# Best Qwen Models Ranked: Which to Run Locally

February 2, 2026 · by Mark Bartlett

Download this guide as PDF

> **Quick Answer:** Qwen 3 is the strongest open-source model family available right now. At every size from 8B to 32B, Qwen 3 matches or beats the previous tier — the 8B performs like Qwen 2.5's 14B, the 14B like the old 32B, and the 32B like the old 72B. The killer feature is hybrid thinking mode: toggle /think for step-by-step reasoning on hard problems, /no_think for fast chat. Qwen 2.5 Coder 32B still matches GPT-4o on coding benchmarks. And Qwen-VL gives you vision capabilities from 2B to 32B. For most local users, Qwen 3 8B (Q4_K_M, ~5 GB VRAM) is the new default small model, Qwen 3 14B is the 12-16 GB sweet spot, and Qwen 3 32B is the best model you can run on 24 GB. The 30B-A3B MoE is tempting on paper but has GPU utilization issues in Ollama — stick with the dense models for now.

📚 **More on this topic:** Qwen 3.5 Local Guide · Qwen 3.5 9B Setup Guide · Llama 3 Guide · DeepSeek Models Guide · Best Models for Coding · VRAM Requirements

While everyone was talking about Llama and DeepSeek, Alibaba quietly built the best open-source model family in the world.

Qwen 3 beats Llama 3 at every comparable size. Qwen 2.5 Coder 32B matches GPT-4o on coding benchmarks. Qwen-VL handles vision tasks that most local users assumed needed cloud APIs. And unlike DeepSeek's 671B behemoths that need a server rack, Qwen ships practical sizes that run on the GPU you already own.

The problem is the lineup is massive. Qwen 3, Qwen 2.5, Coder, VL, MoE variants, thinking mode, non-thinking mode — it's a lot. This guide maps out every model that matters for local use, what hardware it needs, and when to pick Qwen over the alternatives.

---

## The Qwen Family at a Glance

Qwen isn't one model — it's a sprawling ecosystem. Here's what exists and what actually matters for running locally:

| Family | What It Is | Sizes | Why It Matters |
|---|---|---|---|
| **Qwen 3.5** | Newest generation, multimodal | 0.8B, 2B, 4B, 9B, 27B, 35B-A3B, 122B-A10B, 397B-A17B | New architecture, natively multimodal, beats Qwen 3 at every size |
| **Qwen 3** | Previous generation, general purpose | 0.6B, 1.7B, 4B, 8B, 14B, 32B, 30B-A3B (MoE), 235B-A22B (MoE) | Still strong, huge fine-tune base |
| **Qwen 2.5 Coder** | Code-specialized | 0.5B, 1.5B, 3B, 7B, 14B, 32B | 32B matches GPT-4o on code |
| **Qwen3-VL** | Vision-language | 2B, 4B, 8B, 32B | Best local vision model (being superseded by Qwen 3.5's built-in vision) |
| **Qwen 2.5** | Older gen general | 0.5B–72B | Superseded by Qwen 3 and 3.5 |

Everything is Apache 2.0 licensed — fully open for personal and commercial use. Alibaba trained Qwen 3 on 36 trillion tokens across 119 languages. For context, Llama 3.1 was trained on 15 trillion.

---

## Why Qwen Leads Right Now

Three things set Qwen 3 apart from every other open model family in early 2026:

### 1. Benchmark dominance at every size

Each Qwen 3 model performs like the previous generation one tier up:

| Qwen 3 Model | Performs Like | Implication |
|---|---|---|
| 8B | Qwen 2.5 14B | 14B-class performance in 5 GB VRAM |
| 14B | Qwen 2.5 32B | 32B-class performance in 10-12 GB VRAM |
| 32B | Qwen 2.5 72B | 72B-class performance in 20 GB VRAM |

This isn't marketing — it holds up across MMLU, HumanEval, MATH, and GPQA benchmarks. The Qwen 3 32B at Q4 quantization fits on an RTX 3090 and delivers what previously required a dual-GPU setup.

