ROUTEMIA: EXPERT ROUTING LEAKS MEMBERSHIP IN MOE LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Many membership inference attacks on large language models rely on log-probabilities of the input tokens that deployed services do not expose. Mixture-of-Experts (MoE) models, however, leak a different MIA signal: how input tokens are distributed across experts, which may be exposed through side channels even when log-probabilities are hidden. Specifically, we find that fine-tuning shifts this distribution more for members than for non-members. Building on this observation, we propose ROUTEMIA, which infers membership by measuring the Jaccard distance between expert sets selected by a public base model and by its fine-tuned counterpart from the same input. Unlike prior attacks, ROUTEMIA uses only the number of tokens routed to each expert, a signal that side channels may reveal, rather than log-probabilities. Because some routing changes occur regardless of membership, ROUTEMIA calibrates these changes against token identity and base routing margin using non-member reference data. Experiments across three MoE LLMs and four datasets demonstrate routing-based membership leakage and show that the proposed calibration generally improves attack performance. Our findings identify expert routing as a new privacy attack surface introduced by sparse MoE architectures.

1 INTRODUCTION

Membership inference attacks (MIAs) determine whether a sample was used to train a model and are a standard tool for studying training-data privacy (Shokri et al., 2017). For large language models (LLMs), many effective MIAs exploit the fact that training samples exhibit loss or different log-probabilities from unseen samples (Carlini et al., 2021; 2022; Shi et al., 2024). However, these signals are often unavailable in realistic deployments, as most LLM inference engines (e.g., vLLM, SGLang) only expose output text by default (Kwon et al., 2023; Zheng et al., 2024). This creates an important practical gap between existing MIA and deployed LLM services: *without observing the probabilities a model assigns to the input, can we still infer whether the model was trained on it?*

We investigate this question for Mixture-of-Experts (MoE) models, which scale model capacity through sparse expert activation (Shazeer et al., 2017) and have been adopted by many recent open-weight LLMs (Yang et al., 2025). These public models are commonly fine-tuned on private data and self-hosted with open-source inference engines, which do not return log-probabilities typically. We observe that, relative to the base model, fine-tuning changes the experts selected for the member inputs more than for non-member inputs. This is because fine-tuning shifts hidden representations more for members, which reorders experts near the top-k cutoff and alters expert selection. This constitutes a form of *structure-level membership leakage*: **membership is reflected not only in output-based signals such as loss, but also in the model's routing behaviors.**

Based on this observation, we propose ROUTEMIA, a gray-box membership inference attack that exploits changes in expert routing in fine-tuned MoE models without accessing log-probabilities. Given a public base model and its fine-tuned counterpart, ROUTEMIA measures the *Jaccard Distance (JD)* between their top-k expert sets and uses this routing difference as the basic membership signal. Raw JD, however, also varies for two reasons unrelated to the membership. First, some token types naturally exhibit larger routing changes after fine-tuning regardless of membership, meaning that non-member samples containing these tokens can still receive high JD. Second, routing changes also depend on the base-model routing margin, with smaller margins making top-k expert changes

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

more likely under the same representation shift. To reduce the membership-dependent component, ROUTEMIA uses held-out reference data to remove the expected routing variation associated with token identity and routing margin, yielding a signal that more direclty reflects membership.

Beyond the public base models, ROUTEMIA requires only the identities of the experts selected by the target model, which prior work has shown can be recovered through execution side channels in MoE inference (Ding et al., 2025). This makes ROUTEMIA applicable in principle to deployments that return only output text. ROUTEMIA achieves an average AUROC of 0.606 and and an average TPR of 2.0% at 1% FPR across three MoE models and four datasets after three epochs of fine-tuning, matching or outperforming the strongest text-only attack in most settings, and reaches 0.706 with longer fine-tuning. Overall, our contributions are summarized as follows:

- **Routing-based membership leakage.** We find that fine-tuning members tend to exhibit larger changes in selected expert sets than non-members, revealing membership leakage through model-internal computation paths beyond conventional probabilities-based signals.
- **Attack and calibration.** We propose ROUTEMIA, which exploits routing-based side-channel information by measuring the Jaccard distance between expert sets selected by a public base model and its fine-tuned counterpart. It calibrates these changes against token identity and base routing margin using non-member reference data to reduce membership-independent routing variation.
- **Empirical validation.** Experiments across three MoE models, four text datasets, and three fine-tuning lengths demonstrate routing-based membership leakage and show that calibration generally improves attack performance. Comparisons with four output-based black-box attacks further demonstrate the effectiveness of ROUTEMIA, achieving an average AUROC of 0.606.

2 RELATED WORKS

2.1 MIXTURE-OF-EXPERTS

Mixture-of-Experts (MoE) architectures scale model capacity through sparse conditional computation, where a router selects a small subset of experts for each token (Shazeer et al., 2017). Representative architectures include GShard (Lepikhin et al., 2021), Switch Transformer (Fedus et al., 2022), and recent MoE LLMs such as Qwen3 (Yang et al., 2025) and DeepSeek-V4 (DeepSeek-AI, 2026). Existing work has shown that the MoE architecture can introduce new security and privacy risks. For example, GBFA exploits MoE-specific components to induce persistent generation failures (Lin et al., 2026). MoEcho further shows that expert execution can be recovered through hardware side channels during MoE inference (Ding et al., 2025). Our work studies a different privacy question: whether the routing behavior introduced by the MoE architecture can also leak membership information.

