capabilities and Offering Proxy Capabilities to Maximize Capital Investment

Distributed home AI enables multiple household devices to work together without each needing high-end hardware. For instance, a set-top box with an NPU can process data sent from a less capable gateway or transcribe audio for display on a TV when paired with a home AI sidecar. This approach permits phased AI upgrades, letting service providers advertise device capabilities and balance workloads across the network. Incremental integration reduces costs and turns new devices into shared resources, effectively creating an "edge cloud" within the home.

In the distributed CPE AI Proxy model, new devices can join to run AI models and inference tasks. Service Provider smartphones and similar devices, if available, offer substantial processing power for Home Edge AI, though their presence is not guaranteed. The Gateway manages AI deployment within the Home AI group by coordinating secure connections and data exchanges with devices. While this architecture is not described in detail here, it's promising for pooling resources from multiple set-top boxes for various real-time and non-real-time applications. Example uses include:

- STB AI inference platforms – provide proxy packet security and service identification for broadband gateways
- High inference gateway or sidecar solutions – enable translation and transcription services for set-top boxes lacking AI capabilities
- Broadband gateways with integrated AI functions – support proxy-based object inference for IP camera streams
- AI-enabled set-top boxes – facilitate object inference for IP camera feeds through proxying

**Table 39- Distributed CPE/Client Proxy Model**



# 5. Use Case Matrix and Value Creation for Home Edge AI

After reviewing the technology tiers and model mappings, our focus shifts to the practical applications enabled by Home Edge AI and the corresponding value they offer Service Providers. We will introduce a Use Case Matrix that connects technical capabilities with business and operational outcomes, ranging from cost reductions through automation to new revenue streams via advanced AI features. Each use case will be assessed in terms of edge AI's facilitation, as well as the minimum and optimal hardware performance tiers required to ensure satisfactory user experience.

Major use case domains include: Customer Support & Operations, Value-Added Services (VAS) for Revenue, and Core Service Enhancements like network optimization and security. Many use cases span multiple categories (e.g. a feature could both reduce costs and increase customer satisfaction).

## 5.1. Customer Support Automation

Service providers dedicate considerable resources to customer support operations. As a result, customer support has emerged as a primary focus area for artificial intelligence automation and efficiency improvements. The industry trend is to transition a substantial portion of customer support tasks to AI-driven solutions to minimize human resource requirements. Currently, many support functions have implemented first-level chatbots—early expert systems designed to address simple issues before escalating them to human agents. These chatbots are rapidly being replaced by advanced generative AI models, and significant efforts are underway to leverage state-of-the-art large language models for enhanced customer support experiences.

When human customer service representatives become involved, they typically face the challenge of quickly familiarizing themselves with each customer’s specific issue, configuration, location, and home devices. This need for rapid onboarding has led to the development of Customer Support Dashboards, which aim to provide agents with the necessary data to operate efficiently within their respective domains. Generative AI, combined with advancements in voice technology and empathetic communication, presents an opportunity to deliver consistent, high-quality support around the clock without the limitations of human fatigue or frustration.

Additionally, there is growing interest in integrating support functionalities directly into home devices. Deploying lightweight support language models on edge devices could further reduce operational costs, especially given that cloud-based generative customer support solutions tend to be more expensive. Such architectures also enable features like “Talk to Home” or “Talk to Gateway,” allowing for basic troubleshooting and configuration assistance even during internet outages. However, as noted previously, small language models currently have limited capabilities. The primary objective is thus to ensure these models are sufficiently effective so that customers are willing to use them rather than bypassing automated options in favor of human assistance.

Additionally, Home-edge AI offers valuable capabilities in two key areas: proactive issue detection and mitigation, as well as the creation of innovative interactive support paradigms facilitated by AI avatars. For instance, an AI system integrated into the gateway can continuously monitor network health and user device metrics to anticipate potential issues, such as identifying deteriorating signals that may indicate an impending cable modem failure. By detecting and addressing problems—such as initiating automatic reboots or parameter adjustments—prior to user awareness, these solutions help reduce costly truck rollouts and support calls. This approach exemplifies the use of AI for preventive maintenance and cost reduction.

Furthermore, service providers can deploy virtual support agents operating partly on home devices. For example, a support avatar displayed on the television can guide users through troubleshooting procedures even during internet outages by leveraging a local knowledge base, later syncing with the cloud for more complex queries. This strategy enhances customer satisfaction and lowers support demands, allowing a single agent to efficiently resolve numerous basic inquiries. Tier 2 devices are required to enable fundamental avatar interactions, such as implementing compact local language models for understanding common questions and providing on-device text-to-speech capabilities.

There is significant momentum behind the adoption of AI-generated Avatars to facilitate more comfortable human interactions. The widespread use of video conferencing has conditioned individuals to engage effectively with 'people on screens,' fostering greater intimacy and focus during support or sales engagements, as compared to interactions with devices that communicate solely through light signals. Although this topic is not the primary focus of this paper, it is advisable for Service Providers to consider how such advancements may influence customer relationships and engagement. It is conceivable that, in future solutions, each customer could interact with three standard Avatars representing the services offered by the Service Provider, powered by a sophisticated hierarchical AI system. In this framework, generative AI would be utilized locally to optimize cost, privacy, and latency, with processes transferred to the cloud when tasks exceed the capabilities of in-home technology. These three avatars would typically be

- **Home "Jarvis" Avatar**: This avatar offers comprehensive access to your Service Provider's Generative AI solution for both you and your household. The "Jarvis" system supports multimodal interaction on any set-top box or television within the home, enabling users to ask questions or issue commands via both audio and visual interfaces. Applications range from assisting with children's homework and configuring IoT devices, to interacting with current and future cloud-based agentic AI applications.
- **Customer Support Avatar**: This avatar retains knowledge of your household's history and environment, allowing for more personalized support. When issues cannot be resolved automatically, this avatar assists in troubleshooting connectivity, video, smartphone, billing, and other service-related concerns.
- **Sales Avatar**: This avatar provides Service Providers with an innovative method to upsell and promote new services, thereby increasing ARPU. Unlike traditional sales tactics that rely heavily on online portals, cold calls, or email campaigns, this system enables seamless transitions from customer support sessions to sales opportunities, fostering greater customer engagement and facilitating targeted promotions.
- **Premium Value-Added Services:** These create new revenue through offerings like home security monitoring, elder care, and AR/VR entertainment, powered by local Edge AI for privacy and low latency. Smart Security uses AI to monitor cameras and sensors, providing person detection and anomaly alerts as a subscription. Elder care services track daily patterns via sensors, alerting caregivers if abnormalities occur; collaboration with health or insurance partners enhances monetization. Service tiers vary based on feature sophistication, with more advanced analytics requiring higher tiers. Local AI also enables personalized advertising or upselling by analysing household viewing habits without sending data externally, triggering targeted offers and driving revenue using efficient models.
- **Core Service Enhancements:** Network optimization (AI-managed QoS and bandwidth), energy management (AI reduces power use, integrates with smart grids—e.g., home AI adjusts router or set-top usage for savings), and content curation (AI on set-top box analyzes viewing habits for tailored recommendations or highlights). Energy management and basic content curation typically require simple AI (Tier 1), while advanced real-time content analysis uses more complex AI (Tier 2 CPE device).