### 2. Hybrid thinking mode

Qwen 3 can switch between two modes in the same conversation:

- **Thinking mode** (`/think`): The model reasons step-by-step before answering. Uses more tokens, slower, but dramatically better on math, logic, and complex analysis.
- **Non-thinking mode** (`/no_think`): Fast, direct responses for simple questions and chat. Same speed as a standard model.

No other model family offers this toggle. DeepSeek R1 always thinks (and dumps reasoning tokens you don't need for "what's the weather?"). Llama 3 never thinks. Qwen 3 lets you choose per-message.

### 3. 119 languages

Most open models handle English well, maybe a few European languages passably. Qwen 3 was explicitly trained on 119 languages with strong multilingual benchmarks. If you work in Chinese, Japanese, Korean, Arabic, or basically anything besides English, Qwen is the clear choice.

---

## Qwen 3: Every Size Ranked

### 0.6B and 1.7B — Edge Only

These exist for phones, IoT devices, and embedded systems. They're not useful for general conversation or any task requiring real understanding. If you're building something for a Raspberry Pi, they're there. Otherwise, skip them.

```
ollama run qwen3:0.6b
ollama run qwen3:1.7b
```

### 4B — Minimum Viable Model

The 4B is where Qwen 3 starts being genuinely useful. It handles simple Q&A, text formatting, basic summarization, and straightforward coding tasks. Fits on 4 GB VRAM at Q4 quantization.

It's not going to write complex code or handle nuanced reasoning, but for a model that fits on a potato, the quality is surprising.

```
ollama run qwen3:4b
```

**VRAM:** ~2.5 GB at Q4_K_M

### 8B — The New Default Small Model

This is the one most people should start with. Qwen 3 8B outperforms Llama 3.1 8B on nearly every benchmark — MMLU, HumanEval, MATH, GSM8K. It matches what Qwen 2.5 14B could do, but in an 8 GB VRAM envelope.

| Benchmark | Qwen 3 8B | Llama 3.1 8B | Winner |
|---|---|---|---|
| MMLU | 73.8 | 69.4 | Qwen |
| HumanEval | 72.0 | 62.2 | Qwen |
| MATH | 62.8 | 47.2 | Qwen |
| GSM8K | 84.2 | 79.6 | Qwen |

With thinking mode enabled, it gets even better on math and reasoning — at the cost of 2-3x the tokens per response.

```
ollama run qwen3:8b
```

**VRAM:** ~5 GB at Q4_K_M. Runs comfortably on an RTX 3060 12GB with room for context.

### 14B — The Sweet Spot for 16 GB GPUs

If you have 16 GB VRAM (RTX 4060 Ti 16GB, RTX 5060 Ti 16GB, or an older card), the 14B is your model. It performs at the level of Qwen 2.5's 32B — strong coding, solid reasoning, good creative writing.

This is where thinking mode really starts to shine. Toggle `/think` for a coding problem, get step-by-step reasoning. Toggle `/no_think` for casual chat, get instant responses.

```
ollama run qwen3:14b
```

**VRAM:** ~10-12 GB at Q4_K_M. Fits on 16 GB cards with reasonable context windows.

### 32B — Best Model on 24 GB

The Qwen 3 32B is arguably the single best model you can run on a 24 GB GPU. It matches Qwen 2.5 72B performance — a model that previously needed 48+ GB — and fits on a single RTX 3090 or 4090 at Q4 quantization.

For coding, reasoning, analysis, and creative work, this is the local model that makes cloud APIs optional for most tasks.

```
ollama run qwen3:32b
```

**VRAM:** ~20 GB at Q4_K_M. Tight fit on 24 GB but works with moderate context lengths (~4K-8K tokens). At Q3_K_M (~17 GB), you get more context headroom.

### 30B-A3B MoE — Promising but Problematic

On paper, the 30B-A3B is exciting: it's a Mixture of Experts model that only activates 3 billion parameters per token, so it should run fast while having 30B total knowledge. In practice, there's a known Ollama issue (GitHub #10458) where GPU utilization drops significantly compared to dense models.

