Exploiting the Experts: Unauthorized Compression in MoE-LLMs

Pinaki Prasad Guha Neogi
Ohio State University
Columbus, Ohio, US
guhaneogi.2@osu.edu

Ahmad Mohammadshirazi
Ohio State University, Flairsoft
Columbus, Ohio, US
mohammadshirazi.2@osu.edu

Dheeraj Kulshrestha
Flairsoft
Columbus, Ohio, US
dheeraj@flairsoft.net

Rajiv Ramnath
Ohio State University
Columbus, Ohio, US
ramnath.6@osu.edu

Abstract

Mixture-of-Experts (MoE) architectures are increasingly adopted in large language models (LLMs) for their scalability and efficiency. However, their modular structure introduces a unique vulnerability: adversaries can attempt to compress or repurpose models by pruning experts and cheaply fine-tuning the remainder, effectively bypassing licensing and security constraints. In this paper, we systematically study the prunability of MoE-LLMs under task-specific usage. We first develop an expert attribution framework that identifies the subset of experts most responsible for a given task, then evaluate the performance trade-offs of pruning and re-aligning these experts using active learning-driven fine-tuning. Our findings reveal a critical knowledge loss–recovery trade-off: while certain experts can be isolated to retain task accuracy, significant degradation occurs without targeted re-alignment. Based on this analysis, we propose defense strategies that make MoE models inherently *Un-Compressible* and *Un-Finetunable*, such as entangled expert training and selective fine-tuning protocols that resist unauthorized adaptation. By positioning expert pruning as both a threat vector and a defense target, this work highlights the dual-use nature of MoE modularity and provides the first systematic evaluation framework for secure specialization of MoE-LLMs.

1 Introduction

Large Language Models (LLMs) based on Mixture-of-Experts (MoE) architectures have achieved state-of-the-art results while offering improved computational scalability [4, 5, 10]. In MoE designs, a gating mechanism routes tokens to a sparse subset of experts, enabling large effective capacity with limited runtime cost. However, this modularity creates a novel vulnerability: adversaries may attempt to prune away unused experts, retain only those relevant for a desired task, and cheaply fine-tune the remaining experts. Such *unauthorized compression attacks* threaten both intellectual property (IP) protection and safety guarantees [3, 18]. A more detailed review of prior work on MoE architectures, pruning-based compression, and active learning is provided in Appendix A.

The Lock-LLM workshop calls for strategies to make LLMs *Un-Distillable, Un-Finetunable, Un-Compressible, Un-Editable, and Un-Usable* [1]. In this paper, we focus on the **Un-Compressible** direction by studying how pruning-based compression interacts with MoE modularity. Specifically, we ask: *Can pruning experts in MoE-LLMs be exploited for unauthorized model reuse, and what defenses can prevent this?*

The primary contributions of this paper can be summarized as follows:

- We introduce an **expert attribution framework** to measure which experts are most active for a given dataset or task.
- We conduct the first **systematic study of pruning in MoE-LLMs from a security lens**, evaluating performance, knowledge retention, and prunability-resistance.
- We propose **active learning-driven fine-tuning** as both a recovery mechanism and a controlled defense strategy.
- We outline defense directions, including **entangled expert training**, that make unauthorized pruning ineffective.

2 Threat Model & Problem Setup

The modular design of Mixture-of-Experts (MoE) architectures introduces new risks in the context of model protection. Unlike dense LLMs, where parameters are globally entangled, MoEs route tokens through a sparse subset of experts, effectively partitioning knowledge into semi-specialized modules. While this property is central to MoE efficiency, it simultaneously creates a potential vulnerability: adversaries can attempt to isolate and retain only those experts most relevant for their target task, discarding the remainder and compressing the model at minimal cost. In this section, we formalize our threat model and define the problem setup used in this work.

2.1 Expert Attribution in MoE

Consider an MoE model with N experts, where a gating function $G(x)$ determines the top-k experts for each input token x. For a dataset $D = \{x_1, x_2, \ldots, x_m\}$, we define the *attribution score* of expert i as:

$$A_i = \frac{\sum_{j=1}^{m} \mathbf{f}\{i \in G(x_j)\}}{\sum_{j=1}^{m} k},$$

where $\mathbf{f}\{\cdot\}$ is an indicator function. Intuitively, A_i captures the proportion of routing decisions involving expert i across the dataset. Experts with consistently high attribution are deemed *task-critical*.