Another category, **Cost Reduction by Offloading Cloud AI**, offers a strong business case for service providers. By shifting certain AI processes from cloud APIs to edge devices, ongoing cloud inference costs can be reduced. For instance, after investing in an NPU, local processing of voice commands on a Tier 2 gateway can replace frequent paid cloud speech API calls, saving on operational expenses over time. Similarly, handling analytics like Wi-Fi motion detection on-device removes third-party fees. These savings contribute to the ROI of edge AI use cases, as noted in Table 3.

The table below connects AI-enabled features to their primary value for Service Providers: cost savings, new revenue or upselling, and improved customer satisfaction. It also shows the necessary device Tier for each use case, with the main benefit highlighted. While many use cases offer multiple advantages, each is marked by its key value driver.

**Table 40- Home Edge AI Use Case Matrix**

| Use Case Area | AI-Driven Features (Examples) | Value Proposition | Min Device Tier |
|---|---|---|---|
| Voice Assistant & UI | On-device voice assistant (basic commands); Local STT for voice remote; Real-time translation on STB | +Satisfaction (privacy, speed); saves cloud API costs | Tier 1 (basic), Tier 2 for full features |
| Customer Support | AI support avatar on TV; Automated troubleshooting & network self-healing | –Cost (fewer support calls); +Satisfaction (faster resolution) | Tier 2 (avatar), Tier 1 for basic diagnostics |
| Security Monitoring | Local camera analytics (intruder, pet, package detection); Face recognition for family/alerts | +Revenue (security subscription); +Satisfaction (privacy); –Cost (no cloud video streaming) | Tier 2 (single cam), Tier 3 for multi-cam HD setup |
| Elderly Care | Routine learning via sensors; Fall detection via Wi-Fi or camera; Wellness alerts | +Revenue (premium service); +Retention (sticky service) | Tier 1 (simple sensors), Tier 2 (if video or complex patterns) |
| Network Optimization | AI QoS classifier in gateway; Anomaly detection in router (e.g. cable signal diagnostics) | –Cost (less manual config, fewer outages); +Satisfaction (better performance) | Tier 1 (possible on CPU for small models) |
| Energy Management | AI optimizes CPE power use; Home energy alerts (e.g. AC left on) via IoT data analysis | –Cost (if SP powers devices); +Marketing (green initiative) | Tier 1 (low compute needed) |
| Personalized Content | Local content recommendation; AI-curated channels; AR/VR enhancements (sports stats overlay) | +Revenue (engagement, upsells); +Satisfaction (personalization) | Tier 1 (basic recommendations), Tier 2–3 (AR/VR features) |

| Use Case Area | AI-Driven Features (Examples) | Value Proposition | Min Device Tier |
|---|---|---|---|
| Upselling & Advertising | AI identifies upsell opportunities (usage patterns -> offers); Interactive sales avatar | +Revenue (higher take rate on offers) | Tier 1 (analytics), Tier 2 (avatar interface) |
| AI as a Service | Opening home AI platform to third parties (e.g. allow approved partner AI apps on gateway) | +Revenue (partner fees or new services); +Innovation | Tier 2 (to attract useful apps) |

Service providers are beginning to deliver AI platform capabilities by running containerized AI workloads on gateways. This lets third-party apps, like an insurance company's wellness monitoring tool, operate on ISP-supplied gateways, with possible revenue sharing for ISPs. Such a model can turn customer premises equipment (CPE) into a distributed edge cloud, strengthening the provider's role in the ecosystem.

About 70–80% of routine residential AI tasks—such as voice commands, single-camera analysis, and simple automation—can be handled locally with optimized models on CPE devices, offering lower latency and improved privacy. More complex or rare tasks (20–30%) like advanced language processing or multi-camera analytics can be sent to the cloud. This hybrid approach manages most tasks efficiently at the edge, reserving cloud resources for complex needs.

**Table 41- Home AI Use Cases – Value Proposition**

| Use Case Area | AI-Driven Feature | Value Proposition | Min Device Tier | Devices/Stack Involved |
|---|---|---|---|---|
| Voice & Ambient UI | Multilingual on-device voice assistant; Emotion-aware UI; Personalized voice UX | +NPS (localized, fast, private); –Cloud STT cost | Tier 1+ | STB, Smartphone, Cloud |
| | Low-latency voice remote translation (cloud fallback) | +NPS; +ARPU for multilingual households | Tier 2 | STB, Cloud |
| Customer Experience / Support | Predictive churn modeling with real-time in-home signal + usage data | +Retention; –Churn | Tier 1 (data), Tier 3 (prediction) | GW, Cloud |
| | Smart agent on STB/mobile for "explain my bill", QoE queries | –Cost; +Satisfaction | Tier 2 | STB, Smartphone |
| Security & Monitoring | Smart neighborhood watch: pattern anomalies across adjacent CPEs | +ARPU (security plans); +Community trust | Tier 2 | GW, Cloud |
| | AI-based pet/caregiver/presence detection from indoor cameras (privacy preserving) | +NPS; +Premium service upsell | Tier 2+ | STB, GW, Cloud |

| Use Case Area | AI-Driven Feature | Value Proposition | Min Device Tier | Devices/Stack Involved |
|---|---|---|---|---|
| Elderly & Wellness | Passive health trend modeling using Wi-Fi/FMCW (breathing, motion) | +ARPU; +Sticky retention with family services | Tier 1+ | GW, Smartphone, Cloud |
| | AI alerts for change in routine, with escalation logic (no video needed) | +Satisfaction; +Care ecosystem opportunity | Tier 2 | GW, Cloud |
| QoS & Network Efficiency | Adaptive bandwidth allocation based on app/service identification (Edge AI) | –OpEx; +QoE | Tier 1 | GW |
| | Congestion prediction with cross-CPE traffic pattern models | –Outages; +Uptime = +NPS | Tier 2 | Cloud, GW |
| | In-home mesh optimization using AI mesh topology learning | +NPS; –Truck rolls | Tier 2 | GW, Cloud |
| Energy & Sustainability | AI-controlled CPE sleep/wake cycles based on occupancy prediction | –OpEx; +Sustainability PR | Tier 1 | GW, Cloud |
| | Smart home load shaping (with partner devices) via AI orchestration | +Green revenue; +NPS | Tier 2 | GW, Cloud, IoT Hubs |
| Content Discovery & Personalization | Household-profile-based recommendation (not per user) | +Engagement; +Stickiness | Tier 1 | STB, Cloud |
| | Real-time mood-based UX adaptation (color, layout) via camera/voice | +Delight; +Upsell to premium UX | Tier 2 | STB, Smartphone |
| | AR companion app with AI-generated sports overlays / co-viewing | +ARPU; +Retention (next-gen entertainment) | Tier 3 | Smartphone, Cloud, CDN |
| Advertising & Upselling | Dynamic in-home ad slot bidding based on contextual AI (e.g., “kids home”) | +Revenue (targeted advertising) | Tier 1 (for triggers) | GW, Cloud, CDN |
| | AI avatar for on-screen upgrade suggestions during idle moments | +Upsell; +Stickiness | Tier 2 | STB, Cloud |
| Device & Service Management | AI detects and self-resolves CPE degradation (memory, temp, channel noise) | –Support cost; +Device life | Tier 1 | GW |
| | Predictive hardware/service failure warning (e.g., STB HDD, ONT signal) | –Replacement cost; +Trust | Tier 1 | GW, STB |