The full model needs ~19-21 GB at Q4_K_M, which defeats the speed advantage since it's nearly as large as the dense 32B. Until the GPU utilization issue is fixed, stick with the dense models.

```
# You can try it, but expect suboptimal GPU utilization
ollama run qwen3:30b-a3b
```

**VRAM:** ~19-21 GB at Q4_K_M. Not recommended over the dense 32B right now.

### 235B-A22B MoE — Competition for DeepSeek R1

The flagship. A 235B parameter MoE that activates 22B per token. Competitive with DeepSeek R1 and GPT-4o on benchmarks, including a CodeForces ELO of 2056. But at ~140 GB for Q4 quantization, this is multi-GPU territory or heavy CPU offloading.

If you're running a home server with multiple GPUs or substantial RAM for CPU inference, it's the most capable open model available. For everyone else, the 32B dense gets you 80% of the way there on realistic hardware.

```
ollama run qwen3:235b-a22b
```

**VRAM:** ~140 GB at Q4_K_M. Requires multi-GPU or CPU offloading.

---

## Qwen 3.5: Architecture Over Scale

Qwen 3.5 dropped in three waves: the flagship 397B-A17B on February 16, mid-range models (122B-A10B, 35B-A3B, 27B) on February 24, and the small models (9B, 4B, 2B, 0.8B) on March 2, 2026. It's not an incremental update. Alibaba rebuilt the architecture from the ground up, and the results break the "bigger model = better model" assumption.

### What Changed: Gated DeltaNet

Qwen 3.5 replaces the standard transformer attention in most layers with Gated DeltaNet, a form of linear attention. The layout is a 3:1 ratio: three DeltaNet layers for every one full attention layer. This hybrid means:

- Roughly 40% less KV-cache memory at long contexts compared to a standard transformer at 32K
- 262K native context, extendable to 1M with YaRN scaling
- Faster inference at long sequences because linear attention doesn't scale quadratically

All Qwen 3.5 models are also sparse MoE (except the 9B, 27B, and sub-4B dense models) and natively multimodal. Text, images, and video are handled by the same weights through early fusion, not bolted-on vision adapters. The 0.8B can process video on a phone.

### The Headline: 3B Active Beats 22B Active

The 35B-A3B makes the case clearly. It activates only 3B parameters per token out of 35B total. The previous-gen Qwen3-235B-A22B activated 22B out of 235B. The 35B-A3B beats it:

| Benchmark | Qwen 3.5 35B-A3B (3B active) | Qwen 3 235B-A22B (22B active) |
|---|---|---|
| MMLU-Pro | 85.3 | 84.4 |

| Benchmark | Qwen 3.5 35B-A3B (3B active) | Qwen 3 235B-A22B (22B active) |
|---|---|---|
| GPQA Diamond | 84.2 | 81.1 |
| MathVision | 83.9 | 74.6 |
| LiveCodeBench v6 | 74.6 | — |

That's 1/7th the parameters doing more. On an RTX 3090, the 35B-A3B runs at 112 tok/s because only 3B params fire per token. The 235B-A22B can't even load on consumer hardware.

## Qwen 3.5 by Size

**0.8B and 2B** are edge-only. The 0.8B runs video understanding on phones. The 2B handles basic tasks on integrated graphics. Neither is for serious desktop use.

**4B** is the minimum viable multimodal model. At ~2.5GB Q4, it runs on laptop dGPUs and handles code explanations, image reading, and simple chat. LiveCodeBench v6: 55.8, GPQA Diamond: 76.2.

**9B** is the new default for 8GB VRAM. It beats models 13x its size on reasoning (GPQA Diamond 81.7 vs GPT-OSS-120B's 71.5). Fits at 5GB Q4 on Ollama. If you have 8GB, this replaces Qwen 3 8B.

```
ollama run qwen3.5:9b
```

**27B** is a dense model that ties GPT-5 mini on SWE-bench Verified (72.4). At ~16GB Q4, it fits on 24GB cards with room for context. Good for coding, reasoning, and tasks where you want the throughput consistency of a dense model.