2.2 MEMBERSHIP INFERENCE ATTACKS ON LLMS

Membership inference attacks (MIAs) determine whether a candidate sample was used to train a target model. Early MIAs were primarily developed for classification models using prediction scores or internal model information (Shokri et al., 2017; Nasr et al., 2019). For generative LLMs, recent attacks exploit several different types of signals.

Likelihood-based attacks. Many LLM MIAs rely on token probabilities, loss, or related likelihood statistics. Min-K% Prob (Shi et al., 2024) detects pre-training membership from low-probability tokens, while SPV-MIA (Fu et al., 2024) and WBC (Chen et al., 2026a) study fine-tuning membership using probability- or loss-derived signals.

Response-based attacks. More restrictive attacks use only generated text or model responses. PETAL estimates token confidence from semantic similarity (He et al., 2025), SaMIA constructs a pseudo-likelihood score from sampled text (Kaneko et al., 2025), and Pop Quiz infers membership from multiple-choice responses (Chen et al., 2026b).

Internal-state attacks. Other LLM MIAs exploit internal model representations. LUMIA uses hidden activations for membership inference (Ibanez-Lissen et al., 2025), while memTrace extracts membership signals from hidden states and attention patterns (Makhija et al., 2026).

Related but distinct work studies broader training-data detection. For example, DE-COP detects copyrighted training content (Duarte et al., 2024), while CDD studies data contamination from sampled model outputs (Dong et al., 2024).

In contrast, ROUTEMIA identifies the expert routing signal specific to MoE models. It infers membership from changes in selected expert sets during model execution, without requiring target token likelihoods, generated responses, or direct access to hidden representations.

3 THREAT MODEL

Attack Scenarios. We consider an enterprise deployment scenario in which an organization fine-tunes a publicly released open-weight MoE LLM on a private dataset and deploys the resulting model on its own inference infrastructure with self-host service engines. The adversary is an unprivileged user with legitimate query access to the deployed model and can run a co-located process on the same physical machine as the victim inference service. The adversary exploits micro-architectual side channels arising from dynamic expert execution to determine whether a candidate sample was a member in the private fine-tuning dataset.

We consider LoRA fine-tuning, where only the low-rank adapter parameters are updated and the pretrained weights remain frozen. In our main experiments, LoRA is applied to the expert FFNs, while the router remains fixed. Router adaptation is examined in Appendix D.4 and discussed in Section 6.2. Full-parameter fine-tuning is outside the scope of this study due to its computational cost.

Attacker Capabilities. Since the target model is fine-tuned from a publicly available open-weight checkpoint, the attacker can obtain and locally execute the corresponding base model. For any candidate input, the attacker can directly observe the base model's expert selections. In contrast, the attacker has no access to the parameters or internal states of the fine-tuned target model, including hidden states, router logits, token probabilities, loss values, or gradients. The attacker can repeatedly query the target model with chosen inputs and collect side-channel traces from the co-located environment, from which the executed experts can be inferred. The attacker does not modify the victim model and has no access to the private fine-tuning dataset. For calibration, we assume the attacker has access to a close-domain reference dataset which has not been used for fine-tuning and can query it through the same interface.

Attack Objectives. Let M_0 denote the publicly available base MoE model and M_{ft} the target model fine-tuned on a private dataset D_{train}. Given a candidate sample x, the adversary aims to determine whether $x \in D_{\mathrm{train}}$ by comparing the routing behavior of M_0 with which from the M_{ft}.

4 ROUTEMIA

4.1 OVERVIEW

ROUTEMIA exploits membership leakage through changes in MoE expert routing. We first describe how hidden-state drift between a public base model and its fine-tuned counterpart carries membership information and changes expert selections. We then measure these routing changes using the Jaccard Distance and establish its connection to hidden-state drift through analysis and empirical validation. The attack requires only the target model's selected expert identities, without access to its loss or router logits. Finally, we identify membership-independent routing variation associated with token identity and base routing margin, and calibrate JD against these factors using non-member reference data.

4.2 HIDDEN-STATE DRIFT AND ROUTING CHANGE

At MoE layer ℓ and token position t, let $\mathbf{h}^0_{\ell,t}$ and $\mathbf{h}^{\mathrm{ft}}_{\ell,t}$ be the normalized hidden states supplied to their routers for the same input. We define hidden-state drift as

$$\Delta\mathbf{h}_{\ell,t} = \mathbf{h}^{\mathrm{ft}}_{\ell,t} - \mathbf{h}^0_{\ell,t}. \tag{1}$$

Members tend to exhibit larger representation changes than non-members, indicating that hidden-state drift carries membership information. A white-box score based directly on hidden-state drift reaches up to 0.755 AUROC, with full results reported in Appendix A.

