A GRAPH-THEORETIC APPROACH TO PRIVACY AND ROBUSTNESS IN LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) face security risks from both unintended disclosure and adversarial manipulation: they may reproduce sensitive training content, leak membership information, and exhibit substantial performance degradation under small targeted prompt perturbations. Although these security threats differ in their manifestation, they are influenced by how an LLM internally routes attention across tokens. Existing approaches lack a common, training-free framework to study LLM security by analyzing and intervening on this internal process. We adopt a common graph-theoretic lens, representing the causally masked attention map of a decoder-only LLM as a directed acyclic graph, with tokens as nodes and attention scores as weighted edges. Using this representation, we introduce two complementary inference-time approaches for analyzing LLM security vulnerabilities: an edge-centric intervention on attention weights for privacy and a node-centric intervention for context robustness. First, ATTENTIONSWAP identifies structural agreements in dominant edges of a primary model and an auxiliary model before swapping their weights to disrupt verbatim memorization while preserving utility on the downstream tasks. Second, BOTDROP identifies topologically critical tokens and prunes them to stress-test prompt sensitivity. Extensive empirical evaluations demonstrate the effectiveness of this shared graph-based perspective for addressing different LLM security threats through our novel training-free, inference-time interventions.

1 INTRODUCTION

Large language models (LLMs) exhibit security vulnerabilities at seemingly different levels. They can reproduce memorized training sequences, potentially exposing sensitive or proprietary information, and they can be highly sensitive to small prompt perturbations that substantially degrade model behavior (Kiyomaru et al., 2024; Zhu et al., 2024; Romanou et al., 2026). Although these vulnerabilities manifest differently, studying them only through their task-specific outputs can overlook a shared internal substrate: how information is routed among tokens. Both memorized sequence reproduction and prompt sensitivity depend on which token interactions dominate the model's internal computation. In this work, we ask whether these apparently different vulnerabilities can be represented and addressed through a common interface of information routing within an LLM. Our key observation is that the self-attention structure of transformers provides such a representation. The success of LLMs across diverse natural language processing and reasoning tasks is fundamentally rooted in the Transformer architecture, where the attention mechanism serves as the primary engine for context processing and representational routing. Self-attention enables LLMs to construct rich, contextualized representations by dynamically weighting the relationships between tokens (Vaswani et al., 2017; Clark et al., 2019).

Recognizing the structural elegance of the self-attention mechanism, an emerging body of literature has sought to bridge the attention mechanism with graph theory, mapping attention matrices to directed acyclic graphs (DAGs) where nodes correspond to tokens and attention weights represent edge capacities (Abnar & Zuidema, 2020). Existing efforts, however, remain largely specialized to particular applications. Some works employ graph formulations for downstream tasks such as hallucination detection (Kong et al., 2025; Frasca et al., 2025; Kurra et al., 2026), artificial text detection (Kushnareva et al., 2021), or token pruning and routing (Wang et al., 2024). Comparatively

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.

less attention has been devoted to applying graph-theoretic analyses directly on autoregressive LLM attention dynamics to diagnose internal model behaviors.

In this work, we model the token-level attention map of autoregressive LLMs as a DAG and use this graph-based abstraction as a diagnostic lens for studying LLM vulnerabilities. Our contribution is to use the attention DAG as a common interface for security-oriented structural interventions. This enables a unified computational pipeline: construct the attention DAG, identify the security-relevant structure, and apply a sparse intervention to it. Given an attention graph, we apply a task-specific graph operator to identify such an object (e.g., a subset of edges or nodes), and then intervene only on that selected sub-structure. We showcase the diagnostic power of this framework through two standalone, training-free, inference-time methods for privacy and robustness.

Privacy Defense. LLMs trained on web-scale corpora are known to reproduce portions of their training data verbatim, creating privacy risks including leakage of personally identifiable information, confidential records, and other sensitive or proprietary content (Carlini et al., 2021; Tirumala et al., 2022; Biderman et al., 2023a; Kiyomaru et al., 2024; Karamolegkou et al., 2023a; Mueller et al., 2024). A related threat is Membership Inference Attacks (MIAs), which seek to determine whether a specific data sequence was included in the training set (Carlini et al., 2022; Shi et al., 2024; Zhang et al., 2025), which aim to determine whether a specific data sequence was part of the training set. When training-data membership is sensitive, such attacks can reveal that a particular record or individual was included in a restricted dataset, potentially disclosing a non-public association with sensitive domains such as medical or legal data, even without recovering the underlying text (Mireshghallah et al., 2022), posing serious ethical and legal repercussions.

Thus, memorization reduction can have far-reaching benefits— reducing the risk of reproducing protected training data and shielding models from targeted privacy attacks. Training-based defenses—including data curation, architectural modifications, changes to the training objective, or post-hoc unlearning—require additional training or retraining leading to substantial computational costs and may introduce privacy–utility tradeoffs (Kandpal et al., 2022; Ghosal et al., 2025; Shi et al., 2022). Existing training-free approaches primarily operate at the output level, modifying token probabilities (Prashant et al., 2026) or inferring membership from token likelihoods (Shi et al., 2024; Zhang et al., 2025; Xie et al., 2024; Mattern et al., 2023), without directly intervening on the model's internal attention structure. To address these limitations, we propose ATTENTIONSWAP, a training-free, post-hoc approach inspired by two key observations: *First,* Shilov et al. (2026) show that LLMs exhibit a "mosaic memory" where memorized content is assembled from fuzzy-duplicate sequences, suggesting that memorization is a routing phenomenon through specific attention pathways. Consistent with this, (Zaree et al., 2026) demonstrate that internal attention dynamics carry strong membership signals. *Second,* smaller models memorize substantially less than larger ones (Prashant et al., 2026), making them excellent references for non-memorized attention structures. Specifically, ATTENTIONSWAP acts as an inference-time defense that identifies "topological agreements" between a large main model and a small auxiliary model. The central idea is that by substituting the main model's attention weights along these common edges with those of the auxiliary model, we can disrupt the deterministic routing pathways that feed verbatim recall to mitigate MIA risk while preserving the larger model's downstream reasoning capabilities.

Robustness Probing. LLMs are also known to be highly brittle. Small perturbations to the input prompt can disproportionately degrade their outputs (Zhu et al., 2024; Romanou et al., 2026). Our node-centric method, BOTDROP, is an offline adversarial stress-testing framework designed to systematically map, analyze, and evaluate prompt context sensitivity. As an efficient token perturbation technique, BOTDROP acts as a robustness diagnostic to identify topologically salient tokens and evaluate which prompt constructions are most resilient to perturbation. Alternatively, it may be adversarially adapted for sabotaging a target model's utility using minimal pruning budgets.