**35B-A3B** is the most interesting MoE in the lineup. 112 tok/s on a 3090, beats the previous flagship, and handles agentic coding workflows. Needs ~20GB to load despite only 3B active. Best tool use in its class (BFCL-V4: 66.1 at 9B, 72.2 at 122B-A10B).

**122B-A10B** needs 80GB+ and is the best open model for tool use and function calling (BFCL-V4: 72.2). Practical on an M4/M5 Max with 128GB unified memory.

**397B-A17B** is the flagship at ~214GB Q4. M3 Ultra or multi-GPU territory. Competes with frontier closed models.

## Qwen 3.5 vs Qwen 3

| | Qwen 3 | Qwen 3.5 |
|---|---|---|
| **Architecture** | Standard transformer | Gated DeltaNet + MoE hybrid |
| **Multimodal** | Text only (separate VL models) | Native text + images + video |
| **Context** | 32K (1M with update) | 262K native (1M with YaRN) |
| **Thinking mode** | /think toggle | /think toggle |
| **KV-cache at long context** | Standard (grows with context²) | ~40% less (linear attention) |
| **Best 8GB model** | 8B (~5GB Q4) | 9B (~5GB Q4), much stronger benchmarks |
| **Best 24GB model** | 32B (~20GB Q4) | 27B (~16GB Q4) or 35B-A3B (~20GB Q4) |

For new setups, start with Qwen 3.5. Qwen 3 is still worth keeping if you have existing fine-tunes built on it or need a specific dense model size (14B, 32B) that Qwen 3.5 doesn't offer in dense form.

## Ollama Setup (Requires 0.17.4+)

Qwen 3.5 uses the Gated DeltaNet architecture, which needs Ollama 0.17.4 or later. If you're on an older version, update first.

```
# Small models
ollama run qwen3.5:4b
ollama run qwen3.5:9b

# Mid-range
ollama run qwen3.5:27b
ollama run qwen3.5:35b-a3b

# Check your Ollama version
ollama --version
```

For the full deep dive, see our Qwen 3.5 local guide and Qwen 3.5 9B setup guide.

---

# Thinking Mode: Qwen 3's Killer Feature

Hybrid thinking is what separates Qwen 3 from everything else. Here's how to actually use it.

## How it works

When you type `/think` before a prompt (or in the system prompt), the model generates internal reasoning tokens before its answer. These reasoning tokens are visible — you'll see the model's chain of thought. When you type `/no_think`, it responds directly like a standard model.

## When to use thinking mode

| Use Case | Mode | Why |
|---|---|---|
| Math problems | `/think` | Step-by-step reasoning catches errors |
| Code debugging | `/think` | Model traces through logic systematically |
| Complex analysis | `/think` | Breaks down multi-part questions |
| Quick chat | `/no_think` | No overhead, instant responses |
| Text formatting | `/no_think` | Simple task, thinking wastes tokens |
| Translation | `/no_think` | Direct task, no reasoning needed |

## Controlling thinking in Ollama

**Per-message toggle:**

```
>>> /think What is the integral of x^2 * e^x?
>>> /no_think Translate "hello" to French
```

**Interactive session toggle:**

```
>>> /set think      # Enable thinking for all messages
>>> /set nothink    # Disable thinking for all messages
```

**CLI flag:**

```
ollama run qwen3:8b --think=false  # Start with thinking off
```

### The token cost

Thinking mode generates significantly more tokens — typically 2-5x more than non-thinking mode for the same question. This means slower responses and more memory usage for the KV cache. On hard math problems, the tradeoff is worth it. For chat, it's not.