In our setting, fine-tuning adapts the expert networks while keeping the router fixed. Changes in the representations supplied to the router can nevertheless change expert scores and move tokens across top-k routing boundaries. The resulting expert replacements provide an observable signal of the underlying drift and its membership information.

4.3 Modeling Membership Signals as Jaccard Distance

Let $S^0_{\ell,t}$ and $S^{\text{ft}}_{\ell,t}$ denote the top-k expert sets selected by the two models. We quantify routing change by averaging their Jaccard distance across tokens and scoring layers:

$$J(x) = \frac{1}{|\mathcal{L}|T} \sum_{\ell \in \mathcal{L}} \sum_{t=1}^{T} \left(1 - \frac{\left| S^0_{\ell,t} \cap S^{\text{ft}}_{\ell,t} \right|}{\left| S^0_{\ell,t} \cup S^{\text{ft}}_{\ell,t} \right|} \right). \tag{2}$$

where $\mathcal{L}$ denotes the set of MoE layers used for scoring. We write $J_{\ell,t}$ for the individual Jaccard distance inside the parentheses. It is zero when the two models select the same expert set for token t at layer ℓ, and increases as more experts are replaced. The sample-level score $J(x)$ summarizes these changes, with larger values indicating greater routing change after fine-tuning.

Connection to hidden-state drift. Our goal is to show how hidden-state drift can induce changes in JD. When the drift crosses a routing boundary, it changes the selected expert set and therefore produces a nonzero JD. Along a fixed drift direction, JD is also non-decreasing with the drift magnitude.

Fix a token and a layer, omit their indices, and write the drift as $\Delta\mathbf{h} = \sigma\mathbf{u}$, where $\|\mathbf{u}\|_2 = 1$ and $\sigma \geq 0$. Let S and S' denote the base and fine-tuned top-k expert sets, and let $j = |S \setminus S'|$ be the number of displaced base experts. Since both sets contain k experts, $|S \cap S'| = k - j$ and $|S \cup S'| = k + j$, so

$$J = \frac{2j}{k + j}. \tag{3}$$

This expression is strictly increasing in j, so more expert replacements lead to a larger JD.

With a fixed router, hidden-state drift changes the routing logits as

$$\mathbf{g}(\sigma) = \mathbf{g}^0 + \mathbf{W}_g \Delta\mathbf{h} = \mathbf{g}^0 + \sigma\mathbf{v}, \qquad \mathbf{v} = \mathbf{W}_g\mathbf{u}. \tag{4}$$

Let $m = g^0_{(k)} - g^0_{(k+1)}$ be the base routing margin and $c_g = \max_i \|\mathbf{w}_{g,i}\|_2$. Each routing logit changes by at most σc_g. Therefore, if $\sigma < m/(2c_g)$, every base-selected expert still outranks every base-unselected expert, so the top-k set remains unchanged and $J = 0$. Thus, sufficiently small hidden-state drift is not visible in JD.

Now consider a base-selected expert a and a base-unselected expert b. Their logit difference $g_b(\sigma) - g_a(\sigma)$ is linear in σ and is negative at $\sigma = 0$. If it crosses zero, b overtakes a, changing the top-k expert set. By Eq. (3), such replacements increase JD. Moreover, along the fixed direction $\mathbf{u}$, each selected–unselected pair can reverse its ordering at most once. Hence the number of displaced base experts, and therefore JD, is non-decreasing as σ increases.

These results show that JD captures hidden-state drift that changes expert routing: crossing routing boundaries changes the selected expert set and increases JD. Proposition 1 in Appendix E gives the full proof.

Empirical validation. We test whether this connection also appears across inputs by measuring the Spearman correlation between block-level Hidden-State Drift (HSD), the mean $\|\Delta\mathbf{h}_{\ell,t}\|_2$ over tokens and scoring layers, and JD on WikiText-103. Both signals use the last five MoE layers and are evaluated on $1{,}000$ members and $1{,}000$ non-members. Table 1 shows positive correlations across all three models and three fine-tuning lengths ($\rho = 0.385$–0.884 for raw JD). HSD is used only for this white-box diagnostic; the attack uses expert sets. The calibrated variants in the table are defined in Section 4.5. This evidence supports JD as an observable indicator of representation drift.

Table 1: Spearman rank correlation ρ between HSD and routing-based scores on WikiText-103. All signals are averaged over the last five MoE layers and evaluated on the same 2,000 blocks.

	OLMoE (E)			Qwen (E)			DeepSeek (E)		
	3	4	5	3	4	5	3	4	5
$\rho(\mathrm{HSD}, \mathrm{JD})$	0.884	0.854	0.838	0.708	0.595	0.492	0.636	0.521	0.385
$\rho(\mathrm{HSD}, \mathrm{JD}_{\mathrm{id}})$	0.801	0.800	0.807	0.582	0.499	0.422	0.440	0.357	0.256
$\rho(\mathrm{HSD}, \mathrm{JD}_{\mathrm{mg}})$	0.953	0.942	0.933	0.938	0.912	0.874	0.859	0.788	0.677
$\rho(\mathrm{HSD}, \mathrm{JD}_{\mathrm{cal}})$	0.852	0.864	0.875	0.823	0.789	0.743	0.646	0.578	0.470

The target expert identities can be obtained through existing side-channel methods (Ding et al., 2025), without access to the target model's loss, output probabilities, hidden states, or router logits. We recover per-token expert identities using sequential prefix queries with prefix caching; details are provided in Appendix C.

