Continual Model Routing in Evolving Model Hubs

Jack Bell[1] Giacomo Carfi[1] Gerlando Gramaglia[1] Vincenzo Lomonaco[2]

[1]*University of Pisa* [2]*LUISS University, Rome*

jack.bell@phd.unipi.it

Motivation

How should we route a user request to the right model when model hubs contain thousands of rapidly evolving experts?

- Existing routers are inherently static
- Formalise routing as a continual learning problem
- CARvE learns a contrastive embedding that routes queries to models while anchoring past knowledge via checkpoint replay.

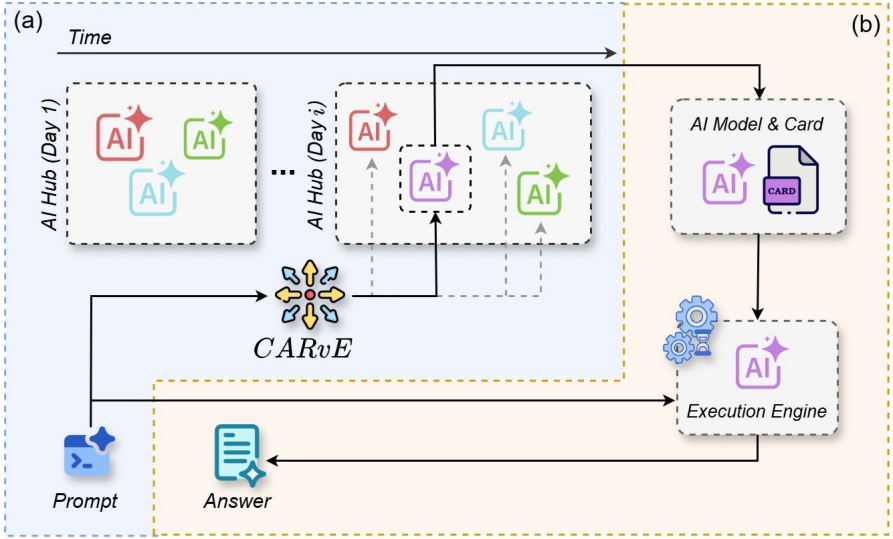


Figure 1: (a) Our adaptive router CARvE learns continually from selection samples to dynamically route a prompt to the most appropriate model. (b) Actual execution of the model (not the focus of this paper)

CMR Formulation

Formulation

Given a growing hub $\mathcal{H}$ and a stream of experiences $\mathcal{E}_1, \ldots, \mathcal{E}_n$, each providing query–model pairs $D_i = \{(q, m^*)\}$, learn a router $f_\theta : \mathcal{Q} \to \mathcal{H}$ that maximises routing accuracy across all experiences, with access only to D_i at step i and without executing any $m \in \mathcal{H}$.

⚠️ Scale

Hubs contain millions of models - routing must remain efficient as the hub grows.