### Permanent thinking control via Modelfile

If you want a version of Qwen 3 that never thinks (or always thinks), create a custom Modelfile:

```
FROM qwen3:8b

# Never think — fast chat mode
PARAMETER think false

# Or: always think — reasoning mode
# PARAMETER think true
```

```
ollama create qwen3-chat -f Modelfile
ollama run qwen3-chat
```

---

## Qwen 2.5 Coder: Still the Best for Code

Qwen 3 is the better general-purpose model, but for pure coding tasks, Qwen 2.5 Coder 32B remains dominant. It was trained specifically on code and scores 88.4% on HumanEval — matching GPT-4o.

| Model | HumanEval | MBPP | Best For |
|---|---|---|---|
| Qwen 2.5 Coder 32B | 88.4% | 82.5% | Code generation, completion |
| DeepSeek Coder V2 16B | 81.1% | 77.8% | Lighter code tasks |
| Llama 3.1 8B | 62.2% | 68.4% | General + some code |
| Qwen 3 32B (thinking) | ~85% | ~80% | Code with reasoning |

The 32B Coder fits on 24 GB at Q4 quantization. If coding is your primary use case — pair programming, code completion, debugging — this is still the model to run.

Smaller sizes exist (7B, 14B) for tighter VRAM budgets. The 7B Coder is a solid option for 8 GB cards when you need code-specific performance over general chat.

```
ollama run qwen2.5-coder:32b
ollama run qwen2.5-coder:14b
ollama run qwen2.5-coder:7b
```

For day-to-day coding with occasional non-code tasks, Qwen 3 32B with thinking mode is catching up and may be the better all-rounder. For dedicated code generation pipelines, the specialized Coder model still wins.

---

## Qwen-VL: Vision That Actually Works Locally

Qwen3-VL is the most capable vision-language model you can run locally. It handles image description, document reading, chart analysis, GUI interaction, and even video understanding.

| Size | VRAM (Q4) | Context | Best For |
|---|---|---|---|
| 2B | ~2 GB | 256K | Phone/edge, basic OCR |
| 4B | ~3 GB | 256K | Lightweight image tasks |
| 8B | ~6 GB | 256K | General vision, documents |
| 32B | ~20 GB | 256K | Complex analysis, GUI agent |

The 8B is the practical choice for most users. It handles document OCR, image Q&A, and chart interpretation well on a 12-16 GB GPU. The 32B is genuinely impressive — it can act as a GUI agent, understanding screenshots and suggesting interactions — but needs 24 GB.

All sizes support 256K native context (expandable to 1M with the Qwen3-2507 update), which is relevant for processing long documents with embedded images.

```
ollama run qwen3-vl:8b
```

**Using vision in Ollama:**

```
# Describe an image
ollama run qwen3-vl:8b "What's in this image?" ./photo.jpg

# Or from the interactive prompt
>>> What does this diagram show? [image: /path/to/diagram.png]
```

Compared to Llama 3.2 Vision 11B, Qwen3-VL 8B is competitive while using less VRAM. The 32B significantly outperforms Llama's vision models on complex tasks.

---

## VRAM Requirements

Here's what you actually need for each Qwen model at practical quantization levels:

| Model | Q4_K_M | Q5_K_M | Q8_0 | FP16 |
|---|---|---|---|---|
| **Qwen 3.5 0.8B** | ~500 MB | ~600 MB | ~900 MB | ~1.6 GB |
| **Qwen 3.5 2B** | ~1.5 GB | ~1.8 GB | ~2.5 GB | ~4 GB |
| **Qwen 3.5 4B** | ~2.5 GB | ~3 GB | ~4.5 GB | ~8 GB |
| **Qwen 3.5 9B** | ~5 GB | ~6 GB | ~9.5 GB | ~18 GB |
| **Qwen 3.5 27B** | ~16 GB | ~19 GB | ~28 GB | ~54 GB |
| **Qwen 3.5 35B-A3B** | ~20 GB | ~24 GB | ~36 GB | ~70 GB |
| **Qwen 3.5 122B-A10B** | ~70 GB | ~81 GB | ~122 GB | ~244 GB |
| **Qwen 3.5 397B-A17B** | ~214 GB | ~250 GB | ~397 GB | ~794 GB |
| Qwen 3 0.6B | ~0.5 GB | ~0.6 GB | ~0.8 GB | ~1.2 GB |
| Qwen 3 1.7B | ~1.2 GB | ~1.4 GB | ~2 GB | ~3.4 GB |
| Qwen 3 4B | ~2.5 GB | ~3 GB | ~4.5 GB | ~8 GB |
| Qwen 3 8B | ~5 GB | ~6 GB | ~9 GB | ~16 GB |
| Qwen 3 14B | ~10 GB | ~12 GB | ~16 GB | ~28 GB |
| Qwen 3 32B | ~20 GB | ~24 GB | ~36 GB | ~64 GB |