4.4 Membership-Independent Routing Variation

However, a large JD does not necessarily arise from training membership alone. Token identity and routing sensitivity can also introduce substantial membership-independent variation. We examine these effects on WikiText-103 using 1,000 members and 1,000 non-members, with JD and base routing margins averaged over the last five MoE layers. Detailed visualizations are provided in Appendix B.

Token identity. For each token type occurring at least 20 times in each class, we compare its mean JD between members and non-members. Across 2,428 token types, the two means are strongly correlated (Pearson $r = 0.79$; frequency-weighted $r = 0.97$), indicating that some token types consistently exhibit large routing changes regardless of membership. Members show an additional average shift of 0.011. This shared token-dependent variation motivates removing a token-specific baseline before computing the membership score.

Base routing margin. Routing changes also depend strongly on the base top-k margin. After grouping tokens into 20 margin quantiles, non-members in the smallest-margin bin exhibit $3.8\times$ the mean JD of those in the largest-margin bin. Members remain higher across all bins, with gaps ranging from 0.006 to 0.013. Similar trends appear in Qwen and DeepSeek, whose token-type correlations at epoch five are 0.89 and 0.87, with corresponding margin-bin ratios of 4.3 and 2.4. These observations motivate a second calibration step that removes the margin-dependent routing baseline.

4.5 Calibrated Routing Score

The preceding observations motivate two sequential corrections: removing the routing variation associated with token identity and then the remaining variation associated with the base routing margin. Both corrections are estimated from held-out non-member reference data.

Reference set. We assume access to an auxiliary reference set $\mathcal{R}$ from the same domain that is known not to have been used for fine-tuning. For each reference sample, we collect the selected expert sets from both models and the routing margins from the public base model, using the same procedure as for candidate samples.

For token t, we first average JD and the base top-k margin over the scoring layers:

$$\bar{J}_t(x) = \frac{1}{|\mathcal{L}|} \sum_{\ell \in \mathcal{L}} J_{\ell,t}(x), \qquad \bar{m}_t(x) = \frac{1}{|\mathcal{L}|} \sum_{\ell \in \mathcal{L}} \left(g^0_{\ell,t,(k)} - g^0_{\ell,t,(k+1)} \right). \tag{5}$$

In our experiments, $\mathcal{L}$ contains the last five MoE layers.

Token calibration. For token identity v, we estimate its expected routing change on the reference set as

$$\widehat{\mu}_{\mathrm{id}}(v) = \frac{\sum_{t\in\mathcal{R}:x_t=v} \bar{J}_t + \lambda \bar{J}_{\mathcal{R}}}{n_{\mathcal{R}}(v) + \lambda}, \tag{6}$$

where $n_{\mathcal{R}}(v)$ is the number of reference occurrences of v, $\bar{J}_{\mathcal{R}}$ is the global reference mean, and $\lambda = 5$. The token-calibrated residual is

$$r_t(x) = \bar{J}_t(x) - \widehat{\mu}_{\mathrm{id}}(x_t). \tag{7}$$

Unseen token identities fall back to the global reference mean.

Margin calibration. We partition $\bar{m}_t$ into 20 quantile bins using the reference set. Let $b(\bar{m}_t)$ denote the corresponding bin. We estimate the remaining margin-dependent baseline from the token-calibrated residuals:

$$\widehat{\mu}_{\mathrm{mg}}(b) = \frac{\sum_{t\in\mathcal{R}:b(\bar{m}_t)=b} r_t + \lambda \bar{r}_{\mathcal{R}}}{n_{\mathcal{R}}(b) + \lambda}, \tag{8}$$

and define

$$r_t^{\mathrm{cal}}(x) = r_t(x) - \widehat{\mu}_{\mathrm{mg}}\left(b(\bar{m}_t(x))\right). \tag{9}$$

Final score. ROUTEMIA averages the calibrated token residuals:

$$\mathrm{JD}_{\mathrm{cal}}(x) = \frac{1}{T}\sum_{t=1}^{T} r_t^{\mathrm{cal}}(x). \tag{10}$$

We use $\mathrm{JD}_{\mathrm{cal}}(x)$ as the final membership score, with larger values indicating more routing change than expected from the reference baselines. Additional estimation details are given in Appendix F.

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

Models. We study three sparse Mixture-of-Experts language models: **OLMoE-1B-7B** [OLMoE] (Muennighoff et al., 2025), **Qwen1.5-MoE-A2.7B** [Qwen] (Qwen Team, 2024) and **DeepSeek-V2-Lite** [DeepSeek] (DeepSeek-AI, 2024); tables use the short names in brackets. Their routing configurations are summarized in Table 4. OLMoE is the primary model.