| Use Case Area | AI-Driven Feature | Value Proposition | Min Device Tier | Devices/Stack Involved |
|---|---|---|---|---|
| Smartphone + CPE Collaboration | Localized AI processing offload to phone from gateway (e.g., private AI tasks) | +Privacy; +Speed; Enables new mobile apps | Tier 1+ | GW, Smartphone |
| | Shared sensor fusion (e.g., Wi-Fi + UWB + BLE) across phone and GW | +Accuracy for wellness/security | Tier 2+ | GW, Smartphone, Cloud |
| AI-as-a-Service (B2B2C) | Open CPE AI APIs for 3rd-party developers (e.g., health, home automation) | +Revenue (API fees, partnerships) | Tier 2+ | GW, Cloud |
| | Allow user-pickable "AI widgets" (recommendation engine, elder care, etc.) from marketplace | +ARPU; +NPS; drives ecosystem play | Tier 2+ | GW, STB, Cloud |

From an overall perspective we are trying to do the following 3 objectives

**Table 42- Top Use Cases to Make, Save and Add Customers**

| Objective | Top Use Cases |
|---|---|
| Make Money (New ARPU) | Elderly care, security monitoring, targeted upselling, AI API marketplace |
| Save Money (Efficiency) | Self-healing networks, power management, AI-driven support, diagnostics |
| Improve NPS (Experience) | Voice UI, personalized content, mesh optimization, low-latency troubleshooting |

# 6. Recommendations and Roadmap for Deployment

Rolling out AI inference at the home edge should be done gradually. Service providers should use a phased strategy, upgrade devices and introducing AI features as business cases allow. There are recommended steps for both short-term (2024–2025) action and mid-term (2025–2028) planning. A hybrid edge-cloud architecture is also outlined to maximize benefits.

## 6.1. Short-Term (Next 12–18 months): Laying the Groundwork

Start by adding select AI features to current high-end CPE devices—like motion sensing via Wi-Fi or simple voice commands for set-top boxes. These upgrades often need only small firmware changes and can run on CPU-only devices. Launching with one or two free AI features, such as a basic voice assistant, provides value and lets you gather real-world data. Use customer feedback and telemetry to improve AI models and settings over time.

### *6.1.1. Edge AI NPU/AI Requirements*

Optimize device specifications during procurement by ensuring alignment with AI requirements. Specifically, confirm that all new CPE models ordered for deployments from 2025 onwards are equipped with at least an entry-level NPU and adequate memory.

- Entry-level NPUs for 2025+ deployments are being considered.
- For security and packet inference with CNN/RNN models, 2.5 TOPS supports 10 Gbps speeds.
- Video processing at low frame rates and sub-4K resolution typically requires 2.5–7 TOPS, mainly for STB applications.
- Higher TOPS are needed for SLM, transformer, and generative AI workloads, driving next-gen device development for 2026+, including AI sidecar research.
- If local SLM and generative AI in homes increase, NPUs over 50 TOPS may become standard within three to five years.

When assessing INT8 quantization and Transformer model benchmarks, consider Tokens per Second. A key issue is the use of DRAM in AI devices—LP-DDR5 offers better performance than DDR4, which is still common in gateways and set-top boxes.

A comparative table shows estimated tokens-per-second for AI inference at different SoC compute levels (2.5–50 TOPS) using DDR4 and LP-DDR5 memory, highlighting how memory bandwidth and latency affect AI performance—especially in Transformer models for voice, NLP, or edge LLM tasks.

**Table 43- Tokens/Sec Inference Throughput: DDR4 vs LP-DDR5**

| SoC Compute | DDR4 (Low-BW, ~25–35 GB/s) | LP-DDR5 (High-BW, ~50–65 GB/s) | % Gain with LP-DDR5 | Application Examples |
|---|---|---|---|---|
| 2.5 TOPS / 2GB | ~500–1,000 tokens/sec | ~700–1,200 tokens/sec | +20–40% | Wake word, basic NLP command parsing |
| 4.5 TOPS / 2–4GB | ~1,500–2,500 tokens/sec | ~2,200–3,200 tokens/sec | +30% | Voice remote STT, FAQ bots |
| 6.8 TOPS / 4GB | ~3,000–4,500 tokens/sec | ~4,200–6,000 tokens/sec | +35–40% | Multi-command processing, local chatbot |
| 25 TOPS / 8GB | ~10,000–15,000 tokens/sec | ~14,000–20,000 tokens/sec | +30–40% | Real-time STT/NLU; streaming translation |
| 50 TOPS / 8GB | ~20,000–30,000 tokens/sec | ~28,000–40,000 tokens/sec | +35–45% | Local LLM agents, multi-user assistants |

**Table 44- Memory Impact Explanation**

| Memory Spec | DDR4 | LP-DDR5 |
|---|---|---|
| Bandwidth | ~25–35 GB/s | ~50–65 GB/s |
| Latency | Higher (~70–90 ns) | Lower (~40–60 ns) |
| Power | Higher | Lower |
| Cost | Lower | Slightly higher |

The Significance of Enhanced Tokens-Per-Second Speed for AI Models

- AI inference for Transformers (especially multi-head attention) is **memory bandwidth-bound**, not just compute-bound.
- LP-DDR5 allows faster token embedding/decoding, larger context windows, and smoother parallelism for streaming workloads.

**Table 45- Recommended Memory Types for Edge/Clients**

| Device Class | Recommended Memory | Reason |
|---|---|---|
| Entry BB GWs (2.5–4.5 TOPS) | DDR4 | Lower cost, sufficient for basic AI |
| Mid-high STB / Smart GW (6.8 TOPS) | LP-DDR5 | Improves local assistant response, STT |
| Premium STB / AI Hub (25–50 TOPS) | LP-DDR5 | Required for real-time multimodal/LLM |
| Mobile Edge Devices (Smartphone/GW Fusion) | LP-DDR5 | Lower power + high throughput |

To illustrate the impact of tokens per second and inference speed limitations on applications, consider several fundamental use cases. Real-time translation, for instance, has garnered significant attention and warrants comprehensive examination. Many new multimodal interfaces introduced by companies such as Meta and Google—particularly XR Glasses—feature real-time translation as a core service. These solutions are generally implemented using cloud-based models, which often require buffering to ensure sufficient contextual information for accurate translation. The processing pipeline typically involves converting speech to text, translating text, and then rendering audio output; efforts are underway to streamline these steps to enhance overall performance.