| Model | Q4_K_M | Q5_K_M | Q8_0 | FP16 |
|---|---|---|---|---|
| Qwen 3 30B-A3B | ~19 GB | ~22 GB | ~34 GB | ~60 GB |
| Qwen 2.5 Coder 32B | ~20 GB | ~24 GB | ~36 GB | ~64 GB |
| Qwen3-VL 8B | ~6 GB | ~7 GB | ~10 GB | ~18 GB |

**Which quantization?** Q4_K_M is the default recommendation — best balance of quality and size. Move to Q5_K_M if you have the VRAM to spare. Drop to Q3_K_M only if you're truly VRAM-constrained. More on quantization.

### What to run on your GPU

| GPU | VRAM | Best Qwen Model | Ollama Command |
|---|---|---|---|
| RTX 3060 / 4060 | 8 GB | **Qwen 3.5 9B** Q4 | `ollama run qwen3.5:9b` |
| RTX 3060 12GB | 12 GB | Qwen 3.5 9B Q6-Q8 | `ollama run qwen3.5:9b` |
| RTX 4060 Ti 16GB | 16 GB | Qwen 3.5 9B Q8 or Qwen 3 14B Q4 | `ollama run qwen3.5:9b` |
| RTX 3090 / 4090 | 24 GB | **Qwen 3.5 27B** Q4 or **35B-A3B** Q4 | `ollama run qwen3.5:27b` |
| 2x RTX 3090 | 48 GB | Qwen 3.5 27B Q8 or Qwen 3 32B FP16 | `ollama run qwen3.5:27b` |
| Mac M4 Pro 48GB | 48 GB | Qwen 3.5 35B-A3B Q4 | `ollama run qwen3.5:35b-a3b` |
| Mac M4/M5 Max 128GB | 128 GB | Qwen 3.5 122B-A10B Q4 | `ollama run qwen3.5:122b-a10b` |

→ Use our Planning Tool to check exact VRAM for your setup.

---

## Qwen vs the Competition

### Qwen 3 8B vs Llama 3.1 8B

Qwen 3 wins across the board. The gap is largest on math (MATH: 62.8 vs 47.2) and coding (HumanEval: 72.0 vs 62.2). Llama 3.1 8B's advantage is the massive fine-tune ecosystem — if you need Dolphin, Hermes, or other community tunes, they're all built on Llama. For the base model, Qwen 3 is the better choice.

### Qwen 3 32B vs Llama 3.3 70B

This is the interesting comparison. Llama 3.3 70B is technically stronger on some benchmarks, but it needs ~43 GB VRAM versus Qwen 3 32B's ~20 GB. Per-VRAM-gigabyte, Qwen 3 32B offers dramatically better value. If you have 24 GB, Qwen 3 32B is your model. If you have 48+ GB, Llama 3.3 70B is worth considering.

### Qwen 3 vs DeepSeek R1 Distills

DeepSeek R1 distills still lead on pure math and formal reasoning (R1-Distill-Qwen-14B scores 69.7% on AIME 2024). But they always output reasoning tokens, making them slower for casual use. Qwen 3's thinking toggle means you get the reasoning when you need it and fast responses when you don't. For a single all-purpose model, Qwen 3 wins. For dedicated math/logic pipelines, the R1 distills are still competitive.

### Qwen 2.5 Coder vs DeepSeek Coder V2

Qwen 2.5 Coder 32B (88.4% HumanEval) beats DeepSeek Coder V2 16B (81.1%) convincingly, but it's also 2x the size. At the same model size, they're closer. The Coder 32B needs 24 GB; if you only have 16 GB, DeepSeek Coder V2 16B is the pragmatic choice.