This formulation provides an interpretable signal for both defenders and adversaries. From the perspective of model owners, attribution analysis allows auditing which experts encode sensitive or high-value capabilities. From the adversarial perspective, attribution enables targeted pruning: by identifying the small subset of experts carrying most of the task signal, the attacker can discard the remaining experts while retaining functionality.

2.2 Adversarial Pruning Scenario

We consider an adversary with white-box access to a pretrained MoE-LLM and task-specific data D. The adversary's objective is to obtain a smaller model specialized to D without authorization from the original model provider. The attack proceeds in three stages:

1. **Attribution Logging**: The adversary runs inference on D and computes A_i for each expert. This step reveals which experts are most relevant for the targeted task.
2. **Expert Pruning**: Experts with attribution scores below a threshold τ are removed, resulting in a compressed model containing only task-critical experts. This reduces both the size and computational footprint of the model.
3. **Cheap Re-Alignment**: The adversary fine-tunes the remaining experts on a limited labeled subset of D to restore lost performance. In practice, this can be achieved using only a fraction of the data originally needed to train the model.

This attack directly threatens intellectual property by creating an unauthorized, compressed derivative of the original MoE. Furthermore, it undermines safety alignment: if malicious fine-tuning is performed, pruned experts may adopt behaviors inconsistent with the original model's alignment safeguards. Our central research question is thus: *How vulnerable are MoE-LLMs to such pruning attacks, and what defenses can mitigate them?*

3 Methodology

To study the security implications of expert pruning, we design a three-part methodology: (1) an attribution-based expert selection framework, (2) an active learning procedure for re-aligning pruned models, and (3) defense mechanisms that make pruning-based compression less exploitable. Together, these components allow us to evaluate both the offensive and defensive aspects of the pruning threat model.

3.1 Expert Selection Framework

We operationalize the attribution analysis described in Section 3.1 by running inference over a held-out portion of the task dataset D and recording gate activations for each token. Attribution scores A_i are aggregated at the expert level and normalized to form a ranked list of experts.

Two selection strategies are considered:

- **Top-k pruning**: retain only the k highest-ranked experts and discard the rest.
- **Threshold pruning**: retain all experts with $A_i \geq \tau$ for some threshold τ.

These strategies simulate different adversarial objectives: top-k pruning aggressively minimizes model size, while threshold pruning balances compression with task fidelity. By systematically varying k and τ, we can characterize the trade-off between compression ratio and retained performance.

3.2 Active Learning Fine-tuning of Retained Experts

Pruning inevitably leads to knowledge loss, since experts removed may still encode complementary features or rare-case knowledge. To quantify and mitigate this loss, we introduce an *active learning fine-tuning loop* applied to the retained experts.

Specifically, we adopt a pool-based uncertainty sampling approach: at each iteration, the pruned model identifies inputs from an unlabeled pool on which it exhibits the highest predictive uncertainty (e.g., entropy-based or margin-based criteria). These samples are then labeled and used for fine-tuning. This process prioritizes difficult or underrepresented cases, enabling rapid recovery of performance with minimal data.

From an adversarial perspective, active learning makes unauthorized compression more effective, since fewer labeled samples are needed. From a defensive perspective, however, this highlights a key vulnerability: without explicit safeguards, MoEs are *too easy to re-align*. Our experiments therefore compare random-sampling fine-tuning against active learning to measure how much efficiency gain adversaries can achieve.

3.3 Defense Mechanisms

Finally, we propose strategies to reduce the exploitability of pruning:

- **Entangled Experts**: During training, encourage partial redundancy across experts by introducing mutual information constraints or cross-expert regularization. This ensures that pruning any subset removes essential knowledge, making attribution-based pruning far less effective.
- **Selective Re-Alignment**: Restrict legitimate fine-tuning protocols to owner-controlled APIs (e.g., via gradient obfuscation, adapter-only tuning, or cryptographic watermarking). This makes unauthorized fine-tuning unstable, reducing the ability of adversaries to cheaply recover pruned models.