Datasets. We fine-tune on four corpora, written in tables by the short names in brackets: **WikiText-103** [wikitext] (Merity et al., 2017), **CC-News** [ccnews] (Hamborg et al., 2017), **Reddit** [reddit] and **Amazon Reviews** [amazon] (Ni et al., 2019). WikiText-103 follows the packed-block protocol: the corpus is tokenized and concatenated into fixed *512-token* blocks. The other three keep one document per block, truncated to 512 tokens. From a document-level split we draw 10,000 member blocks (used for fine-tuning) and 10,000 non-member blocks from held-out documents that are disjoint from the members and filtered by 13-gram overlap; a further 1,000 held-out blocks form a validation set for perplexity monitoring.

Fine-tuning. We follow the fine-tuned-LLM membership setup of WBC (Chen et al., 2026a)—documents of at least 512 tokens, 10,000 members and disjoint held-out non-members, one block per sample—with two changes: we split documents contiguously and de-duplicate by 13-gram overlap instead of drawing a random split, and we adapt only the expert feed-forward networks with LoRA (rank 16, α=32, dropout 0), *freezing the router and the attention layers*, rather than fine-tuning all parameters. Optimization uses AdamW (learning rate 5×10^{-5}, weight decay 0.1), batch size 16, 500 warmup steps, bf16 with gradient checkpointing. To vary the memorization regime we train for $E \in \{3, 4, 5\}$ epochs, yielding three checkpoints of increasing over-fit. The member/non-member PPL gaps are given in Appendix D.2.

Baselines. We compare against four membership attacks that need only generated text or token-level scores from the target: **SaMIA** (Kaneko et al., 2025), **PETAL** (He et al., 2025), **DE-COP** (Duarte et al., 2024) and **CDD** (Dong et al., 2024). SaMIA and CDD are scored with the authors' code. Three attacks deviate from their published setting for cost reasons: SaMIA caps each sampled continuation

Table 2: Black-box attacks and our side-channel signal (AUROC) on three MoE models and four datasets at E=3, 1000 blocks per class. $\mathrm{JD}_{\mathrm{cal}}$ is the calibrated Jaccard distance of Section 4.5. Best in **bold**, second best underlined.

	OLMoE				Qwen				DeepSeek			
Method	wikitext	ccnews	reddit	amazon	wikitext	ccnews	reddit	amazon	wikitext	ccnews	reddit	amazon
SaMIA	.514	.526	.511	.511	.544	.548	.519	.506	.507	.527	.514	.502
PETAL	.622	.575	.564	.591	.623	.597	.587	.611	**.586**	.554	**.563**	**.583**
DE-COP	.509	.507	.509	.509	.511	.513	.514	.510	.511	.516	.502	.505
CDD	.501	.502	.502	.501	.501	.503	.501	.501	.501	.502	.504	.501
$\mathrm{JD}_{\mathrm{cal}}$ (ours)	**.642**	**.605**	**.599**	**.634**	**.658**	**.602**	**.622**	**.645**	**.586**	**.563**	.556	.557

at 128 new tokens instead of generating up to a 1024-token context, PETAL uses the first 64 words of a block, and DE-COP needs paraphrases of the target text, which we generate with Qwen3-30B-A3B (Yang et al., 2025) (3 paraphrases per block) since the original work uses a proprietary model. All are run at the same 1000 blocks per class as our signal.

Evaluation. Each signal is used directly as a membership score. We report AUROC and the true-positive rate at a low false-positive rate, TPR@1%FPR, on the first 1000 blocks of each class.

5.2 Comparison of RouteMIA with black-box MIAs

We first evaluate how competitive routing-based membership signals are against existing black-box methods for detecting training-data membership.

Setup. We include two black-box LLM MIAs, SaMIA and PETAL, and two related training-data detection methods, DE-COP and CDD. The comparison covers three MoE models and four datasets, with each model fine-tuned for three epochs on 10,000 member blocks. All methods are evaluated on the same 1,000 member and 1,000 non-member blocks under the configurations described in Section 5.1. RouteMIA uses the calibrated routing score $\mathrm{JD}_{\mathrm{cal}}$ and assumes access to expert selections through a side channel.

Results. Table 2 reports AUROC across the 12 model–dataset settings. Under the evaluated configurations, RouteMIA achieves the highest AUROC in nine settings, ties for the highest in one, and ranks second in the remaining two. It outperforms all four baselines on every dataset for OLMoE and Qwen, with gains over the strongest baseline of up to 0.043 AUROC on OLMoE–Amazon Reviews (0.634 versus 0.591). On DeepSeek, it leads on CC-News, matches PETAL on WikiText-103, and ranks behind PETAL on Reddit and Amazon Reviews. Across all settings, it consistently outperforms SaMIA, DE-COP, and CDD. Qwen and DeepSeek also contain shared experts that process every token (Qwen Team, 2024; DeepSeek-AI, 2024). Changes within these always-active experts do not directly produce expert-set flips, which may limit the membership information captured by routing changes. These results show that expert routing carries a useful membership signal and, under the assumed side-channel access, can outperform the evaluated output-based attacks without requiring target output probabilities or hidden states.

5.3 Signals Across Models and Datasets

We next study how routing-based membership signals change with fine-tuning epochs and how much calibration improves the raw JD signal.

Setup. We evaluate JD and $\mathrm{JD}_{\mathrm{cal}}$ across three MoE models, four text datasets, and $E \in \{3, 4, 5\}$ fine-tuning epochs, using 1,000 member and 1,000 non-member blocks per setting.

Results. Figure 1 reports the AUROC of $\mathrm{JD}_{\mathrm{cal}}$ across fine-tuning epochs and the corresponding calibration gain, $\Delta\mathrm{AUROC} = \mathrm{AUROC}(\mathrm{JD}_{\mathrm{cal}}) - \mathrm{AUROC}(\mathrm{JD})$. The calibrated score increases with fine-tuning epoch across all model–dataset settings. Calibration improves AUROC in 34 of 36 settings, with gains of up to 0.061.

Full AUROC and TPR@1%FPR results are reported in Table 9 in Appendix G. The table also includes the token-only $\mathrm{JD}_{\mathrm{id}}$ and margin-only $\mathrm{JD}_{\mathrm{mg}}$ variants; combining both corrections generally provides the strongest performance.

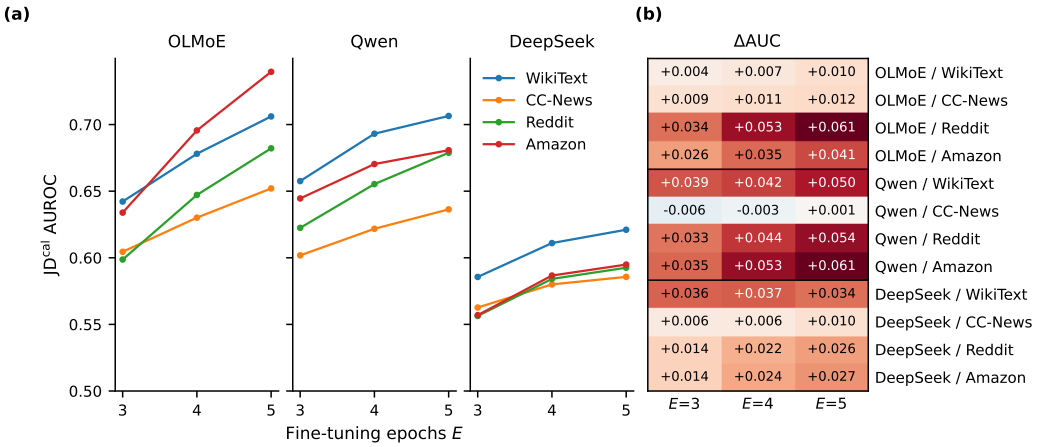


Figure 1: $\mathrm{JD}_{\mathrm{cal}}$ across fine-tuning epochs (1000 blocks per class). **(a)** AUROC of $\mathrm{JD}_{\mathrm{cal}}$ against the number of fine-tuning epochs E for OLMoE, Qwen and DeepSeek, one curve per dataset. **(b)** Calibration gain for every model, dataset and E, measured as the difference in AUROC, $\Delta\mathrm{AUC} = \mathrm{AUC}(\mathrm{JD}_{\mathrm{cal}}) - \mathrm{AUC}(\mathrm{JD})$, on the same blocks; 34 of 36 cells are positive, and the gain generally grows with E.

5.4 Layer-wise Analysis

We next examine which MoE layers expose the strongest membership signal.

Setup. For each MoE layer, we compute $\mathrm{JD}_{\mathrm{cal}}$ using only that layer's expert selections and report its AUROC on WikiText-103 across the three models and $E \in \{3, 4, 5\}$, using the same 1,000 blocks per class (Figure 2).

Results. The signal sits in the deep layers of every model. The first five MoE layers are close to chance at every E (mean AUROC 0.51–0.53 at E=5), and the AUROC rises through the network to peak in the last layer: 0.701 for OLMoE, 0.686 for Qwen and 0.654 for DeepSeek at E=5. Longer fine-tuning deepens the colours but does not move the pattern, and the last five layers, which the main results average, hold most of the signal (mean AUROC 0.673, 0.673 and 0.600 at E=5). DeepSeek shows the same shape at a lower level, with a dip at layer 23 (0.534) right before its strongest layers.

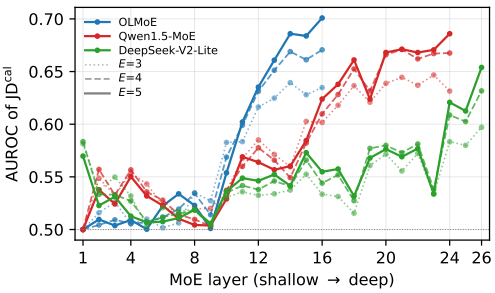


Figure 2: Per-layer membership-inference AUROC of $\mathrm{JD}^{\mathrm{cal}}$ on WikiText-103 across three MoE models and fine-tuning lengths. Colors denote models, and line styles denote fine-tuning epochs (E=3: dotted, E=4: dashed, E=5: solid). Full numerical results are reported in Table 10.