Deploying efficient real-time translation models at the edge remains challenging, both in terms of model availability and technical constraints. Whisper is currently the most prominent public model; however, it tends to favor translation from various languages into English rather than direct language-to-language translation. As a result, it necessitates a dual-stage pipeline, increasing both processing time and memory requirements.

**Table 46- Typical AI Tasks and Tokens/Sec Required**

| Task | Min Tokens/sec |
|---|---|
| Wake word detection | ~100 |
| Voice command parsing (STT + intent) | ~1,000–2,000 |
| Conversational agent (chat) | ~5,000–10,000 |
| Real-time translation | ~10,000+ |
| Tiny LLM summary (~100 tokens/sentence) | ~15,000–20,000 |

### *6.1.2. Home Edge CPE Memory Recommendation*

Memory type is crucial for AI-based CPE devices, with the shift to LP-DDR5 driven by the need for better performance and lower power consumption. DRAM manufacturers are moving to DDR5 to meet growing demand and improve AI inference. Memory quantity also matters for AI and machine learning,

but cost and ROI have traditionally limited capacity. Now, rising requirements—across gateways, STBs, smartphones, and laptops—are pushing up DRAM needs, especially as these devices increasingly run advanced language models locally. For gateways and STBs:

- Gateway – With containerized software and new features, expanding Gateway memory to 4GB DRAM is likely. Integrating AI models may require 8GB, as SLM and Transformer models benefit greatly from increased memory.A rule of thumb for Gateways is that there is typically ~500MB available for applications and AI in a 2GB footprint with
- STB – because of the AI values for Video based and Audio based inference – the NPU is standard in almost all STB SOC shipping today. The STB silicon is also advancing the TOPS performance to cover more and more AI applications. As STB have been trending to commoditized MPEG decoders/Streaming App players – they have been under strict cost pressure so memory has been conservative. However, 4GB has now set as the baseline but with many including the author of this paper promoting that a move to 8GB of DRAM opens up a new level of performance nad future capability for AI based features. Additionally LP-DDR5 seems to be the path to move to in the next 12-24 months. We recommend looking at future STB not as MPEG decoders but as a key in room multi modal AI portal ***first*** and as a video player secondary. In this context moving to 8GB LP-DDR5 to 'own the living room' for AI Multimodal functions and actions seems a good investment.
- Technologists now emphasise adequate memory before procurement, recommending a minimum of 2 GB RAM for budget devices by 2025, and 4–8 GB for mid- to high-tier devices by 2027. Many existing devices have 1 GB or less, limiting performance. Selecting SoCs with integrated NPUs or GPUs is encouraged despite slightly higher costs; the difference in processing power (e.g., 5 vs. 25 TOPS) can greatly affect AI functionality. Interim solutions like USB AI dongles or mesh Wi-Fi nodes with AI acceleration may be considered, but built-in capability remains ideal for user experience.

Presenting figures on capital investment in AI inference and DRAM, compared to cloud costs, helps clarify potential advantages for service providers. Free AI chatbots mainly drive user acquisition and revenue rather than direct profit. The rise of app-based chatbots marks a major change in consumer internet and app engagement, potentially shifting the ecosystem toward agent-like AI functions. This paper offers projections and hypotheses from the author to contextualize why edge investments may be beneficial, which will be evaluated for accuracy.

- Free AI chatbot apps operate at a loss and use user prompts to train models.
- The $20 tier also loses money, limits usage (especially video inference), and collects prompt data unless disabled.
- The $200–$300 tiers continue to lose money, particularly with automated or video-intensive use, and still impose usage limits.

Currently, AI inference and chatbot services typically cost companies about $100 per month, not $20. This price reflects the value of access to legal, financial, and knowledge resources, aligning with annual spending on professional advice. When advanced educational and career support are included, $100 monthly is a modest investment compared to the potential salary gains from using AI.

Returning to the central focus of this paper- the investment made by service providers in Capital for Home Edge infrastructure- this enables local capabilities that can reduce cloud expenses for certain or all segments of the AI service chain.

When evaluating **AI inference capacity per second** across various model architectures (CNN, RNN, Transformer) and **service provider CPE hardware platforms available annually**—ranging from **edge SoCs** to **cloud GPU/TPU instances**—a comparison can be made based on current public **cost-per-inference metrics** using **published rates from major cloud vendors** (AWS, Google Cloud, Azure) for 2024–2025.

**Table 47- CNN Model Tokens/Sec and Cost Comparison (2024–2025)**

| Model Type | Hardware Type | Approx. Inferences/sec(Batch 1) | Cloud Service Example | Inference Cost per 1M Inferences (USD) | Notes |
|---|---|---|---|---|---|
| CNN (e.g., MobileNet) | Edge SoC (2.5–10 TOPS) | ~50–500 | On-device | $0 (local) | Power/thermal limited |
| | NVIDIA T4 (GCP/AWS) | ~2,000–5,000 | GCP n1-standard-4 + T4 | ~$0.10–$0.20 | ImageNet-class models |
| | NVIDIA A100 (80GB) | ~10,000–30,000 | AWS p4d.24xlarge | ~$0.02–$0.05 | High batch size + INT8 |
| | TPU v5e | ~20,000–40,000 | GCP TPU | ~$0.03 | Optimal for batch CNN workloads |

**Table 48- RNN/LSTM Model Tokens/Sec and Cost Comparison (2024-2025)**

| Model Type | Hardware Type | Approx. Inferences/sec(Short sequence RNN) | Cloud Service Example | Inference Cost per 1M Inferences (USD) | Notes |
|---|---|---|---|---|---|
| RNN/LSTM (e.g., speech or time-series) | Edge SoC (4–10 TOPS) | ~100–500 | On-device | $0 | Latency bound |
| | NVIDIA T4 | ~3,000 | AWS g4dn.xlarge | ~$0.15 | Audio or stock prediction |
| | A100 | ~10,000+ | Azure Standard_ND96amsr | ~$0.03 | |
| | TPU v5e | ~20,000 | GCP | ~$0.02–$0.04 | Accelerated |

**Table 49- Transformer Model Tokens/Sec and Cost Comparison (2024-2025)**

| Model Type | Hardware Type | Tokens/sec (INT8 Transformer inference) | Cloud Service Example | Inference Cost per 1M Tokens (USD) | Notes |
|---|---|---|---|---|---|
| Transformer (e.g., BERT-base) | Edge SoC (6.8–25 TOPS) | ~2,000–10,000 | N/A | $0 | Local LLM assistant, offline |
| | NVIDIA T4 | ~10,000 | AWS/GCP T4 | ~$0.10–$0.20 | For 110M–300M param models |
| | NVIDIA A100 | ~25,000–100,000 | AWS p4d, Azure ND96 | ~$0.03–$0.06 | LLaMA2-7B quantized |
| | TPU v5p | ~100,000–250,000 | GCP | ~$0.01–$0.03 | Great for batching or streaming |

| Model Type | Hardware Type | Tokens/sec (INT8 Transformer inference) | Cloud Service Example | Inference Cost per 1M Tokens (USD) | Notes |
|---|---|---|---|---|---|
| | Dedicated API (e.g., OpenAI) | N/A | GPT-4 API: ~1,000–2,000* | OpenAI Pricing | $3–$30 per 1M tokens |