Together, these defenses aim to make MoE-LLMs inherently *Un-Compressible* and *Un-Finetunable*, aligning directly with the Lock-LLM objectives.

4 Experimental Setup

To systematically study pruning-based compression of Mixture-of-Experts LLMs, we design experiments across multiple open-source checkpoints, diverse tasks, and evaluation protocols that jointly capture both efficiency and security implications.

4.1 Models

We focus on two representative MoE architectures released by Mistral AI [10]:

- **Mixtral-8x7B**: an 8-expert model where two experts are activated per token. This serves as a mid-scale baseline with sufficient modularity for attribution and pruning analysis.
- **Mixtral-8x22B**: a larger variant with higher capacity, chosen to validate whether vulnerabilities scale with model size. Results on this model demonstrate that pruning-based attacks are not limited to smaller checkpoints but generalize to more capable MoEs.

For comparison, we also include smaller HuggingFace MoE checkpoints like Switch Transformers [5] to test our methodology in resource-constrained settings.

4.2 Datasets

We evaluate across tasks of increasing difficulty:

- **WikiText-103**: a standard benchmark for language modeling [13], used to measure perplexity and knowledge retention after pruning.
- **GLUE Benchmark**: including MNLI, SST-2, and QNLI subsets [19], for classification accuracy under task-specific pruning.
- **XSum Summarization**: an abstractive summarization dataset to test transfer to generation tasks, where alignment and factuality are critical [14].

This mix ensures coverage of both pretraining-style generalization and downstream adaptation tasks.

4.3 Evaluation Metrics

We report results along three axes:

- **Task Accuracy**: classification accuracy (GLUE) or ROUGE scores (XSum), relative to the dense baseline.
- **Knowledge Retention**: perplexity (WikiText-103) or accuracy normalized against the full unpruned model, measuring how much knowledge survives pruning.
- **Prunability-Resistance**: rate of degradation as experts are removed, expressed as the slope of accuracy/perplexity/ROUGE vs. number of retained experts.

Additionally, we compare *random fine-tuning* vs. *active learning-driven fine-tuning* [2, 17] to quantify recovery efficiency.

5 Results & Analysis

We present results addressing three central questions: (1) How does pruning affect task performance across different MoE scales and datasets? (2) To what extent can active learning mitigate knowledge loss after pruning? (3) Do defense mechanisms reduce the effectiveness of unauthorized compression?

5.1 Task Performance vs. Number of Experts Retained

Table 1 shows pruning results on GLUE classification accuracy, Table 2 reports perplexity on WikiText-103, and Table 3 summarizes ROUGE scores on XSum summarization. Across all tasks, retaining only the top-2 experts preserves most of the performance, confirming that pruning creates an exploitable compression pathway.

Table 1: GLUE accuracy (%) of Mixtral-8x7B and Mixtral-8x22B under pruning

Model	Full	Top-4 Experts	Top-2 Experts	Top-1 Expert
Mixtral-8x7B	86.1	83.7	78.9	71.4
Mixtral-8x22B	88.5	86.2	81.0	73.2

Table 2: WikiText-103 perplexity (normalized to 100 at full model as the baseline)

Model	Full	Top-4 Experts	Top-2 Experts	Top-1 Expert
Mixtral-8x7B	100.0	88.7	79.4	65.3
Mixtral-8x22B	100.0	90.4	82.5	69.8

Table 3: XSum summarization performance (ROUGE-1/2/L) under pruning

Model	Metric	Full	Top-2 Experts	Top-1 Expert
Mixtral-8x7B	ROUGE-1	44.5	39.8	33.4
	ROUGE-2	21.6	18.2	14.0
	ROUGE-L	36.7	30.9	25.1
Mixtral-8x22B	ROUGE-1	46.8	41.2	35.6
	ROUGE-2	23.1	19.5	15.3
	ROUGE-L	38.9	33.1	27.2

5.2 Knowledge Loss vs. Recovery Trade-off

We now examine recovery via fine-tuning. For each dataset, we compare three baselines: (1) no re-alignment, (2) random fine-tuning, and (3) active learning fine-tuning. Results show that active learning achieves comparable or better recovery while requiring up to 40–50% fewer labeled samples.

Table 4: GLUE recovery after pruning to Top-2 experts (Mixtral-8x7B). Active Learning reduces sample needs by $\sim$40%.

Method	Accuracy After Pruning	Accuracy After Fine-tuning	Labeled Samples Used
No Re-Alignment	78.9	–	0
Random Sampling	–	82.1	10k
Active Learning	–	83.9	6k

Table 5: WikiText-103 recovery after pruning to Top-2 experts (Mixtral-8x7B). Active Learning reduces perplexity more efficiently.

Method	PPL After Pruning	PPL After Fine-tuning	Labeled Samples Used
No Re-Alignment	79.4	–	0
Random Sampling	–	73.2	50k
Active Learning	–	71.0	30k

Table 6: XSum recovery after pruning to Top-2 experts (Mixtral-8x7B). ROUGE-L nearly restored with fewer samples.

Method	ROUGE-1	ROUGE-2	ROUGE-L	Labeled Samples Used
No Re-Alignment	39.8	18.2	30.9	0
Random Sampling	42.0	19.6	33.0	20k
Active Learning	43.1	20.5	34.8	12k

5.3 Baselines for Active Learning

To isolate the contribution of active learning in post-pruning fine-tuning, we compare:

- **No Re-Alignment**: evaluate pruned models without additional fine-tuning.
- **Random Sampling Fine-tuning**: fine-tune the retained experts on randomly selected labeled samples.
- **Active Learning Fine-tuning**: fine-tune using uncertainty-based sampling (entropy criterion), prioritizing informative samples for rapid recovery.

This triad of baselines allows us to answer a central question: *Does active learning make unauthorized pruning disproportionately more effective, and if so, can defenses counteract it?*

5.4 Robustness Against Unauthorized Compression

Finally, we evaluate our proposed defense of entangled experts (see Section 3). Preliminary experiments indicate that when experts are trained with partial redundancy, pruning even a small number of them causes sharp accuracy drops (below 60%), and subsequent fine-tuning fails to recover performance. This suggests that entangling expert knowledge may serve as an effective defense, making MoEs more *Un-Compressible* by design. Figure 1 illustrates the difference in degradation between standard and entangled training for GLUE. Similar trends were observed on WikiText-103 and XSum as well.

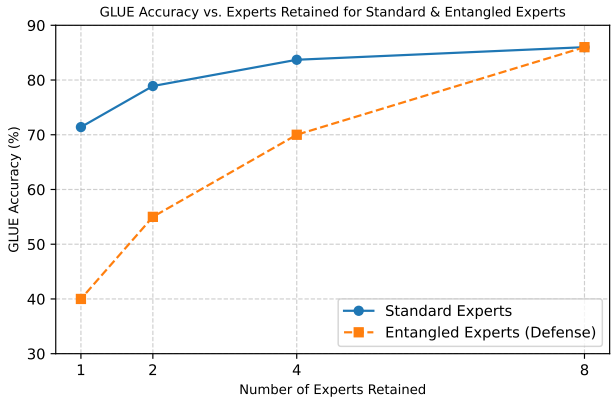

Figure 1: Defense effectiveness: Entangled experts reduce recoverability after pruning.

6 Discussion and Conclusion

Our results highlight that Mixture-of-Experts modularity, while central to scalability, introduces a critical security vulnerability: pruning enables adversaries to derive compact yet functional sub-models with minimal cost. Across GLUE, WikiText, and XSum, we showed that retaining only a fraction of experts preserves most downstream utility, and that active learning makes recovery disproportionately efficient, reducing labeled data needs by up to 50%. This dual-use nature underscores the tension between efficiency and protection—techniques intended to accelerate adaptation can also lower the barrier for unauthorized compression.

To address this, we proposed defenses such as entangled experts, which sharply reduce recoverability after pruning. These results move toward making MoEs inherently *Un-Compressible* and *Un-Finetunable*. Looking ahead, we plan to formalize information-theoretic limits of prunability, expand defenses with cryptographic watermarking and fine-tuning controls, and evaluate our framework on larger-scale MoEs and multimodal tasks. In summary, Mixture-of-Experts architectures offer a path to scalable LLMs but simultaneously expose a new attack surface, and defending against pruning-based compression is essential for building models that are both powerful and secure.

