Post-Training a 9B Open Model to Read Wholesale Receipts Better than GPT-5.2 at a Fraction of the Cost

Sahil Mahendrakar Suhaas Katikaneni
Cerebellum Labs

October 2026

Abstract

A restaurant back-office software company used GPT-5.2 to turn photos of supplier receipts into structured records for ingredient costing. We post-trained Qwen3.5-9B with LoRA on public and synthetic receipts only, with no customer data in training. On 810 held-out ReceiptBench receipts the tuned model gets 73% of receipts fully correct against 54% for GPT-5.2, answers in 4.4 s median against 8.7 s, and costs $0.28 per thousand receipts on a fully loaded GPU ($2.78 when serving one request at a time), against $15.30 at GPT-5.2's standard price. On the customer's own long receipt photos it reads the whole photo in one call and finds every line item, where the customer's GPT-5.2 pipeline finds 75%.

1 The task

The customer computes ingredient costs from photos of wholesale supplier receipts. Each photo becomes a JSON record with seven header fields (vendor, invoice number, date, items count, subtotal, tax, total) and ten fields per line (SKU, description, pack size, quantity, unit type, unit price, price basis, extended amount, catch weight, category). A receipt is only useful when every money field is right. The schema also carries conventions: weighed items are priced per pound, descriptions exclude the brand, a subtotal is reported only when printed, and repeated identical rows are separate lines.

Wholesale receipts are long and photographed tall. The customer's pipeline therefore cut each photo into a header crop, overlapping tiles and a band, sent each piece to GPT-5.2 and merged the answers, with a whole-photo retry when too few rows came back. That took six to seven calls per receipt, about $0.16 and 50 s, and the merge collapsed identical repeated rows.

2 Method

Data. We relabeled 4,632 purchase and dining receipts from the public ReceiptBench dataset [1] into the customer's schema with a large open-weight teacher (Kimi K3 [2]), checked that every label's line amounts add up to its total, and relabeled failures from the photo. After a hand check we tightened the prompt, relabeled 2,149 receipts and removed 566 duplicate photos, some of which spanned train and test. The cleaned labels are 99.4% correct on priced-line fields in a hand-checked sample. ReceiptBench receipts are short (median three lines), so we added 1,500 synthetic wholesale receipts with exact labels (4–80 rows, weighed and repeated rows, three printer styles, blur, rotation and crops). They follow the customer's conventions but use none of their items, prices, prompts or photos.

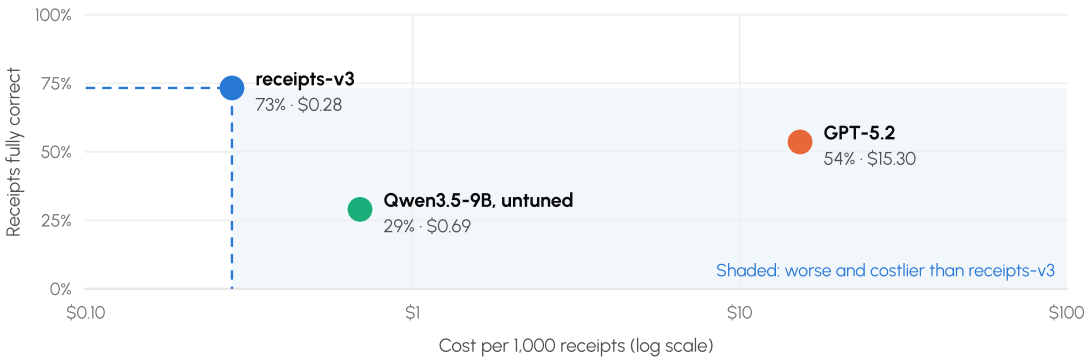


Figure 1: Accuracy against cost on 810 held-out ReceiptBench receipts. The tuned model is both the most accurate and the cheapest.

Training. We fine-tuned Qwen3.5-9B [3] with LoRA [4] (rank 32) on every language-model layer, vision encoder frozen, for 2 epochs on 2,880 ReceiptBench and 1,500 synthetic receipts. Targets are compact single-line JSON. Training took 7.8 hours on one NVIDIA L40S. We call the result *receipts-v3*.

Serving. We serve the model with vLLM [5] using FP8 weights and speculative decoding [6] from the multi-token-prediction head [7] that ships with Qwen3.5, drafting four tokens per step. We did not tune the head, yet 97.5% of its drafts are accepted, because compact JSON is highly predictable. Together these cut median latency from 14.9 s to 4.4 s with no change in accuracy. The model reads each photo whole, in one call, on one GPU.

Evaluation. Every model gets the same prompt, photos and scorer; GPT-5.2 runs with reasoning off at temperature 0. A receipt is *fully correct* when it has the right number of lines and every line's quantity, unit price and amount, plus the subtotal, tax and total, are right. Open-model cost assumes one L40S at $1.86 per hour serving 32 requests at once (6,676 receipts per hour); serving one request at a time it reads 668 per hour, ten times the cost per receipt; GPT-5.2 cost is the standard API price.

3 Results

Public benchmark. Fine-tuning takes Qwen3.5-9B from well behind GPT-5.2 to well ahead of it (Figures 1 and 2). Receipts fully correct rise from 29% to 73% against GPT-5.2's 54%, and field F1 from 0.862 to 0.954 against 0.914. The tuned model leads on every field and predicts the right number of lines on 90% of receipts against 82%. It is twice as fast per request and between 5× (one request at a time) and 55× (a fully loaded GPU) cheaper per receipt. Part of the lead on this set comes from conventions the model learned from the teacher's labels, so we also test on hand-labeled customer receipts.