Instead of relying on computationally expensive gradient searches (Sundararajan et al., 2017) for finding important tokens or random deletions, BOTDROP treats the attention DAG as an information transmission network. Under this framework, paths between tokens act as routing channels where the capacity is bounded by the weakest link (the "widest path" problem). We introduce a graph-theoretic centrality metric that counts how often a token acts as a bottleneck on the widest paths between all node pairs. BOTDROP strategically prunes a minimal budget of these bottleneck tokens to produce a distorted prompt. The main idea is that even with a small pruning budget, this perturbed prompt can cause notable performance degradation when evaluated on a black-box target model.

Our Contributions. Our contributions are as follows. We represent token-level attention in autoregressive LLMs as a DAG, establishing a shared graph-based view for studying different security vulnerabilities. This abstraction yields a common pipeline that identifies security-relevant graph structures and selectively intervenes on them using task-specific operators. As an edge-centric privacy intervention, we introduce ATTENTIONSWAP, a training-free inference-time method that identifies cross-model agreement on dominant attention edges and selectively modifies their weights to reduce memorization and membership leakage while preserving model utility. For robustness probe, we design BOTDROP to identify and remove routing-critical prompt tokens to expose sensitivity to structural context loss. To demonstrate the effectiveness of these approaches, we perform extensive experiments on memorization, membership inference, and reasoning benchmarks.

2 PRIVACY DEFENSE AND ROBUSTNESS PROBING AS GRAPH PROBLEMS

We formulate LLM security analysis through a common graph-theoretic abstraction of internal information routing. Given an autoregressive LLM and an input sequence, we represent self-attention as a weighted DAG, where tokens correspond to nodes and attention relations correspond to weighted edges. Rather than designing independent representations for individual security problems, we use this attention graph as a shared structural interface. A security objective is then instantiated by (i) identifying the graph structures most relevant to the objective and (ii) applying a targeted intervention to those structures. Under this formulation, privacy mitigation becomes an *edge-centric intervention problem*, whereas robustness analysis becomes a *node-centric intervention problem*.

2.1 THE GRAPH FRAMEWORK

The self-attention mechanism in Transformer exhibits parallels with message-passing operations on fully connected graphs (El et al., 2025). Formally, an input sequence of N tokens can be represented as a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the node set $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$ corresponds directly to the sequence of tokens. In this work, we focus exclusively on autoregressive, decoder-only language models. This dictates that information can only flow causally from an earlier token to a later token. In this context, **causality** entails that a directed edge $e_{i,j} \in \mathcal{E}$ pointing from node v_i to node v_j exists if and only if $i < j$. The resulting topology of $\mathcal{G}$ is therefore a directed acyclic graph (DAG) where: $\mathcal{E} = \{e_{i,j} \mid 1 \leq i < j \leq N\}$. The edge weight $w(e_{i,j})$, representing the relative magnitude of information propagated from token v_i to token v_j at a given layer l, is determined by the corresponding attention score. Let $h_i \in \mathbb{R}^{d_{model}}$ and $h_j \in \mathbb{R}^{d_{model}}$ denote the hidden representation vectors from the preceding layer $l-1$ for tokens v_i and v_j, respectively. Under the standard multi-head attention formulation, the edge weight for a single attention head is formalized as $w(e_{i,j}) = \frac{\exp\left((W^Q h_j)^\top (W^K h_i)\right)}{\sum_{k=1}^{j} \exp((W^Q h_j)^\top (W^K h_k))}$, where W^Q and W^K are the learnable query and key projection matrices. This DAG formulation serves as the structural foundation for our subsequent analyses.

2.2 OUR PROBLEM SETUP: STRUCTURAL SELECTION AND INTERVENTION

Let $\mathcal{G}$ denote an attention graph and let $\mathcal{O}$ denote a security objective, such as reducing privacy leakage or measuring robustness to prompt perturbations. We define a task-specific *structural selection operator* $\Phi_{\mathcal{O}}$, which maps the attention graph, together with optional task-specific reference information $\mathcal{R}$, to a security-relevant structural object: $\mathcal{S}_{\mathcal{O}} = \Phi_{\mathcal{O}}(\mathcal{G}; \mathcal{R})$. The selected structure $\mathcal{S}_{\mathcal{O}}$ may consist of nodes, edges, paths, or other graph-derived structures. Given $\mathcal{S}_{\mathcal{O}}$, an *intervention operator* $\Psi_{\mathcal{O}}$ modifies either the model's internal attention computation (changing the edge weights) or the graph's structure directly (e.g. dropping a node): $\widetilde{\mathcal{G}} = \Psi_{\mathcal{O}}\left(\mathcal{G}; \mathcal{R}; \mathcal{S}_{\mathcal{O}}\right)$.

Conceptually, in our framework, the security objective determines which structural property is relevant, the selection operator determines where to intervene, and the intervention operator determines how to act on the selected structure. The overall goal is to identify a compact structural set $\mathcal{S}_{\mathcal{O}}$ whose targeted intervention produces a meaningful change in the corresponding security criterion while respecting a task-specific utility or intervention-budget constraint.

Problem 1: Edge-Centric Privacy Intervention. Let $\mathcal{G}^{\text{main}} = (\mathcal{V}, \mathcal{E}, \mathcal{W}^{\text{main}})$ and $\mathcal{G}^{\text{aux}} = (\mathcal{V}, \mathcal{E}, \mathcal{W}^{\text{aux}})$ denote the attention DAGs of a main model and a smaller auxiliary model over the same token sequence. Our goal is to identify a sparse subset of attention edges whose targeted modification reduces memorization-related privacy leakage while preserving the utility of the main model. Formally, let $\mathcal{S}_{\text{priv}} = \Phi_{\text{priv}}\left(\mathcal{G}^{\text{main}}, \mathcal{G}^{\text{aux}}\right) \subseteq \mathcal{E}$ denote the set of security-relevant edges selected by a graph operator Φ_{priv}. Given $\mathcal{S}_{\text{priv}}$, an intervention operator Ψ_{priv} produces a modified attention-weight matrix $\widetilde{\mathcal{W}}^{\text{main}} = \Psi_{\text{priv}}\left(\mathcal{W}^{\text{main}}, \mathcal{W}^{\text{aux}}, \mathcal{S}_{\text{priv}}\right)$. Our objective is to select $\mathcal{S}_{\text{priv}}$ such that the modified model $f_{\widetilde{\mathcal{W}}^{\text{main}}}$ minimizes privacy leakage $\mathcal{L}_{\text{priv}}$ subject to a utility constraint $\mathcal{L}_{\text{util}}$ relative to the original model $f_{\mathcal{W}^{\text{main}}}$:

$$\min_{\mathcal{S}_{\text{priv}} \subseteq \mathcal{E}} \mathcal{L}_{\text{priv}}\left(f_{\widetilde{\mathcal{W}}^{\text{main}}}\right) \quad \text{s.t.} \quad \mathcal{L}_{\text{util}}\left(f_{\widetilde{\mathcal{W}}^{\text{main}}}, f_{\mathcal{W}^{\text{main}}}\right) \leq \epsilon$$