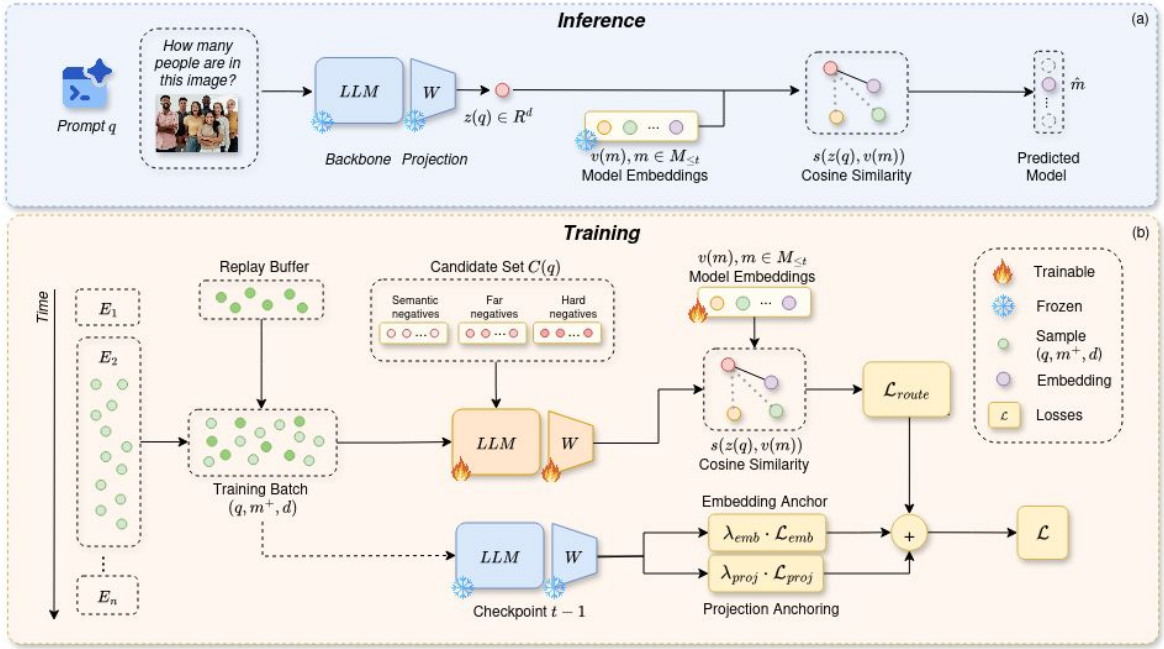
⚡ Forgetting

Sequential training on new domains causes catastrophic forgetting of prior routing decisions.

🎯 Label Shift

Each new experience introduces new model families and query distributions, shifting the label space.

CARvE: Method Overview



① Query Prompt | **② CARvE Router** | **③ Selected Model**

CMRBench: Experimental Setup

Datasets	Baselines	Metrics
• APIBench	• Sequential Fine-tuning	• Model-ID Acc (M-Acc)
• ToolMMBench	• SFT with Random Replay (10%)	• Family Acc (F-Acc)
• HuggingBench	• EWC (regularisation)	• Domain Acc (D-Acc)
• 4 experiences · 2,000+ models	• Cumulative (upper bound)	• Forgetting

💡 Implementation: PyTorch + TRL framework · GPU: H100 (×1) · Code available at: https://github.com/collagelab/CMR

Main Results

80.7%
Domain Accuracy
(CARvE, 10% replay)