6 Discussion

6.1 Filtering Content-Driven Tokens

We next examine how token identities with intrinsically large routing changes affect the JD membership signal. Instead of calibrating these effects, we directly remove such tokens from scoring and measure the resulting change in AUROC. We also inspect which token identities contribute most strongly to this membership-independent routing variation.

Setup. We rank token identities by their mean JD on non-member reference data and exclude the highest-ranked identities from scoring, targeting $q\%$ of reference token occurrences. We evaluate raw JD and margin-calibrated $\mathrm{JD}_{\mathrm{mg}}$ on WikiText-103 at $E{=}5$, using 1,000 blocks per class. Random removal and removal of the lowest-ranked identities serve as controls (Table 6 in Appendix D.3).

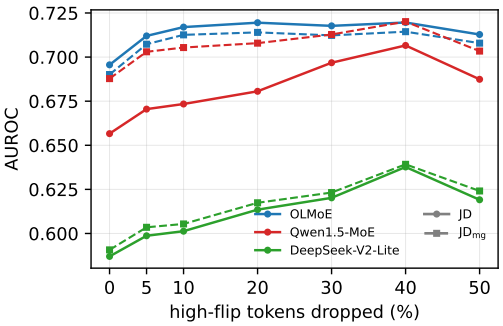


Figure 3: AUROC as reference token occurrences are removed on WikiText-103 ($E{=}5$). Tokens with the largest non-member mean JD are removed first. Solid: JD; dashed: $\mathrm{JD}_{\mathrm{mg}}$. See Appendix Tables 6 and 7 for full results and dropped tokens.

Results. Figure 3 shows that removing high-JD token identities generally improves membership AUROC, with the strongest performance around 40% removal and a decline at 50%. The highest-impact identities include section markers (=), punctuation, units (`km`, `cm`), and reporting words (`said`, `noted`; Table 7). In contrast, random removal provides little benefit, while removing low-JD identities weakens the signal (Table 6). These results indicate that a small subset of token identities contributes disproportionately to routing changes that are unrelated to membership. Filtering them therefore strengthens the JD membership signal even without token-identity calibration.

6.2 Possible Mitigation

Load-balancing regularization. Strong load-balancing regularization reduces routing-based membership leakage in our OLMoE–WikiText-103 experiment (Appendix D.4). With the router and experts jointly fine-tuned, $\alpha = 1.0$ lowers calibrated AUROC from 0.706 for the frozen-router reference to 0.629, with little change in perplexity. The pre-training coefficient $\alpha = 0.01$ has little effect, and even stronger regularization leaves measurable leakage.

Differentially private fine-tuning. Differentially private training limits the influence of individual training samples (Abadi et al., 2016) and has been applied to language-model fine-tuning (Li et al., 2022). In our setting, this would require protecting updates to the expert parameters, since freezing the router alone does not prevent membership leakage. Its effectiveness against routing-based membership inference and its effect on model utility remain to be evaluated.

7 Conclusion

We identified routing-based membership leakage in fine-tuned MoE models: members tend to exhibit larger changes in selected expert sets than non-members. Based on this observation, we proposed RouteMIA, which measures the Jaccard distance between expert sets selected by a public base model and its fine-tuned counterpart. The attack uses expert identities obtainable through side channels, without requiring target hidden states, router logits, or output probabilities. Calibration against token identity and base routing margin using non-member reference data reduces membership-independent routing variation. Experiments across three MoE models, four text datasets, and three fine-tuning lengths demonstrate this leakage and show that calibration generally improves attack performance. These findings highlight the need to account for membership leakage through model-internal computation paths beyond conventional output-based signals.

AI Use Statement

We used generative AI tools to polish the language of the manuscript.

REFERENCES

Martín Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 308–318. Association for Computing Machinery, 2016. doi: 10.1145/2976749.2978318. URL https://doi.org/10.1145/2976749.2978318.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting training data from large language models. In *USENIX Security Symposium*, pp. 2633–2650, 2021.

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramèr. Membership inference attacks from first principles. In *IEEE Symposium on Security and Privacy (S&P)*, pp. 1897–1914, 2022. doi: 10.1109/SP46214.2022.9833649. URL https://doi.org/10.1109/SP46214.2022.9833649.

Yuetian Chen, Yuntao Du, Kaiyuan Zhang, Ashish Kundu, Charles Fleming, Bruno Ribeiro, and Ninghui Li. Window-based membership inference attacks against fine-tuned large language models. In *35th USENIX Security Symposium (USENIX Security 26)*, pp. 1187–1206. USENIX Association, 2026a. URL https://www.usenix.org/conference/usenixsecurity26/presentation/chen-yuetian.

Zeyuan Chen, Yihan Ma, Xinyue Shen, Michael Backes, and Yang Zhang. Pop quiz attack: Black-box membership inference attacks against large language models. *arXiv preprint arXiv:2605.06423*, 2026b. URL https://arxiv.org/abs/2605.06423.

DeepSeek-AI. DeepSeek-V2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*, 2024.

DeepSeek-AI. DeepSeek-V4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026. URL https://arxiv.org/abs/2606.19348.