In our method, ATTENTIONSWAP, Φ_{priv} identifies structurally agreed-upon dominant edges between $\mathcal{G}^{\text{main}}$ and $\mathcal{G}^{\text{aux}}$, while Ψ_{priv} selectively "swaps" their corresponding attention weights.

Problem 2: Node-Centric Robustness Intervention. Let $\mathcal{G}^{\text{sur}} = (\mathcal{V}, \mathcal{E}, \mathcal{W}^{\text{sur}})$ denote the attention DAG constructed from a surrogate model for an input sequence $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$, and let f^{tar} denote a black-box target model. Our goal is to identify a sparse subset of token nodes whose removal maximally degrades the utility of the target model, thereby exposing its sensitivity to structurally important prompt context.

Formally, let $\mathcal{S}_{\text{rob}} = \Phi_{\text{rob}}(\mathcal{G}^{\text{sur}}) \subseteq \mathcal{V}$ denote the set of security-relevant nodes selected by a graph operator Φ_{rob}. Given $\mathcal{S}_{\text{rob}}$, an intervention operator Ψ_{rob} removes the corresponding tokens from the original input to produce a perturbed sequence $\widetilde{\mathcal{V}} = \Psi_{\text{rob}}(\mathcal{V}; \mathcal{S}_{\text{rob}})$. Our objective is to select $\mathcal{S}_{\text{rob}}$ such that the resulting perturbed sequence maximizes degradation in the target model's utility under a constrained token-removal budget:

$$\max_{\mathcal{S}_{\text{rob}} \subseteq \mathcal{V}} \mathcal{L}_{\text{rob}}\left(f^{\text{tar}}(\widetilde{\mathcal{V}}), f^{\text{tar}}(\mathcal{V})\right) \qquad \text{s.t.} \qquad |\mathcal{S}_{\text{rob}}| \leq k,$$

where $\mathcal{L}_{\text{rob}}$ measures the degradation in target-model performance induced by the intervention and k denotes the token-removal budget.

In our method, BOTDROP, Φ_{rob} identifies routing-critical nodes according to their frequency of occurrence as interior nodes on widest paths in $\mathcal{G}^{\text{sur}}$, while Ψ_{rob} removes the tokens corresponding to $\mathcal{S}_{\text{rob}}$ from the input prompt.

3 OUR METHODS

3.1 ATTENTIONSWAP: AN EDGE-CENTRIC MECHANISM FOR PRIVACY

Our method builds upon the core intuition of TokenSwap (Prashant et al., 2026). TokenSwap is an output-level defense that mitigates memorization by swapping the next-token vocabulary logits of a large main model with those of a smaller auxiliary model exclusively for high-frequency syntactic tokens. This design exploits the fact that while smaller models lack the parameter capacity to memorize long pretraining sequences verbatim, they remain highly capable of processing basic grammatical and syntactic structures. However, the success of TokenSwap relies on a heuristic list of target words. Our ATTENTIONSWAP avoids this by shifting the intervention upstream into the latent layers. Instead of heuristically swapping output logits for pre-selected words, we dynamically identify internal token-routing pathways where both models agree. By swapping the attention weights on these shared backbone edges, we break the deterministic pathways that feed verbatim recall in the main model while preserving its superior downstream reasoning capabilities.

3.1.1 STRUCTURAL AGREEMENT VIA MAXIMUM SPANNING ARBORESCENCE

Let $\bar{A}^{\text{main}}, \bar{A}^{\text{aux}} \in \mathbb{R}^{N \times N}$ denote the target layer's attention matrices for the main and auxiliary models, respectively, where the head dimension is averaged: $\bar{A}_{j,i} = \frac{1}{H}\sum_{h=1}^{H} A_{j,i}^{(h)}$. These matrices serve as the weighted adjacency matrices of the causally-masked DAGs: $\mathcal{G}^{\text{main}} = (\mathcal{V}, \mathcal{E}^{\text{main}})$ and $\mathcal{G}^{\text{aux}} = (\mathcal{V}, \mathcal{E}^{\text{aux}})$. For any destination node (query token) $v_j \in \mathcal{V}$, we extract its maximum incoming edge $e_{i^*,j}$ where: $i^* = \arg\max_{1<i<j} \bar{A}_{j,i}$.

Because the causal mask restricts the topology to a DAG, selecting the single maximum incoming edge for each node v_j yields the Maximum Spanning Arborescence (MSA) of the graph. We exclude v_1 (initial token) from the $\arg\max$ operation because it acts as a known attention sink (Xiao et al., 2024; Barbero et al., 2025), which is far from a sequence-specific semantic signal. We identify a structural agreement at node v_j if both the main and auxiliary models select the same incoming edge to construct their respective MSAs. We define the binary agreement matrix $M \in \{0,1\}^{N\times N}$ as:

$$M_{j,i} = \mathbb{I}(\underset{1<k<j}{\operatorname{argmax}}\, \bar{A}^{\text{main}}_{j,k} = \underset{1<k<j}{\operatorname{argmax}}\, \bar{A}^{\text{aux}}_{j,k} = i)$$

where $\mathbb{I}(\cdot)$ is the indicator function.

3.1.2 Attention Modification and Redistribution

Once structural agreements are identified, we perform a targeted intervention. For matched edges ($M_{j,i} = 1$), we "swap" the main model's attention weight with the auxiliary model's weight. The attention delta is given by: $\Delta_{j,i} = \left(\bar{A}^{\text{aux}}_{j,i} - \bar{A}^{\text{main}}_{j,i}\right) \cdot M_{j,i}$.