5.9%
Domain Forgetting

+4.8pp
over Random Replay D-Acc

Setting	Accuracy (%) (↑)			Forgetting (%) (↓)		
	M-Acc	F-Acc	D-Acc	M-Fgt	F-Fgt	D-Fgt
Retrieval / LLM-controller baselines						
BGE-M3 (Chen et al., 2024a)	**13.6 ± 0.0**	**16.2 ± 0.0**	44.0 ± 0.0	2.5 ± 0.0	3.9 ± 0.0	3.3 ± 0.0
Gorilla RAG (Patil et al., 2024)	6.7 ± 0.2	10.4 ± 0.2	43.0 ± 0.5	**0.0 ± 0.2**	**0.8 ± 0.2**	**0.1 ± 0.2**
HuggingGPT Qwen3-32B (Shen et al., 2023)	–	–	**51.7 ± 0.2**	–	–	–
Continual learning experiments						
Random Replay (5%)	36.5 ± 0.4	44.2 ± 0.4	70.2 ± 4.2	19.5 ± 0.5	23.2 ± 0.5	24.0 ± 4.4
Random Replay (10%)	39.1 ± 0.5	47.3 ± 0.7	75.9 ± 0.2	14.0 ± 0.1	16.8 ± 0.3	13.1 ± 0.2
Random Replay (20%)	41.3 ± 0.2	49.8 ± 0.2	78.1 ± 0.1	8.6 ± 0.4	11.1 ± 0.3	7.8 ± 0.1
Random Replay Qwen2.5-7B (10%)	40.7 ± 0.5	49.2 ± 0.4	78.2 ± 0.2	14.2 ± 0.6	16.5 ± 0.6	10.3 ± 0.5
Random Replay Qwen3-4B (10%)	39.6 ± 0.2	47.9 ± 0.1	77.2 ± 0.2	14.9 ± 0.2	18.2 ± 0.4	11.2 ± 0.4
Sequential Finetuning	28.0 ± 0.1	34.8 ± 0.1	64.3 ± 0.2	34.6 ± 0.2	41.8 ± 0.2	37.2 ± 0.2
Model Merging (TIES) (Yadav et al., 2023)	7.6 ± 0.3	10.9 ± 0.2	28.6 ± 0.5	15.0 ± 0.2	19.7 ± 0.1	32.6 ± 0.2
LwF (Li & Hoiem, 2018)	28.8 ± 0.3	35.9 ± 0.5	56.4 ± 0.2	20.8 ± 0.3	25.7 ± 0.2	39.5 ± 0.2
EWC (Kirkpatrick et al., 2017)	31.3 ± 0.4	38.4 ± 0.4	66.2 ± 0.1	26.6 ± 0.3	32.7 ± 0.2	31.4 ± 0.5
CARvE EWC (10% replay)	42.2 ± 2.4	47.7 ± 2.3	80.5 ± 2.2	9.1 ± 1.9	9.8 ± 3.3	6.1 ± 4.8
CARvE (5% replay)	40.5 ± 0.7	45.4 ± 0.6	78.5 ± 0.4	15.9 ± 1.7	17.4 ± 1.1	10.1 ± 0.7
CARvE (10% replay)	43.1 ± 0.2	48.5 ± 0.3	80.7 ± 0.0	11.4 ± 0.4	12.5 ± 0.3	5.9 ± 0.3
CARvE Qwen3-4B (10% replay)	36.7 ± 3.4	42.1 ± 3.9	78.6 ± 3.3	13.5 ± 3.3	14.3 ± 4.5	9.1 ± 5.0
CARvE Qwen2.5-7B (10% replay)	42.4 ± 0.1	48.1 ± 0.3	81.5 ± 0.3	11.2 ± 0.2	12.3 ± 0.1	6.2 ± 0.6
CARvE (20% replay)	**46.4 ± 0.1**	**51.9 ± 0.2**	**82.9 ± 0.2**	**6.4 ± 0.8**	**7.3 ± 0.8**	**3.0 ± 0.3**

Ablation Study

Method	D-Acc (%)	D-Fgt (%)	Δ D-Fgt
CARvE (full)	**80.7**	**5.9**	—
No emb. anchor	78.4	7.4	+1.5
No proj. anchor ★	74.2	9.4	+3.5
Domain emb. initialisation	78.8	10.2	+4.3
Sequential FT	64.3	35.7	+29.8

▸ Projection anchor is the most critical component, removing it raises D-Fgt from 5.9 to 9.4 (+3.5 pp forgetting). Random embedding initialisation is also an important design choice to avoid constraining learning.

Conclusion

Key Takeaways

- We formalised Continual Model Routing (CMR) - selecting the right model from a growing hub without forgetting past domains.
- CARvE achieves 80.7% domain accuracy on CMRBench with only 5.9% forgetting
- CMRBench is released as a new evaluation suite for the growing model-routing research community.

Future Work

Hybrid CARvE with retrieval

Post inference routing

Theoretical analysis of the contrastive anchoring objective

Thank You!

📄 Paper: https://arxiv.org/abs/2605.28577

💻 Code: https://github.com/collagelab/CMR

📧 Email: jack.bell@phd.unipi.it

🌐 Lab: vincenzolomonaco.com/collagelab