**Table 50- Example ML/Training Models Offered by Hyperscalers**

| Instance Type | Provider | GPUs | RAM | vCPUs | Use Case |
|---|---|---|---|---|---|
| g4dn.xlarge | AWS | 1× NVIDIA T4 (16 GB) | 16 GB | 4 | Inference, light training |
| p4d.24xlarge | AWS | 8× NVIDIA A100 (40 GB each) | 1.1 TB | 96 | LLM training, large DL models |
| TPU v5e (1 chip) | Google Cloud | 1× TPU v5e chip (~140 TFLOPs) | ~N/A | N/A | Efficient training/inference |
| Standard_ND96amsr | Azure | 8× NVIDIA A100 (80 GB each) | ~900 GB | 96 | GPT/ViT training, HPC |

**Table 51- AI Cloud Platform Example Costs**

| Provider | Instance Type | GPU | Hourly Rate (USD) | Est. Tokens/sec | Cost per 1M Tokens |
|---|---|---|---|---|---|
| AWS | g4dn.xlarge | T4 | ~$0.52/hr | ~10k | ~$0.15 |
| AWS | p4d.24xlarge | 8×A100 | ~$32/hr | ~800k | ~$0.04 |
| GCP | TPU v5e (1 chip) | TPU | ~$1.5/hr | ~100k–250k | ~$0.02 |
| Azure | Standard_ND96amsr | A100 | ~$31/hr | ~600k | ~$0.05 |
| OpenAI API | GPT-4 | – | – | ~1k–2k | $30 (1M tokens) |

CPE devices can be used more effectively if their implementation challenges and resource limits are addressed.

Entry-level home AI PCs, priced at around $3,000, can run large models but aren't on par with top-tier AI systems. By contrast, chatbot subscriptions may cost $2,400 annually for limited use. This cost comparison supports using local hardware for advanced AI tasks, with a hybrid strategy that processes complex queries in the cloud and simpler ones locally.

To summarize – we can then see some following potential vectors to track

- **On-device inference** (e.g., 2.5–10 TOPS) is **free per inference**, but constrained by memory and thermal limits.
- **Cloud inference** costs can vary from **$0.01 to $0.20 per 1 million tokens**, depending on hardware and batching efficiency.
- **Transformer models are token-rate driven**, so high throughput is crucial to reduce per-token costs.

- **TPUs offer best value** at high throughput for large batches.
- APIs like **OpenAI/GPT** offer convenience but come at a **10–100× markup** compared to self-hosting.

And to look at the first potential candidates to leverage for implementation and savings – the following areas are ones to consider

**Table 52- Cost-Effective Recommendations by Model Type**

| Use Case | Suggested Platform | Reason |
|---|---|---|
| Local wake word / command parsing | Edge SoC (2.5–6 TOPS) | No latency or cloud cost |
| Image classification / video frame AI | A100 or TPU (batch) | High throughput, low unit cost |
| Real-time voice + LLM agent | 25–50 TOPS on-prem or A100 cloud | Needs fast token throughput |
| Multilingual STT or translation | TPU v5e + Whisper | Low cost, high speed |
| Chat/LLM integration at scale | Self-hosted LLaMA3 + batching | Cost-effective below $0.05/1M tokens |

### *6.1.3. Home Edge AI - Focus on CPU Upgrades Too*

The CPU is essential in AI workloads for tasks like pre- and post-processing as well as model orchestration. While most CPE devices use modest CPUs to reduce cost and power use, more advanced ARM cores, such as the Cortex-A55 or A7x series, can help prevent system bottlenecks. For instance, voice assistants need substantial CPU power for dialogue management, and weak CPUs can increase system latency despite fast NPUs. Currently, gateways tend to run quad-core A53/A55 CPUs at 1.5 GHz or higher, while set-top boxes require even more processing capability. The industry is shifting towards high-performance gateways with CPUs over 50,000 DMIPS, like quad-core Cortex-A78 chips exceeding 2 GHz, which some mobile chipsets are expected to reach.

### *6.1.4. Start Internal AI Trials*

Conduct controlled trials to assess AI orchestration methods. For example, test firmware using an STB's NPU for distributed AI processing, or deploying a small local language model in gateways to compare query response rates versus cloud services. Findings will inform production architecture and highlight key metrics like latency and accuracy for cloud integration decisions.

### *6.1.5. Develop AI Orchestration Logic*

Begin developing a hierarchical AI orchestration system to manage where AI inference tasks are executed—locally, on other home devices, at the edge, or in the cloud—based on model availability, latency, device workload, and privacy. Start with a rule-based approach (e.g., process local voice commands if possible; send complex tasks to the cloud), and plan to advance toward a dynamic, meta-AI-powered routing system. Implement a lightweight framework for this decision engine within your CPE software and backend.

### 6.1.6. Privacy & Security Foundations

Prior to deploying AI features, organizations should revise privacy policies and security protocols to address these technologies. It is essential to provide clear transparency to users regarding which data is processed locally, and which is transmitted to the cloud. For any aggregated data that must leave the premises, implement robust safeguards such as differential privacy or anonymization. From a security perspective, regard the AI subsystem as mission-critical: employ secure boot processes, verify models originate from trusted sources to prevent tampering, and evaluate whether new attack vectors—such as the potential exploitation of AI APIs for information extraction or operational disruption—may arise. Integrating trust into AI features from the outset will facilitate smoother user adoption.

### 6.1.7. Marketing as Enhancement, Not Extra (Yet)

Present new AI features as integrated enhancements to existing services instead of paid add-ons. For example, include basic motion detection in new routers or add voice controls to set-top boxes through software updates. This approach reduces customer resistance to fees and allows features to prove their worth. Monetize only clearly distinct offerings, like full home security solutions, from the start. Be cautious about charging for upgrades seen as standard to current equipment. Use early rollout phases to gather usage data (with consent), assess performance, and guide future pricing decisions.

By the end of 2025, several AI-powered features will be deployed for select users. Next-generation routers and set-top boxes will feature NPUs as standard, and an in-house edge AI team will develop software expertise. Both hardware and software platforms will be ready for future expansion.

## 6.2. Mid-Term (2027–2028): Scaling and Optimizing

Service providers are advised to synchronize the development of home edge AI with ongoing device upgrades and evolving customer requirements, while also evaluating prospects arising from emerging home technology innovations.

- Wi-Fi 8 is anticipated to become relevant for service providers in the second half of 2027, with rapid adoption expected throughout 2028.
- The deployment of 10G to 50G PON technologies is projected to accelerate from 2027 onwards.
- Within the next two years, greater clarity is expected regarding advancements in Edge AI, Cloud and Hybrid AI, as well as new architectural approaches and cost optimization for silicon capabilities.
- There will be an increased focus on the implications of DRAM size in CPE devices, particularly in relation to ROI considerations for service providers.
- Developments in smartphone and PC silicon will continue to drive innovation at the Mobile Edge, while also influencing static Home Edge environments.
- By 2030, initial implementation of 6G-level functions in cellular networks is targeted, including significant enhancements in sensing capabilities and foundational integration of AI within proposed solution architectures.