---

## Setup Guide

### Basic Ollama setup

Install Ollama if you haven't, then pull the model for your GPU:

```
# 8 GB VRAM
ollama run qwen3:8b

# 12-16 GB VRAM
ollama run qwen3:14b

# 24 GB VRAM
ollama run qwen3:32b

# Coding-focused
ollama run qwen2.5-coder:32b
```

```
# Vision
ollama run qwen3-vl:8b
```

## Custom Modelfile for optimized settings

Create a file called `Modelfile`:

```
FROM qwen3:8b

# Set context length (adjust based on VRAM)
PARAMETER num_ctx 8192

# Temperature for creative tasks
PARAMETER temperature 0.7

# System prompt
SYSTEM """You are a helpful coding assistant. Be concise and provide working code examples."""
```

```
ollama create my-qwen -f Modelfile
ollama run my-qwen
```

## Choosing the right quantization

```
# Default (Q4_K_M) — recommended
ollama run qwen3:8b

# Higher quality if you have VRAM
ollama pull qwen3:8b-q5_K_M

# Maximum quality
ollama pull qwen3:8b-q8_0

# Lower quality for tight VRAM
ollama pull qwen3:8b-q3_K_M
```

### Using with Open WebUI

If you're running Open WebUI, Qwen models work out of the box. Pull the model in Ollama, and it appears in the model dropdown automatically. Thinking mode tokens are displayed in a collapsible section in most UIs.

---

## Common Problems

### Thinking tokens are verbose

Qwen 3's thinking mode can generate hundreds of tokens of reasoning before answering a simple question. Use `/no_think` for straightforward queries, or create a Modelfile with `PARAMETER think false` for a permanently non-thinking version.

### Which Qwen should I pick?

If you're overwhelmed by the lineup:

- **Just chatting?** Qwen 3 at whatever size fits your GPU
- **Coding?** Qwen 2.5 Coder 32B (24 GB) or 14B (16 GB)
- **Images/documents?** Qwen3-VL 8B
- **Math/reasoning?** Qwen 3 with `/think` enabled
- **Non-English?** Qwen 3 at any size — best multilingual support

### MoE models running slow

The 30B-A3B MoE has a known GPU utilization issue in Ollama (tracked in GitHub issue #10458). Until this is resolved, dense models (8B, 14B, 32B) will give you better real-world performance despite theoretically lower parameter counts.

### Context length limits

Qwen 3 supports long contexts (32K+ natively, 1M with Qwen3-2507 update), but your actual usable context depends on VRAM. Each token in the KV cache costs memory. At Q4_K_M:

- 8B with 8K context: ~6 GB total
- 14B with 8K context: ~12 GB total
- 32B with 4K context: ~21 GB total

If you're hitting out-of-memory errors, reduce `num_ctx` in your Modelfile before dropping quantization quality.

### Qwen 2.5 vs Qwen 3 — when to keep the older model

Qwen 3 supersedes Qwen 2.5 for general use. The exceptions:

- **Qwen 2.5 Coder**: Still the best for pure code tasks
- **Qwen 2.5 72B**: If you have the VRAM for it, the 72B hasn't been fully replaced until you step up to the 235B MoE
- **Existing fine-tunes**: Community fine-tunes built on Qwen 2.5 haven't all been ported to Qwen 3 yet

For everything else, Qwen 3 is the upgrade.

---

## Bottom Line

Qwen 3.5 is the model family to beat in March 2026. The architectural shift to Gated DeltaNet and native multimodal changes what's possible at every VRAM tier. The 9B fits in 5GB and outscores models 13x its size. The 35B-A3B with only 3B active beats the previous flagship that needed 7x the parameters. Architecture won over scale.

For new setups: Qwen 3.5 9B at 8GB, Qwen 3.5 27B or 35B-A3B at 24GB. Qwen 3 still matters for fine-tuned variants and dense model sizes (14B, 32B) not yet available in 3.5. Qwen 2.5 Coder 32B remains the best for FIM autocomplete. Llama still has the fine-tune ecosystem edge.

```
# The new defaults
ollama run qwen3.5:9b     # 8GB VRAM
ollama run qwen3.5:27b    # 24GB VRAM
```



---