Ruyi Ding, Tianhong Xu, Xinyi Shen, Aidong Adam Ding, and Yunsi Fei. MoEcho: Exploiting side-channel attacks to compromise user privacy in mixture-of-experts LLMs. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pp. 2159–2173, 2025. doi: 10.1145/3719027.3765174. URL https://doi.org/10.1145/3719027.3765174.

Yihong Dong, Xue Jiang, Huanyu Liu, Zhi Jin, Bin Gu, Mengfei Yang, and Ge Li. Generalization or memorization: Data contamination and trustworthy evaluation for large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 12039–12050. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-acl.716. URL https://aclanthology.org/2024.findings-acl.716/.

André Vicente Duarte, Xuandong Zhao, Arlindo L. Oliveira, and Lei Li. DE-COP: Detecting copyrighted content in language models training data. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 11940–11956. PMLR, 2024. URL https://proceedings.mlr.press/v235/duarte24a.html.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022. URL https://www.jmlr.org/papers/v23/21-0998.html.

Wenjie Fu, Huandong Wang, Chen Gao, Guanghua Liu, Yong Li, and Tao Jiang. Membership inference attacks against fine-tuned large language models via self-prompt calibration. In *Advances in Neural Information Processing Systems*, volume 37, pp. 134981–135010. Curran Associates, Inc., 2024. doi: 10.52202/079017-4290. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/f36ad694188bb4c4bbbd61e2038e069e-Abstract-Conference.html.

Felix Hamborg, Norman Meuschke, Corinna Breitinger, and Bela Gipp. news-please: A generic news crawler and extractor. In *Proceedings of the 15th International Symposium of Information Science*, pp. 218–223, 2017.

Yu He, Boheng Li, Liu Liu, Zhongjie Ba, Wei Dong, Yiming Li, Zhan Qin, Kui Ren, and Chun Chen. Towards label-only membership inference attack against pre-trained large language models. In *34th USENIX Security Symposium (USENIX Security 25)*, pp. 1609–1628. USENIX Association, 2025. URL https://www.usenix.org/conference/usenixsecurity25/presentation/he-yu.

Luis Ibanez-Lissen, Lorena Gonzalez-Manzano, Jose Maria de Fuentes, Nicolas Anciaux, and Joaquin Garcia-Alfaro. Lumia: Linear probing for unimodal and multimodal membership inference attacks leveraging internal llm states. In *European Symposium on Research in Computer Security*, pp. 186–206. Springer, 2025.

Masahiro Kaneko, Youmi Ma, Yuki Wata, and Naoaki Okazaki. Sampling-based pseudo-likelihood for membership inference attacks. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 8894–8907. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-acl.465. URL https://aclanthology.org/2025.findings-acl.465/.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pp. 611–626, 2023.

Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. GShard: Scaling giant models with conditional computation and automatic sharding. In *International Conference on Learning Representations (ICLR)*, 2021. URL https://openreview.net/forum?id=qrwe7XHTmYb.

Xuechen Li, Florian Tramèr, Percy Liang, and Tatsunori Hashimoto. Large language models can be strong differentially private learners. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=bVuP3ltATMz.

Huakang Lin, Tiancheng Zheng, Mingxuan Sun, Tianhong Xu, Fan Zhang, Yunsi Fei, and Ruyi Ding. Groundhog bit-flip attack: Seeding infinite generation loops in mixture-of-experts llms through bit flips, 2026. URL https://arxiv.org/abs/2608.25276.

Disha Makhija, Manoj Ghuhan Arivazhagan, Vinayshekhar Bannihatti Kumar, and Rashmi Gangadharaiah. Neural breadcrumbs: Membership inference attacks on llms through hidden state and attention pattern analysis. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5608–5620, 2026.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations (ICLR)*, 2017.

Niklas Muennighoff, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Jacob Morrison, Sewon Min, Weijia Shi, Pete Walsh, Oyvind Tafjord, Nathan Lambert, Yuling Gu, Shane Arora, Akshita Bhagia, Dustin Schwenk, David Wadden, Alexander Wettig, Binyuan Hui, Tim Dettmers, Douwe Kiela, Ali Farhadi, Noah A. Smith, Pang Wei Koh, Amanpreet Singh, and Hannaneh Hajishirzi. OLMoE: Open mixture-of-experts language models. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2409.02060.

Milad Nasr, Reza Shokri, and Amir Houmansadr. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *2019 IEEE Symposium on Security and Privacy (SP)*, pp. 739–753. IEEE, 2019. doi: 10.1109/SP.2019.00065. URL https://doi.org/10.1109/SP.2019.00065.

Jianmo Ni, Jiacheng Li, and Julian McAuley. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 188–197, 2019.

Qwen Team. Qwen1.5-MoE: Matching 7B model performance with 1/3 activated parameters. https://qwenlm.github.io/blog/qwen-moe/, 2024.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations (ICLR)*, 2017. URL https://openreview.net/forum?id=B1ckMDqlg.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. Detecting pretraining data from large language models. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=zWqr3MQuNs.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 3–18, 2017. doi: 10.1109/SP.2017.41.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Sun, Jeff Huang, Cody H Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E Gonzalez, et al. Sglang: Efficient execution of structured language model programs. *Advances in neural information processing systems*, 37:62557–62583, 2024.