**Phase 1 (by ~2027):** Set Tier 2 Baseline. By 2027, mid- to high-tier CPE devices should meet Tier 2 specs—gateways will require 2.5–25 TOPS performance and 8GB RAM, while premium "home AI hub" models need at least 50 TOPS with 8GB+. Budget devices will feature smaller NPUs (2.5–7 TOPS, 4GB RAM), so nearly all new devices include hardware acceleration. Collaboration with silicon vendors is key to integrating NPUs, especially as mobile and laptop CPUs reach 42 TOPS. Table XX summarizes the

minimum and optimal hardware specs; by 2027, most products should aim for optimal performance to ensure excellent user experiences.

**Phase 2 (2026-2027):** Expand the AI Feature Set Year-over-Year. After completing hardware deployment, we systematically introduce additional AI-driven features on an annual basis. If initial offerings in 2025–26 included voice integration and core security functions, enhancements for 2026 might encompass use cases such as elderly care monitoring, advanced security capabilities (for example, facial recognition alerts), and an interactive learning assistant for children—leveraging a local small language model for TV applications.

Use the platform's modular architecture to add new features through software updates or downloadable AI apps on CPE, minimizing hardware changes. After this phase, consider opening the platform to third parties via an SDK or app store, allowing external developers to deploy AI models at the home edge while maintaining oversight for safety and compliance. This approach encourages innovation and creates new revenue opportunities, such as partner-provided health monitoring services running directly on your devices.

**Phase 3 (2027–2030):** Focus on optimizing edge-cloud integration as residential edge computing expands. Prioritize seamless coordination between local devices and cloud infrastructure, using dynamic load balancing to route AI tasks efficiently based on device capability and network conditions. Maintain transparent operation for users, with similar model architectures on both platforms and a real-time engine selecting optimal processing locations.

At the same time, deploy federated learning so edge devices update the global model with anonymised data, improving intelligence without sharing raw information. By 2028, aim for a hierarchical AI system where compact device models handle quick tasks, while complex jobs go to edge or cloud servers. This approach reduces latency and costs while enhancing system performance.

Throughout these phases, maintain a close eye on ROI. Track the cost savings from offloaded cloud queries and from reduced support calls, as well as new revenue from AI-based services. This will help fine-tune your investments. Also monitor user sentiment and trust – make privacy a selling point of your approach (e.g. "Your data stays in your home").

By 2030, Service Providers aim to transform their CPE fleet into a distributed edge AI platform, enabling advanced services and competitive differentiation. Those who execute this transition effectively can move beyond connectivity, becoming AI service enablers and gaining efficiency and new revenue.

### 6.3. Performance vs. Cost: Finding the Sweet Spot

A key factor in the roadmap is balancing cost and value throughout each stage. Device upgrades for better AI performance drive up CAPEX, while using cloud resources adds to OPEX. The trade-off spans from 2.5 TOPS/2GB devices to cloud-scale solutions exceeding 1000 TOPS. The main question is: when do diminishing returns occur for edge AI performance from a service provider's viewpoint?

**First Principles Analysis:** Tier 2 CPE devices (around 25 TOPS, 4–8 GB) are expected to be the most cost-effective for wide deployment, handling about 70% of routine tasks locally and offering benefits like lower latency, better privacy, and less reliance on cloud services. The small extra investment per device is worthwhile if it supports new services or reduces cloud costs. Upgrading all devices to Tier 3 specs would enable more local processing, but those added use cases are rare and do not justify the higher expense. Beyond a certain point, increasing device power gives limited value; specialised tasks can be handled through cloud computing as needed.

**Cost Tiers vs. Capabilities:** Adding a modest 2–5 TOPS NPU and 4GB RAM to a device introduces minimal bill of materials (BOM) costs—a straightforward decision given the foundational AI features enabled. Scaling up to approximately 25 TOPS and 8GB represents a strategic inflection point that service providers have yet to fully explore. To justify this investment, a comprehensive business case must be developed, assessing which AI use cases merit increased CAPEX and evaluating the potential for cost savings, new revenue streams, or customer acquisition resulting from these enhancements.

Service Provider insertion in the AI Value Chain – this has to tease out out – but one path seems clear for the SPs to

- Build strategic partnerships with Frontier AI players.
- Propose Hybrid AI solutions that combine SP investments in CPE devices with cloud cost reduction for Frontier Model AI or Hyperscalar partners, aiming for 50% of common AI tasks at the edge and 70% for specific SP requirements.
- Address Privacy, Latency, Ethical, and Connectivity needs, adding a Service Provider value layer to partner offerings.
- Bundle AI services with Broadband, Video, and Mobility, emphasizing added value and adopting a collaborative rather than competitive approach, similar to OTT video trends.
- Develop SP-specific AI models leveraging dynamic home data, focusing on holistic home solutions for Broadband, Video, IoT, and related services.
    - The Global Telecom AI Alliance is working on SP-focused models, currently prioritizing cloud but likely to expand into edge-based initiatives as SPs define their AI roles for homes and subscribers.

Mid/high-tier equipment costs are reasonable and can be offset by premium services, customer retention, and cloud savings over a typical 3–5-year device lifespan. However, moving to 100+ TOPS and over 8GB is significantly more expensive with current NPU silicon and DRAM prices, requiring strong business justification. As silicon, DRAM, and AI investments grow, costs for cloud and edge inference should decrease over the next decade, but widespread high-performance Home Edge AI may take 5–10 years to become feasible. If home AI costs, currently $100–$300 per month for cloud-only services, persist, there will be increased efforts toward Capex-based solutions.

**Diminishing Returns Point:** The boundary between Tier 2 and Tier 3 marks where device capabilities—around 50 TOPS and 8 GB—are sufficient for tasks like multi-camera analytics and running small language models. Recent examples, such as the Nvidia DGX Spark system at $3,000 in 2025, can handle 70% of common AI services. This hints at possible Home AI Server setups in the next 5-10 years, depending on household demand for inference. The need will be shaped by how many home devices require AI processing and whether offloading compute to a central server can save costs or enhance features beyond built-in hardware. Although this shift is substantial, rapid advances in computing, power, and latency are reshaping the field. The author highlights several emerging directions and trends to watch.

- AI service costs to consumers will influence Edge Capex; higher monthly fees mean more local investment.
- Real-time, low-latency AI—like in humanoid robots—should ideally process data locally, with some cloud offloading initially.
- For privacy, video inference should happen locally before sending limited data to the cloud; audio files should never be cloud-uploaded due to their sensitive nature.
- As AI automations grow, offline capabilities and local backups will become more important, especially during rare connectivity issues.