Customer receipts. Figure 3 compares receipts-v3, reading the whole photo in one call, with GPT-5.2 inside the customer's pipeline. The tuned model finds all 125 line items and gets the money right on 98% of them; the pipeline finds 75%. GPT-5.2 returns no rows on one photo and invents rows and misreads the total on another, identically across three runs, and the pipeline's retry rule does not recover them. The tuned model reads SKUs correctly almost three times as often (74% against 26%), since the pipeline's tiles lower the resolution of each character. It is also

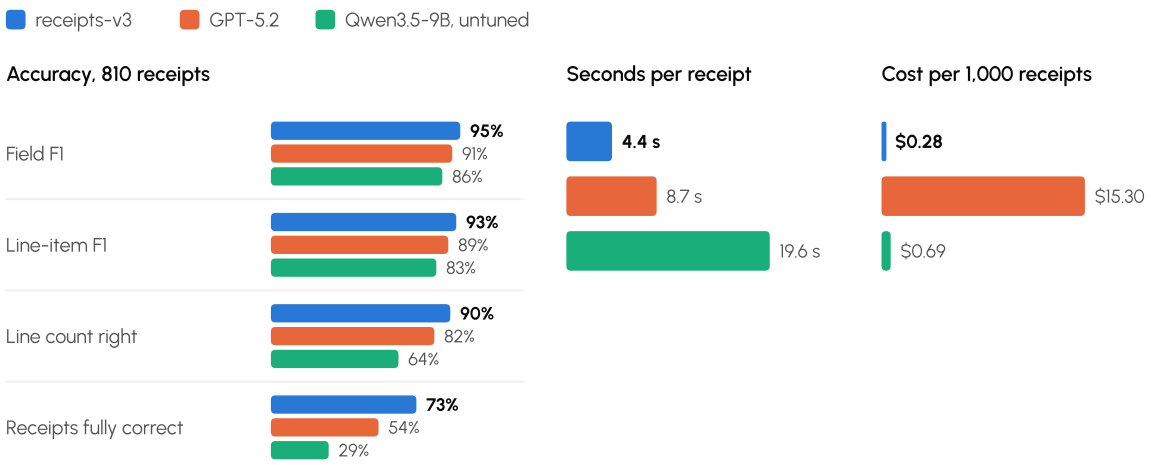


Figure 2: ReceiptBench test set, 810 receipts. Seconds per receipt is the median, one request at a time; the tuned model's latency was measured on the previous checkpoint, which has the same base and serving setup.

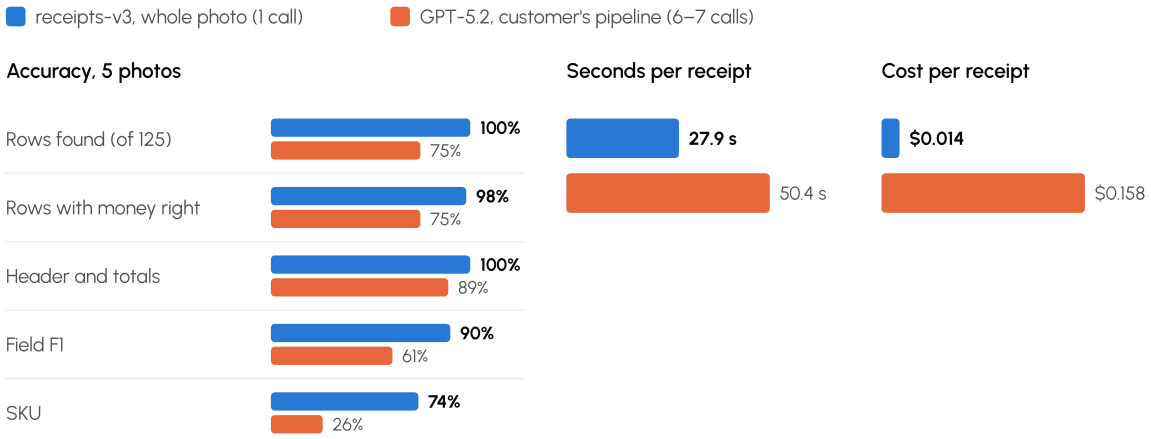


Figure 3: The customer's own receipts: 5 photos of 4 long receipts, 125 line items, hand-labeled and never used in training. A missed line counts as wrong.

faster (27.9 s against 50.4 s on these long receipts) and about 11× cheaper per receipt even when serving one request at a time. The synthetic receipts matter here: a model trained on ReceiptBench alone reaches field F1 0.767 on these photos, and adding them raises it to 0.902 with no loss on ReceiptBench.

4 Limitations and conclusion

ReceiptBench scores are measured against teacher labels, so the tuned model shares the teacher's conventions, and the customer test set is small (125 lines). We did not compare other frontier models or reasoning settings.

Even so, a 9B open model fine-tuned with LoRA on about 4,400 public and synthetic receipts on one GPU beats GPT-5.2 on this task in accuracy, latency and cost, replaces a multi-call pipeline with one call, and never saw the customer's data. It gets 73% of receipts fully correct against 54% for GPT-5.2, answers in half the time (4.4 s against 8.7 s median), and costs 5× to 55× less depending on GPU load.

References

[1] ReceiptBench: A benchmark for structured receipt understanding. arXiv:2605.22413, 2026.
[2] Moonshot AI. Kimi K3. Model release, 2026.
[3] Qwen Team. Qwen3.5 technical report. 2026.
[4] E. J. Hu et al. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022.
[5] W. Kwon et al. Efficient memory management for large language model serving with PagedAttention. In *SOSP*, 2023.
[6] Y. Leviathan, M. Kalman, and Y. Matias. Fast inference from transformers via speculative decoding. In *ICML*, 2023.
[7] F. Gloeckle et al. Better & faster large language models via multi-token prediction. In *ICML*, 2024.