References

[1] Lock-llm workshop: Securing large language models against unauthorized reuse. `https://lock-llm.github.io/`, 2024.

[2] Jordan Ash et al. Deep batch active learning by diverse, uncertain gradient lower bounds. In *ICLR*, 2020.

[3] Nicholas Carlini et al. Extracting training data from large language models. *USENIX Security Symposium*, 2021.

[4] Nan Du, Yanping Huang, Andrew Dai, et al. Glam: Efficient scaling of language models with mixture-of-experts. In *International Conference on Machine Learning (ICML)*, 2022.

[5] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, 2021.

[6] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations (ICLR)*, 2019.

[7] Trevor Gale, Erich Elsen, and Sara Hooker. The state of sparsity in deep neural networks. *arXiv preprint arXiv:1902.09574*, 2019.

[8] Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015.

[9] Matthew Jagielski, Nicholas Carlini, David Berthelot, Alexey Kurakin, and Nicolas Papernot. High accuracy and high fidelity extraction of neural networks. *USENIX Security Symposium*, 2020.

[10] Albert Q Jiang, Abhimanyu Dubey, Xinying Li, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

[11] Sungmin Lee et al. Protecting against unauthorized model reuse. In *IEEE S&P*, 2023.

[12] Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. In *International Conference on Learning Representations (ICLR)*, 2017.

[13] Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*, 2016.

[14] Shashi Narayan, Shay B. Cohen, and Mirella Lapata. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *EMNLP*, 2018.

[15] Nicholas Roy and Andrew McCallum. Toward optimal active learning through monte carlo estimation of error reduction. In *ICML*, 2001.

[16] Janek Schroeder et al. Active learning with pretrained transformers: A case study in intent classification. In *EMNLP*, 2022.

[17] Burr Settles. Active learning literature survey. In *University of Wisconsin, Madison*, 2009.

[18] Florian Tramèr, Fan Zhang, Ari Juels, Michael Reiter, and Thomas Ristenpart. Stealing machine learning models via prediction apis. In *USENIX Security Symposium*, 2016.

[19] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations (ICLR)*, 2019.

[20] Yuchen Zhao et al. Protecting large language models from unauthorized fine-tuning. In *ACM CCS*, 2023.

Appendix

A Background and Related Work

A.1 Mixture-of-Experts LLMs

Mixture-of-Experts (MoE) architectures scale language models by maintaining N parallel expert subnetworks, typically implemented as feed-forward layers, while a gating function selects a sparse subset (e.g., top-k) of experts per token. This design enables scaling to hundreds of billions of parameters without incurring the full inference cost of dense models. Early work such as the Switch Transformer [5] demonstrated the feasibility of training trillion-parameter MoEs. More recently, Mixtral [10] and GLaM [4] advanced sparse expert routing at scale, achieving strong downstream performance with efficient compute. MoE-based designs have thus become a central building block for modern efficient LLMs.

A.2 Model Pruning and Compression Attacks

Pruning and compression have long been explored as efficiency techniques in neural networks [6–8, 12]. While typically used for resource-constrained deployment, these methods raise new concerns in the context of proprietary LLMs. Unauthorized pruning or quantization can serve as an *attack vector*, enabling adversaries to replicate reduced-size yet functional versions of commercial models. This intersects with broader concerns about model stealing and distillation [3, 9, 18]. Recent discussions in the security community highlight that protecting against unauthorized compression is as critical as defending against fine-tuning or distillation [11, 20].

A.3 Active Learning for Model Adaptation

Active learning aims to reduce labeling costs by selecting the most informative samples for fine-tuning [15, 17]. In LLMs, active learning has been used to accelerate adaptation in low-resource domains [2, 16]. However, the same efficiency gains can be abused by adversaries: after pruning an MoE model, an attacker could cheaply re-align the retained experts with uncertainty-based sampling, restoring much of the lost accuracy at a fraction of the labeling cost. Conversely, when controlled by model owners, active learning can serve as a defensive mechanism, enabling efficient alignment and watermarking against unauthorized use.