Identifying key use cases driving consumer AI adoption, along with their frequency and costs, will help determine how AI services scale for both average and advanced home users. Given growing demand from major providers, investments in computing, power, and connectivity will be maximized by AI. It's important to consider the five-year value of any current capital expenditure.

We do need to be careful however and not follow the ‘shiny object’ of AI. All use cases have to be vetted against

- Edge is best for privacy and latency, while Cloud offers greater flexibility and accuracy.
- Edge AI can reduce costs compared to public cloud or LLM solutions.
- AI sales avatars may boost sales by enhancing customer engagement beyond traditional web and app experiences.
- Attracting new customers requires offering innovative technology and differentiation, which also helps service providers maintain free cash flow growth in a competitive market.

A High Inference Home AI server is a future goal, but service providers should focus on practical Hybrid Edge and Cloud AI solutions. Starting with heavy cloud use and gradually shifting functions to the edge is effective; hybrid architectures balance this shift. For example, aiming for edge devices to handle 70% of daily tasks and letting centralized resources manage the remaining complex 30%. Investing in edge performance beyond ~50 TOPS offers limited benefits for wide deployment—funds are better spent on scalable cloud/edge infrastructure to support heavier workloads.

**Hybrid Architecture:** We propose a practical architecture where small optimized models run on CPE for the majority of interactions, and a tiered cloud-edge backend handles the rest. In this architecture:

**The Home Edge:** The Gateway/STB (and optional sidecar) runs efficient local models—like CNNs or RNNs—for quick tasks such as keyword spotting, simple voice commands, motion detection, and local anomaly alerts. These devices also aggregate and pre-process data, summarize inputs (e.g., flagging “no movement” instead of sending raw video), and serve as proxy points for AI services.

**The Network Edge**: Service provider-owned edge servers in local hubs or central offices run heavier, latency-sensitive models that are too large for home devices. For instance, medium conversational AI or multi-camera processing engines operate here, with servers (50–1000 TOPS) serving many homes on demand. Their proximity ensures minimal added latency (10–20 ms) and cost efficiency by sharing resources across users.

**The Cloud** (hyperscale or centralized data center) handles the most complex tasks and utilizes the latest large models, such as running a 175B-parameter LLM for occasional advanced queries or multi-minute video analysis. Continuous training and global model updates also occur here, with distilled insights sent back to edge models to complete federated learning.

An AI framework managed by the service provider orchestrates all layers, ensuring raw personal data remains local and only essential insights or queries are sent to the cloud. For instance, home cameras process feeds locally, sending minimal events like "person_detected" to the cloud if needed, conserving bandwidth and protecting privacy. Inference is tiered: quick responses happen locally, while the cloud or edge handles heavier processing.

This hybrid approach means service providers invest enough in edge AI for most needs, benefiting latency, privacy, and cost savings, while relying on cloud resources for peak demands or new advances. As home devices grow more capable, more processing will shift local, but balancing resources will remain an ongoing process.

To summarize, we recommend upgrading CPE to Tier 2 AI for most devices and using a tiered edge/cloud system for extra needs. Regularly assess cost-effectiveness to match edge performance with specific use cases. This strategy balances customer demands for privacy and responsiveness with Service Provider ROI.

## 6.4. Hardware Guidelines by 2028

Our analysis of use cases and technology trends indicates that by 2028, new devices should meet these baseline specs to reliably support standard home AI features (consistent with earlier phases):

***Broadband Gateway (premium model):*** ~16–25 TOPS NPU, ~4 GB RAM, quad-core ARM CPU ~20k DMIPS minimum. This would handle multi-camera security feeds and a basic voice assistant concurrently. Mid-range gateways at least 5 TOPS, 2 GB to handle one AI task at a time.

***Set-Top Box / Home Hub:*** ~40+ TOPS, 8 GB RAM for high-end (essentially Tier 2 to low-Tier 3). That supports complex multimodal assistants (e.g. voice + vision) and advanced video features. Standard STBs at least 5–10 TOPS, 4 GB to cover most AV enhancements.

***Dedicated Home AI Server (if offered):*** Only needed for niche enthusiast segment by 2028, because mainstream devices will cover most needs. But if offered, something like 100+ TOPS, 16 GB+ would attract those wanting full GenAI locally (e.g. running a 13B parameter model fully on-premise).

Some applications are limited by memory (e.g., LLM avatars needing over 4 GB RAM), while others require more compute power (e.g., video analytics needing higher TOPS). Devices should match resources to use cases—for instance, AI tutor features need at least 8 GB RAM, since more TOPS can't make up for low memory, whereas security camera hubs benefit from higher TOPS with 2–4 GB RAM being enough. Providers might offer different hardware models, such as a high-TOPS gateway for heavy video use and a standard option for basic needs.

Below we provide Table 4 which summarizes suggested hardware thresholds (minimum vs optimal) for some major AI use cases, based on our earlier discussion:

**Table 53- Suggested Hardware Thresholds for Major Use Cases**

| Use Case | Min HW Required (approx.) | Optimal HW for Best Experience | Resource Bound |
|---|---|---|---|
| LLM-powered Avatar (small 7B model) | ~8 TOPS, 4 GB RAM, 15k DMIPS CPU | 30+ TOPS, 8–16 GB RAM, 50k DMIPS | *Memory-bound* (needs ~4GB+ for model) and compute-bound for generation speed. |
| Full GenAI Assistant (large models) | ~50 TOPS, 16 GB RAM, 50k+ DMIPS | 250+ TOPS, 64+ GB, 100k DMIPS (cloud-scale) | *Heavily memory & compute bound* – essentially requires cloud-level hardware. |
| Home Security AI (multi-camera + faceID) | ~4 TOPS, 2 GB RAM, 10k DMIPS | 16 TOPS, 4 GB RAM, 20k DMIPS | *Compute-bound* (CNNs for detection). Face recognition |

| Use Case | Min HW Required (approx.) | Optimal HW for Best Experience | Resource Bound |
|---|---|---|---|
| | | | adds some CPU load for database matching. |
| Elderly Monitoring (sensors + routine AI) | ~1 TOPS, 0.5 GB, 5k DMIPS (simple) | 4 TOPS, 2 GB, 10k DMIPS (multi-sensor AI) | Compute-bound if including video or complex pattern recognition; otherwise light. |
| Energy Management AI | ~1 TOPS, 0.5 GB, 5k DMIPS | 2 TOPS, 1 GB, 10k DMIPS | Mostly low compute (periodic analysis); slightly memory-bound for data logs. |

The table indicates that small LLM avatars mainly require ample memory (at least ~4 GB), while more TOPS simply speeds up responses. For video security, high TOPS prevents frame drops, and memory matters less aside from a couple GB per camera. Choose device specs accordingly: prioritise RAM for smart assistant hubs, and TOPS for camera hubs.