To maintain the row-stochastic property of the attention matrix, any mass adjusted on the matched edge must be redistributed across the remaining active edges. We define the set of valid, non-matched autoregressive target nodes $\mathcal{D}_j$ as: $\mathcal{D}_j = \{i \mid 1 \le i < j,\, M_{j,i} = 0,\, v_i \notin \mathcal{V}_{\text{pad}}\}$. The resulting attention weight adjustment matrix $W_\Delta \in \mathbb{R}^{N\times N}$ is formulated as:

$$W_{\Delta,j,i} = \begin{cases} \Delta_{j,i} & \text{if } M_{j,i} = 1 \\ -\frac{\sum_k \Delta_{j,k}}{|\mathcal{D}_j|} & \text{if } i \in \mathcal{D}_j \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Instead of modifying attention matrices in-place, AttentionSwap directly injects the correction term $W_{\Delta,j,i}$ within the representation space. Let $V \in \mathbb{R}^{B\times N\times d_{\text{attn}}}$ be the concatenated value state representations across all attention heads and W^O be the trained output projection matrix. The updated attention block output h^{new}_j is computed as $h^{\text{new}}_j = h_j + W_\Delta V (W^O)^\top$ where h_j is the standard attention output layer before residual addition.

AttentionSwap Against MIA. Recent literature has revealed a strong relationship between the attention mechanism and memorization, which may facilitate MIA (Zaree et al., 2026). Therefore, it is natural to leverage AttentionSwap to defend against MIA (see Appendix A.2 for the formal definition). We propose two variants of AttentionSwap for this purpose: (1) AttS-aux: this is the vanilla AttentionSwap method, with the attention modification shown in Equation 1. (2) AttS-rand: this method substitutes the above attention correction with a random uniform matrix $U(-1,1)$ while preserving every other step of AttentionSwap. For both methods, we have $M_{j,j} = 0,\ \forall j \in [N]$.

3.2 BotDrop: A Node-Centric Approach to Robustness

While AttentionSwap protects model privacy by disrupting sequence recall capability, the graph structure can also be used to identify bottleneck tokens whose removal provides a targeted test of context dependence or LLM robustness to prompt perturbation. To this end, we introduce **BotDrop**, an adversarial framework designed to degrade an LLM's capability by strategically pruning a minimal budget of k prompt tokens. Our approach operates under a realistic black-box transfer attack paradigm. The user's clean input prompt is intercepted and passed through a locally accessible *surrogate model* to analyze its internal attention graph topologies. From these topologies, BotDrop identifies and prunes the core information routing bottlenecks, producing a distorted prompt. This mutated prompt is then transmitted to the black-box *target model*, successfully sabotaging its task performance and reasoning capabilities.

Information Transmission. In our causal attention DAG $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a path $\mathcal{P} = (u_0, u_1, \ldots, u_m)$ from a source token $s = u_0$ to a destination token $t = u_m$ acts as a multi-step message-passing route. Under this framework, the capacity of a path is bounded by its weakest link. We define the *path width* of $\mathcal{P}$ as the bottleneck attention weight along the path: $\text{width}(\mathcal{P}) = \min_{0\le i<m} w(e_{u_i,u_{i+1}})$. For any pair of tokens (s,t) with $s < t$, the widest path value $\mathcal{W}^*(s,t)$ represents the maximum possible bottleneck capacity among all valid directed paths connecting them: $\mathcal{W}^*(s,t) = \max_{\mathcal{P}\in\mathbf{P}_{s,t}} \text{width}(\mathcal{P})$,

where $\mathbf{P}_{s,t}$ is the set of all directed paths from s to t in $\mathcal{G}$. Because the causal mask of a decoder-only model guarantees that $\mathcal{G}$ is a DAG, we can compute the widest paths from a given source node s to all downstream targets $t > s$ in a single topological pass using dynamic programming (see Algorithm 1 in the appendix). By exploiting the sorted sequence indices, Algorithm 1 runs in $O(N^2)$, where N is the number of tokens.

Widest-Path Routing Frequency. To quantify the degree to which a token acts as a bottleneck for prompt-wide information routing, we introduce the metric *Widest-Path Routing Frequency (WPRF)*. Our metric counts how often each token v appears as an interior node ($s < v < t$) on the widest paths of all node pairs:

$$F_{\text{WPR}}(v) = \sum_{s=1}^{N-2} \sum_{t=s+2}^{N} \mathbb{I}\big(v \in \mathcal{P}^*_{s,t} \setminus \{s,t\}\big) \tag{2}$$

where $\mathbb{I}(\cdot)$ is the indicator function and $\mathcal{P}^*_{s,t}$ is the computed widest path from s to t. This frequency reveals the critical communication junctions of the attention space. To maximize the impact of the token-drop budget, we identify the alphanumeric tokens with the highest routing frequency scores.

4 Experiments

We show results for our three-tier experiments to empirically verify the applicability of our graph-based methods for: (1) reducing memorization, (2) defending against MIA, and (3) degrading LLMs' performance via prompt manipulation. Our code is available at `https://anonymous.4open.science/r/attn_graph-E56F/`

4.1 Evaluation Framework

4.1.1 Datasets

Memorization. To assess verbatim memorization and downstream capability preservation, we leverage target-specific text corpora alongside standard reasoning benchmarks. First, following prior work establishing widespread LeetCode problem memorization (Karamolegkou et al., 2023b), we evaluate `Llama-3-8B` on LeetCode problem statements (Xia et al., 2025). Given a 20-token prompt prefix, the model samples a 100-token continuation. Second, we evaluate `SmolLM2-1.7B-Instruct` on `SmolTalk` (allal et al., 2025). Finally, to ensure our interventions preserve general capabilities, downstream model utility is measured across five standard benchmarks: ARC, WinoGrande, OpenBookQA, SciQ, and MMLU.

MIA. We use the two most common datasets in the literature of MIA for LLMs, namely: WikiMIA (Shi et al., 2024) and MIMIR (Duan et al., 2024). The former labels ground-truth membership based on temporal cutoffs of Wikipedia events with varying sequence lengths and paraphrased settings. The latter derives member and non-member samples from diverse domains in the Pile dataset (Gao et al., 2020), with minimal distribution shifts, and maximizes similarity between the two sets (see Appendix A.4.1 for more details and statistics).

Prompt Perturbation. We utilize three logical and mathematical reasoning benchmarks: GSM8K, StrategyQA (Geva et al., 2021), and ZebraLogic (Lin et al., 2025).

4.1.2 Baselines

Memorization Reduction Methods. We compare AttentionSwap against three primary configurations: (1) Standard, representing the default model behavior with no intervention; (2) TokenSwap (Prashant et al., 2026), a state-of-the-art (SOTA) output-level defense that intercepts and swaps vocabulary logits; and (3) a Combined baseline. Since AttentionSwap modifies upstream, internal routing topologies while TokenSwap operates at the late-stage decoding boundary, the two frameworks are entirely orthogonal and integrate seamlessly, allowing us to evaluate their joint effect in defense.

MIA and Defense Methods. We evaluate our defense against two classes of state-of-the-art MIA methods for LLMs: 1) *Token-likelihood scoring* methods, including Min-K (Shi et al., 2024) and Min-K++ (Zhang et al., 2025), which respectively score membership from token-level conditional

likelihoods by probing outlier tokens and normalized mode likelihoods. 2) *Contrastive likelihood* methods, including RECALL (Xie et al., 2024) and Neighbor (Mattern et al., 2023), which respectively compare the target likelihood against likelihoods under non-member contexts and semantically perturbed neighbors.

We evaluate defense against the above attacks for our two defense variants, ATTS-AUX and ATTS-RAND, along with two random baselines. They both use the same random attention delta matrix as ATTS-RAND, but with a mask matrix different from the agreement matrix M of Equation 3.1.1. (1) TOP-RAND: for every token, this substitutes the attention value to the highest attended other token—the incoming edge with the highest weight—with a random value using mask $M_{j,i} = \mathbb{I}(\text{argmax}_k \bar{A}^{\text{main}}_{j,k} = i)$. (2) TOK-RAND: this substitutes a single attention value across every row of the target model's attention matrix as $M_{j,i} = \mathbb{I}(k_j = i)$, s.t. $k_j \sim U(\{1, \ldots, N\})$, and $\forall j \in [N]$. Note that for all the above methods we have $M_{j,j} = 0,\ \forall j \in [N]$.

Prompt Perturbation Methods. To evaluate BOTDROP 's effectiveness in locating critical tokens, we compare it against five baselines under identical budget constraints k: (1) Standard, measuring accuracy on unperturbed prompts; (2) Random, randomly pruning k tokens; (3) Top Outflow, pruning token at $\arg\max_i \sum_j A_{j,i}$; (4) Value Norm, pruning token at $\arg\max_i \sum_j A_{j,i}\|V_i\|$ where $\|V_i\|$ is the Value vector; (5) Yake (Campos et al., 2020), an unsupervised statistical keyword extractor.

Table 1: Memorization reduction and downstream utility across defense configurations. The Combined baseline maximizes privacy protection while ATTENTIONSWAP provides a superior performance trade-off over TokenSwap.

	LeetCode			ARC-C	WinoGrande	OpenBook
	ROUGE-L ↓	Levenshtein ↑	Frac. EM ↓	Accuracy ↑		
Standard	0.5788	0.4113	0.2227	0.79	0.59	0.72
TokenSwap	0.2517	0.6746	0.1026	0.78	0.59	0.71
ATTENTIONSWAP	0.3764	0.5907	0.1477	**0.78**	**0.60**	**0.73**
Combined	**0.2188**	**0.6962**	**0.093**	**0.78**	**0.60**	**0.73**

(a) Main: LLaMa3-8B / Auxiliary: MobileLLM-R1-140M-base

	SmolTalk2 - SystemChats			ARC-E	SciQ	MMLU-Religion
	ROUGE-L ↓	Levenshtein ↑	Frac. EM ↓	Accuracy ↑		
Standard	0.2424	0.6783	0.1414	0.72	0.88	0.72
TokenSwap	0.2035	0.6986	0.1138	0.29	0.23	0.37
ATTENTIONSWAP	0.2252	0.6925	0.1176	**0.68**	**0.57**	**0.72**
Combined	**0.1862**	**0.7117**	**0.0981**	0.27	0.23	0.37

(b) Main: SmolLM2-1.7B-Instruct / Auxiliary: SmolLM-135M

4.1.3 PERFORMANCE METRICS

Memorization Metrics. We adopt three standard metrics widely established in prior memorization literature: ROUGE-L, Levenshtein distance, and the fractional exact match rate. ROUGE-L measures the longest common subsequence to capture structural alignment between the model's generated completions and the ground-truth data. Levenshtein distance computes character- or token-level edit discrepancies to quantify fine-grained deviations from verbatim recall, while the fractional exact match rate tracks the proportion of tokens reproduced with absolute fidelity.

Reference-Free Downstream Evaluation. The standard MIA datasets we use are fact-based and consist solely of facts x without answers y. The lack of a reference answer renders standard value-/set-matching evaluation metrics inapplicable. Thus, to compare our defense methods with the baselines, we leverage semantic-similarity checks. Since all the language models we use are causal, they are designed for auto-regression tasks to generate responses given input prompts. In other words, the DAG topology of the causal models implies that $\forall i, j \in [N],\ i < j$, the information of latent state h_i is implicitly encoded in h_j, not vice versa, and is used to generate the next token at $j + 1$. Thus, appending a suffix cannot leak information backward, ensuring no impact on scoring-step MIAs. Given this, for performance-preserving verification, we: (1) append a suffix for a conclusive statement with a few tokens, e.g., `<<Summarize:>>`, to encourage maximal aggregation of infor-

Table 2: ATTS maintains the strongest overall trade-off on MIMIR, competing closely with baselines in AUROC while preserving model semantics and achieving better Cos./AUROC. Best is bold, and second-best is underlined. Lower AUROC, higher Cos., and higher Cos./AUROC are better.

Defense	Min-K			Min-K++			RECALL			Neighbor		
	AUROC	Cos.	Cos./AUROC	AUROC	Cos.	Cos./AUROC	AUROC	Cos.	Cos./AUROC	AUROC	Cos.	Cos./AUROC
TOK-RAND	**0.51**$^{\pm 0.00}$	0.90$^{\pm 0.00}$	1.77$^{\pm 0.01}$	**0.53**$^{\pm 0.00}$	0.90$^{\pm 0.00}$	1.70$^{\pm 0.01}$	**0.50**$^{\pm 0.01}$	0.90$^{\pm 0.00}$	1.80$^{\pm 0.03}$	**0.52**$^{\pm 0.00}$	0.90$^{\pm 0.00}$	1.73$^{\pm 0.00}$
TOP-RAND	**0.51**$^{\pm 0.00}$	0.86$^{\pm 0.00}$	1.69$^{\pm 0.01}$	**0.53**$^{\pm 0.00}$	0.86$^{\pm 0.00}$	1.63$^{\pm 0.01}$	0.53$^{\pm 0.01}$	0.86$^{\pm 0.00}$	1.63$^{\pm 0.02}$	0.53$^{\pm 0.00}$	0.86$^{\pm 0.00}$	1.62$^{\pm 0.00}$
ATTS-RAND	**0.51**$^{\pm 0.00}$	**0.96**$^{\pm 0.00}$	**1.87**$^{\pm 0.01}$	0.54$^{\pm 0.00}$	**0.96**$^{\pm 0.00}$	**1.77**$^{\pm 0.01}$	0.52$^{\pm 0.01}$	**0.96**$^{\pm 0.00}$	1.82$^{\pm 0.03}$	0.53$^{\pm 0.00}$	**0.96**$^{\pm 0.00}$	1.81$^{\pm 0.00}$
ATTS-AUX	**0.51**$^{\pm 0.00}$	**0.96**$^{\pm 0.00}$	**1.87**$^{\pm 0.00}$	0.55$^{\pm 0.00}$	**0.96**$^{\pm 0.00}$	1.76$^{\pm 0.00}$	0.51$^{\pm 0.01}$	**0.96**$^{\pm 0.00}$	**1.89**$^{\pm 0.02}$	**0.52**$^{\pm 0.00}$	**0.96**$^{\pm 0.00}$	**1.84**$^{\pm 0.00}$

mation while assuming minimal deviation from the actual input prompt's signal-to-noise ratio due to the suffix's shortness; (2) measure the semantic similarity between the post-suffix hidden states produced by the model, before and after applying the defense mechanism. A higher similarity shows a lower impact on the model's internals and its performance on the downstream task.

Specifically, to compare the performance of defense methods, we employ three metrics: (1) AUROC of the attacker's performance. The better the defense method, the lower the attacker's success rate is, so a lower AUROC is favorable. (2) Due to a lack of reference, as discussed above, we use cosine similarity, denoted as Cos., to measure semantic similarity. (3) To make MIA results more interpretable and ease comparison, we introduce the composite metric Cos./AUROC, which measures the ratio of Cos. over AUROC. This metric captures defense performance in both attack confrontation and model performance preservation.

4.2 MEMORIZATION RESULTS

Table 1 shows that both TokenSwap and ATTENTIONSWAP reduce verbatim memorization, with their combination yielding the strongest defense. While TokenSwap achieves a higher raw memorization drop, it incurs a severe utility penalty, causing downstream performance to plummet on SmolLM2-1.7B-Instruct. In contrast, ATTENTIONSWAP retains the majority of baseline model capability, delivering a superior privacy-utility trade-off that generalizes across architectures.

4.3 MIA RESULTS

General Setup. Since the attention DAGs used by ATTS are constructed from tokens, the main and auxiliary models should share identical tokenizers. In addition, to remain faithful to the temporal filter of the WikiMIA dataset for membership labeling, we evaluate on LLM models released prior to 2023, or that used pre-trained data collected prior to this date. Therefore, for all experiments and ablation studies, we used Pythia-160m and Pythia-6.9b (Biderman et al., 2023b) for the auxiliary and main models, respectively. This particular choice also adheres to the membership labeling of the MIMIR dataset, as these architectures are pre-trained on the Pile dataset from which MIMIR is derived. We provide ablation studies on more model architectures in Appendix A.6.3.

ATTS: A Performance-friendly Defense. The main results of our MIA experiments are presented in Tables 2 and 6 (Table 6 can be found in section A.5 of the Appendix). Essentially, the MSA matching step in ATTS-AUX and ATTS-RAND constrains attention weight substitutions to consensus pathways. Consequently, unconstrained baselines like TOP-RAND and TOK-RAND inject more noise across attention matrices, leading to a more aggressive reduction in raw membership detection scores (lower AUROC). However, an effective defense must balance privacy protection with utility retention. In this regard, ATTS-AUX consistently preserves downstream task performance, achieving the highest Cosine Similarity across model architectures. When evaluating the joint privacy-utility trade-off via the composite metric (Cos./AUROC), ATTS-AUX achieves the optimal results across almost all attack and dataset scenarios, making it highly beneficial for harmless privacy protection.

4.4 ABLATION STUDIES

Since ATTS substitutes at most a single attention edge weight per token, the marginal number of swapped attention weights increases linearly with the number of tokens, while the number of unchanged weights–size of the attention matrices–increases quadratically. Thus, one might expect ATTS to be less effective for longer sequences. This motivates us to evaluate ATTS for defending

Table 3: Downstream accuracy degradation under a highly restrictive token-pruning budget ($k = 1$) using a cross-family `Qwen-2.5-3B` surrogate. Bottleneck consistently drives the greatest performance disruption across both target models: `Llama3.1-8B-Instruct` and `Gemma2-9B-It`.

	Llama3.1-8B-Instruct			Gemma2-9B-It		
Method	GSM8K	StratQA	ZebraLogic	GSM8K	StratQA	ZebraLogic
Standard	82.00	85.00	92.31	81.00	90.00	92.31
Random	74.00	83.50	89.23	76.00	87.33	90.26
Top Outflow	77.00	85.50	89.23	79.00	87.00	95.38
Value Norm	80.00	84.50	89.23	79.50	87.00	95.38
Yake	81.00	85.00	89.23	80.00	86.50	93.85
Bottleneck	**67.50**	**76.50**	**67.69**	**65.50**	**86.00**	**73.85**

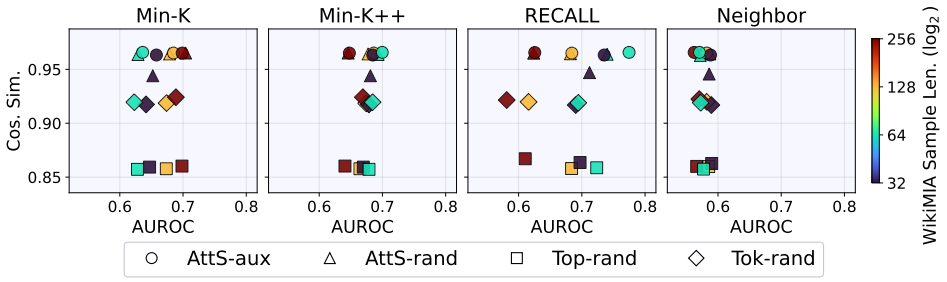


Figure 1: ATTS remains effective across prompt lengths, with all attention-based defenses showing stable or improved attack mitigation on longer WikiMIA sequences while preserving semantics.

against attacks across a range of prompt lengths. We apply the defense baselines across the sample splits of 32, 64, 128, and 256 tokens from WikiMIA. The results of Figure 1 are not generally in favor of the above premise. In other words, the overall pattern weakly suggests that all the attention-based defense baselines, including ours, perform steadily or even better on longer prompts, confirming the importance of attention in preventing likelihood- and contrastive-based attackers. Thus, ATTS-AUX and ATTS-RAND are effectively applicable to sequences of greater lengths. In a similar manner, the defense methods' impact on the model's semantics is generally uncorrelated with the prompt length.

4.5 PROMPT-PERTURBATION ATTACK RESULTS

Evaluated under the most restrictive token budget ($k = 1$), BOTDROP consistently drives the highest performance degradation, outperforming the baselines as shown in Table 3. To preserve real-world plausibility, we adopt a black-box transfer paradigm using a smaller, cross-family surrogate model (`Qwen-2.5-3B`). Despite architectural mismatches and size disparities, surrogate-identified bottlenecks transfer effectively to larger models, demonstrating that critical information-routing paths are shared across architectures. Finally, to maintain efficiency under realistic prompt-interception, the attack restricts attention matrix extraction to a single middle layer, proving that effective bottleneck identification requires minimal layer access and low runtime overhead.

5 CONCLUSION

In this work, we have shown that token-level attention graphs provide a common basis for both intervention and diagnosis of LLM security vulnerabilities. We have introduced ATTENTIONSWAP, which uses edge-level correspondence between a primary and an auxiliary model to modify selected edges and mitigate memorization-related privacy risks, including verbatim memorization and membership inference attacks. Complementary to this edge-centric approach, we have developed BOTDROP, a node-centric adversarial probing method. It computes the widest paths on attention graphs to identify and prune critical attention bottleneck tokens enabling stress-testing of model robustness to minimal prompt perturbations. Both methods operate training-free at inference time, demonstrating that internal attention structure can be exploited for security interventions beyond output-level manipulation and likelihood-based analysis across diverse models, tasks, and benchmarks.

AI USE STATEMENT

In this work, we used generative AI tools for: formulating mathematical claims, designing or providing feedback on research methodology or experiments, implementing methods. We have not used generative AI tools for: helping develop theoretical models or conceptual frameworks, proposing or refining hypotheses, cleaning and reformating dataset, interpreting results and generating synthetic data sets, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, assisting with translation, and supporting qualitative and thematic data analysis are not applicable to this work. Additionally, we used generative AI tools for: creating or editing software code, summarizing or analysing existing literature, brainstorming, sourcing/searching for information, editing a research paper to improve readability. We have reviewed all AI-assisted work. Specifically, LLMs were used to assist in discovering relevant literature, but all cited papers were manually reviewed before inclusion in the bibliography. All LLM-generated code was manually verified, tested, and edited for correctness. LLMs were also consulted for minor methodological refinements, but the core research vision and main methodology remain entirely our own. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies security vulnerabilities in LLMs through a graph-based lens on internal attention dynamics. Our proposed methods are intended for defensive and diagnostic purposes, including mitigating memorization-related privacy leakage and evaluating robustness under adversarial prompt perturbations. While our proposed techniques such as BOTDROP could potentially be adapted for offensive prompt engineering, our experiments are conducted in controlled research settings with the goal of improving the safety, reliability, and trustworthy deployment of LLM systems. Any released code or artifacts are intended solely for research purposes and controlled evaluation of LLM security and robustness.

REPRODUCIBILITY STATEMENT

To ensure the reproducibility of our work, we provide our source code at: `https://anonymous.4open.science/r/attn_graph-E56F/`. Detailed experimental setups, hardware configurations, and hyperparameter choices are specified in Sections 4 and Appendix A.4. Furthermore, all experiments are conducted using publicly available datasets introduced in prior literature.

REFERENCES

Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pp. 4190–4197, 2020.

Loubna Ben allal, Anton Lozhkov, Elie Bakouch, Gabriel Martin Blazquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Agustín Piqueres Lajarín, Hynek Kydlíček, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan Son NGUYEN, Ben Burtenshaw, Clémentine Fourrier, Haojun Zhao, Hugo Larcher, Mathieu Morlon, Cyril Zakka, Colin Raffel, Leandro Von Werra, and Thomas Wolf. SmolLM2: When smol goes big — data-centric training of a fully open small language model. In *Second Conference on Language Modeling*, 2025. URL `https://openreview.net/forum?id=3JiCl2A14H`.

Federico Barbero, Alvaro Arroyo, Xiangming Gu, Christos Perivolaropoulos, Petar Veličković, Razvan Pascanu, and Michael M. Bronstein. Why do LLMs attend to the first token? In *Second Conference on Language Modeling*, 2025. URL `https://openreview.net/forum?id=tu4dFUsW5z`.

Stella Biderman, USVSN PRASHANTH, Lintang Sutawika, Hailey Schoelkopf, Quentin Anthony, Shivanshu Purohit, and Edward Raff. Emergent and predictable memorization in large language models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 28072–28090. Curran Associates, Inc., 2023a. URL

https://proceedings.neurips.cc/paper_files/paper/2023/file/59404fb89d6194641c69ae99ecdf8f6d-Paper-Conference.pdf.

Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar Van Der Wal. Pythia: a suite for analyzing large language models across training and scaling. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023b.

Ricardo Campos, Vítor Mangaravite, Arian Pasquali, Alípio Jorge, Célia Nunes, and Adam Jatowt. Yake! keyword extraction from single documents using multiple local features. *Information Sciences*, 509:257–289, 2020.

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 *30th USENIX Security Symposium (USENIX Security 21)*, pp. 2633–2650. USENIX Association, August 2021. ISBN 978-1-939133-24-3. URL https://www.usenix.org/conference/usenixsecurity21/presentation/carlini-extracting.

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramer. Membership inference attacks from first principles. In *2022 IEEE symposium on security and privacy (SP)*, pp. 1897–1914. IEEE, 2022.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. What does BERT look at? an analysis of BERT's attention. In Tal Linzen, Grzegorz Chrupała, Yonatan Belinkov, and Dieuwke Hupkes (eds.), *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 276–286, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4828. URL https://aclanthology.org/W19-4828/.

Arthur Conmy, Augustine Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. *Advances in Neural Information Processing Systems*, 36:16318–16352, 2023.

Sunghee Dong, Sungwon Yi, Kangmin Bae, Jaeyoon Kim, and Seongyeop Kim. Graphshield: Graph-theoretic modeling of network-level dynamics for robust jailbreak detection. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=vGk4D0fUzv.

Michael Duan, Anshuman Suri, Niloofar Mireshghallah, Sewon Min, Weijia Shi, Luke Zettlemoyer, Yulia Tsvetkov, Yejin Choi, David Evans, and Hannaneh Hajishirzi. Do membership inference attacks work on large language models? In *Conference on Language Modeling (COLM)*, 2024.

Batu El, Deepro Choudhury, Pietro Liò, and Chaitanya K Joshi. Towards mechanistic interpretability of graph transformers via attention graphs. *arXiv preprint arXiv:2502.12352*, 2025.

Fabrizio Frasca, Guy Bar-Shalom, Yftah Ziser, and Haggai Maron. Neural message-passing on attention graphs for hallucination detection. *arXiv preprint arXiv:2509.24770*, 2025.

Jian Gao, Xiao Zhang, Miao Li, and Ji Wu. Enhancing elusive clues in knowledge learning by contrasting attention of language models. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence and Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence and Fifteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'25/IAAI'25/EAAI'25. AAAI Press, 2025. ISBN 978-1-57735-897-8. doi: 10.1609/aaai.v39i22.34563. URL https://doi.org/10.1609/aaai.v39i22.34563.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The Pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The pile: An 800gb dataset of diverse text for language modeling. *CoRR*, abs/2101.00027, 2021. URL https://arxiv.org/abs/2101.00027.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361, 2021.

Gaurav Rohit Ghosal, Pratyush Maini, and Aditi Raghunathan. Memorization sinks: Isolating memorization during LLM training. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 19307–19326. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/ghosal25a.html.

Sinan He and An Wang. Graphattack: Exploiting representational blindspots in llm safety mechanisms. *arXiv preprint arXiv:2504.13052*, 2025.

Nikhil Kandpal, Eric Wallace, and Colin Raffel. Deduplicating training data mitigates privacy risks in language models. In *International Conference on Machine Learning*, pp. 10697–10707. PMLR, 2022.

Antonia Karamolegkou, Jiaang Li, Li Zhou, and Anders Søgaard. Copyright violations and large language models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7403–7412, Singapore, December 2023a. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.458. URL https://aclanthology.org/2023.emnlp-main.458/.

Antonia Karamolegkou, Jiaang Li, Li Zhou, and Anders Søgaard. Copyright violations and large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7403–7412, 2023b.

Hirokazu Kiyomaru, Issa Sugiura, Daisuke Kawahara, and Sadao Kurohashi. A comprehensive analysis of memorization in large language models. In Saad Mahamood, Nguyen Le Minh, and Daphne Ippolito (eds.), *Proceedings of the 17th International Natural Language Generation Conference*, pp. 584–596, Tokyo, Japan, September 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.inlg-main.45. URL https://aclanthology.org/2024.inlg-main.45/.

Linggang Kong, Yunlong Zhang, Xiaofeng Zhong, Haoran Fu, Yongjie Wang, and Huijun Liu. Halugnn: Hallucination detection in large language models using graph neural network. *Expert Systems with Applications*, pp. 130857, 2025.

Sailesh Kiran Kurra, Shiek Ruksana, and Borusu Vishal. Unmasking hallucinations: A causal graph-attention perspective on factual reliability in large language models. In *2026 IEEE International Conference on Emerging Computing and Intelligent Technologies (ICoECIT)*, pp. 1–5. IEEE, 2026.

Laida Kushnareva, Daniil Cherniavskii, Vladislav Mikhailov, Ekaterina Artemova, Serguei Barannikov, Alexander Bernstein, Irina Piontkovskaya, Dmitri Piontkovski, and Evgeny Burnaev. Artificial text detection via examining the topology of attention maps. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 635–649, 2021.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori B Hashimoto, Luke Zettlemoyer, and Mike Lewis. Contrastive decoding: Open-ended text generation as optimization. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 12286–12312, 2023.

Bill Yuchen Lin, Ronan Le Bras, Kyle Richardson, Ashish Sabharwal, Radha Poovendran, Peter Clark, and Yejin Choi. Zebralogic: On the scaling limits of LLMs for logical reasoning. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=sTAJ9QyA6l.

Justus Mattern, Fatemehsadat Mireshghallah, Zhijing Jin, Bernhard Schölkopf, Mrinmaya Sachan, and Taylor Berg-Kirkpatrick. Membership inference attacks against language models via neighbourhood comparison. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 11330–11343, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.719. URL https://aclanthology.org/2023.findings-acl.719/.

Fatemehsadat Mireshghallah, Kartik Goyal, Archit Uniyal, Taylor Berg-Kirkpatrick, and Reza Shokri. Quantifying privacy risks of masked language models using membership inference attacks. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 8332–8347, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.570. URL https://aclanthology.org/2022.emnlp-main.570/.

Felix B Mueller, Rebekka Görge, Anna K Bernzen, Janna C Pirk, and Maximilian Poretschkin. Llms and memorization: On quality and specificity of copyright compliance. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, volume 7, pp. 984–996, 2024.

Parjanya Prashant, Kaustubh Ponkshe, and Babak Salimi. Tokenswap: A lightweight method to disrupt memorized sequences in llms. *Advances in Neural Information Processing Systems*, 38: 95743–95777, 2026.

Angelika Romanou, Mark Ibrahim, Candace Ross, Chantal Shaib, Kerem Oktar, Samuel J. Bell, Anaelia Ovalle, Jesse Dodge, Antoine Bosselut, Koustuv Sinha, and Adina Williams. Brittlebench: Quantifying llm robustness via prompt sensitivity, 2026. URL https://arxiv.org/abs/2603.13285.

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 B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 51826–51843, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/e32ad85fa27be4a9868d55703f01323e-Paper-Conference.pdf.

Weiyan Shi, Ryan Shea, Si Chen, Chiyuan Zhang, Ruoxi Jia, and Zhou Yu. Just fine-tune twice: Selective differential privacy for large language models. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 6327–6340, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.425. URL https://aclanthology.org/2022.emnlp-main.425/.

Igor Shilov, Matthieu Meeus, and Yves-Alexandre de Montjoye. The mosaic memory of large language models. *Nature Communications*, 2026.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pp. 3319–3328. PMLR, 2017.

Kushal Tirumala, Aram H. Markosyan, