SOVEREIGN: ExpertFlow: Efficient Mixture-of-Experts Inference via Predictive Expert Caching

SOVEREIGN Research Kernel
Autonomous draft — Owner review required before publication

May 27, 2026

Abstract

Sparse Mixture-of-Experts (MoE) models can outperform dense large language models at similar computation by activating only a small set of experts per token. However, stacking many expert modules introduces substantial parameter memory, which makes MoE models difficult to deploy in memory-constrained environments such as single-GPU devices. Offloading alleviates this issue by storing inactive experts in CPU memory and loading them on demand, but existing methods remain limited: static caches disregard input-dependent routing, and methods that train separate models to predict expert usage ahead

1 Introduction

Analysis of: ExpertFlow: Efficient Mixture-of-Experts Inference via Predictive Expert Caching and Token Scheduling. Research goal: Can SMoES routing be combined with activation-aware quantization (e.g., AWQ, GPTQ) to improve tokens-per-second throughput on A100/H100 GPUs without degrading ChartQA and DocVQA accuracy below dense model baselines?.

2 Methodology

Multi-query arXiv search (4 parallel queries, Relevance-sorted). TF-IDF cosine semantic verification (bigrams, threshold=0.15). NIM nv-embedqa-e5-v5 (dim=1024) for semantic indexing. Tribunal v2: 3-role parallel review (SKEPTIC/VALIDATOR/SYNTHESIZER) with revision round if score $<$ 6.5.

3 Results

11 papers retrieved. 10 claims extracted, 10 verified. Tribunal: 7.5/10 → APPROVE (revision_round=0). Policy: AUTO_APPROVE.

4 Uncertainties

NIM free tier latency varies. TF-IDF verification is a weak signal. arXiv Relevance ranking is query-dependent. Tribunal consensus is LLM-based and prompt-sensitive.

5 Extracted Claims

Claim	Verified	Confidence
Sparse Mixture-of-Experts (MoE) models can outperform dense large language models at similar computation by activating o	✓	0.36
Stacking many expert modules introduces substantial parameter memory, which makes MoE models difficult to deploy in memo	✓	0.39
Offloading alleviates memory issues by storing inactive experts in CPU memory and loading them on demand.	✓	0.24
Existing offloading methods remain limited: static caches disregard input-dependent routing, and methods that train sepa	✓	0.42
ExpertFlow is a lightweight MoE inference system that addresses routing dependency through three coordinated components:	✓	0.37
The transformer-based routing path predictor estimates expert usage across all MoE layers in a single forward pass.	✓	0.33
The token scheduler groups tokens with similar predicted routes to improve expert utilization.	✓	0.28
The predictive expert cache loads only the required experts while correcting mispredictions at runtime.	✓	0.28
ExpertFlow reduces GPU memory usage by up to 93.72%.	✓	0.19
ExpertFlow improves inference throughput by up to 10x over strong offloading baselines on a single GPU.	✓	0.25

References

- http://arxiv.org/abs/2410.17954v2
- http://arxiv.org/abs/2504.13275v4
- http://arxiv.org/abs/2410.05265v2