By 2030, broadband gateways should offer about 16–25 TOPS and 8 GB RAM as standard (Tier 2). Advanced hubs and STBs may provide 40+ TOPS and 8+ GB to support sophisticated multimodal features. Full local GenAI experiences depend on how Generative AI pricing evolves; if monthly costs exceed $100–$200, a Home AI server or upfront investment may be favored. Cloud AI pricing is expected to be reduced due to multi-trillion-dollar investments by companies and government initiatives. Early retail devices such as AMD’s 275+ TOPS models and Nvidia's 1K+ TOPS devices are entering the market for heavy AI use, potentially influencing home architecture. Service providers could invest in offering Home Servers with hybrid cloud options to enhance customer experience. Additionally, Apple has upgraded its MAC line with more powerful M4 processors.

It is essential to consider return on investment (ROI): each specification enhancement should correspond to measurable benefits, such as reduced costs or increased revenue. The objective is to achieve a balanced allocation of resources. For example, investing in an additional 4 GB of RAM for gateways may be justified if it supports a privacy-preserving chatbot feature that attracts new customers or decreases churn. Conversely, implementing an AI sidecar exclusively for feature-driven applications with a defined ROI must be weighed against potentially more cost-effective cloud-based options. The 70/30 rule can serve as a guideline: allocate approximately 70% of AI workload to edge solutions, which deliver the greatest day-to-day user value, while utilizing cloud resources for the remaining 30%, representing more complex tasks.

A hierarchical edge AI strategy helps Service Providers optimize performance, cost, and customer experience. By 2030, those who adopt this approach will turn access devices into a distributed AI cloud, outperforming competitors dependent on hyperscalers. This positions them as trusted AI orchestrators for the home and provides a clear competitive edge. Although planning and investment are required, the benefits include distinct services and improved operations.

## 7. Conclusion

The Service Provider is currently navigating significant changes brought about by advancements in artificial intelligence. Future economic growth across multiple sectors will increasingly derive from engagement within the AI value chain. To avoid being relegated to a mere conduit for AI services and AI-enhanced solutions delivered via their networks, Service Providers should prioritise integrating AI as a central aspect of their strategic planning. A forward-looking vision is required—one that transforms home technology into a cohesive, AI-optimized ecosystem, seamlessly combining cloud resources with on-premises capabilities.

This paper's analysis highlights that requisite hardware and architectural components are emerging to support this transformation. By capitalizing on their unique positioning at the network edge and their access to in-home devices, Service Providers can bridge the divide between cloud-based AI and end-users. This approach enables the delivery of faster, more secure, and reliable AI services directly to consumers' homes.

We examined how targeted investment in Customer Premises Equipment (CPE) upgrades—such as the integration of Network Processing Units (NPUs) and additional memory—can empower Service Providers to deploy a broad spectrum of AI features, ranging from AI-driven avatar assistants to essential security monitoring capabilities. These enhancements also serve to alleviate bandwidth demands and reduce cloud-related expenses.

Equally critical is the implementation of a hybrid architecture in which edge devices manage routine AI tasks, while collaborating seamlessly with network-edge and cloud infrastructure for more complex processing requirements. This hierarchical model optimizes both cost and performance, supporting Service Providers’ return-on-investment objectives and delivering immediate, privacy-focused AI experiences for customers.

It is important to note that the architectural landscape remains dynamic, with rapid advancements expected in silicon development and ROI realization. With Edge AI-capable devices projected for 2025, future targets for metrics such as AI tokens per second, inference rates, and machine learning at the edge will evolve significantly by 2030. The substantial investment in this field highlights the necessity for continued strategic vision and the identification of use cases that justify and sustain further funding.

The balance between edge and cloud computing will continue to develop alongside technological advancements. However, a prevailing principle has emerged: employing a distributed AI strategy—encompassing home, network, and cloud components—offers advantages over a solely cloud-based approach for many new services. Service Providers can significantly benefit by adopting this model, transitioning from merely transmitting data to delivering comprehensive AI-driven solutions. This shift enables them to strengthen customer relationships through personalized and continuous support, generate additional revenue via AI-enhanced premium offerings, and enhance operational efficiency through intelligent automation and workload distribution.

Home Edge AI serves defensive and offensive purposes by tackling privacy, latency, and competition while enabling new services and partnerships. The recommended steps—from upgrading devices and piloting features to developing platform architecture—offer operators a clear roadmap. Early, strategic investment in home-edge AI can give Service Providers a strong advantage in delivering intelligent, integrated subscriber experiences.

Going forward, industry collaboration among operators, device makers, and AI software providers is key to enabling interoperability and supporting a robust home AI ecosystem. Establishing standards for edge model deployment, privacy-preserving data sharing, and resource management will speed up adoption.

Service Providers are positioned to deliver intelligent, responsive home edge solutions that extend cloud capabilities to devices throughout the home. By following the approaches in this paper, they can lead to the expansion of AI in the home and strengthen their role in its advancement.

# Abbreviations

| | |
|---|---|
| AP | access point |
| bps | bits per second |
| FEC | forward error correction |
| HD | high definition |
| Hz | hertz |
| K | kelvin |
| SCTE | Society of Cable Telecommunications Engineers |
| LLM | Large Language Model - AI |
| SLM | Small Language Model - AI |
| CNN | Convolutional Neural Network |
| RNN | Recurrent Neural Network |
| TOPS | Trillions of Operations per Second |
| NPU | Neural Processing Unit |
| GPU | Graphics Processing Unit |
| CPU | Computer Processing Unit |
| CPE | Consumer Premises Equipment |
| LSTM | Long Short-Term Memory – a type of RNN |
| SP | Service Provider |

# Bibliography & References

1. IEA – **Energy Efficiency in Edge vs. Cloud Computing** (2020): Edge computing can reduce energy consumption by up to 30% compared to cloud.

2. ArXiv (2025) – **Hybrid Edge-Cloud Energy Analysis**: Numerical model showing ~65% energy savings per device with 80% workload on edge (674 vs. 1927 kWh/yr); heavy AI workloads could save ~10,000 kWh annually per device by using edge. https://arxiv.org/abs/2501.14823

3. Nutanix Forecast (Nov 2024) – **Can Edge AI Help Sustainability?** Edge avoids "power-hungry data transmission," and allows localized renewables integration for improved carbon footprint. Case study of Dryad wildfire sensors: fully edge AI powered by solar, lowering emissions.

4. ObjectBox (Nov 2024) – **Edge for a Sustainable Future**: Shifting computation to edge can cut 60–90% of data traffic and significantly reduce energy and $CO_2$ emissions. Notes that cloud data centers are already 300 Mt $CO_2$ and growing.

5. Digi (2023) – **How Edge Computing Supports Sustainability**: Edge processing consumes much less energy than centralized data centers and minimizes power-hogging network usage. Many battery-powered edge devices can run for years, highlighting efficiency.

6. NRDC/EnergyStar Data – **Set-Top Box Energy Use**: DVR cable boxes historically draw ~25–40 W continuously, with 50–70% of energy spent in idle standby (hence importance of low-power design for edge devices).

7. IoT Analytics (2024) – **Edge vs Cloud in IoT**: Researchers in *Internet of Things* journal note edge computing's efficiency gains (lower latency, less energy) and its role in sustainable AI systems.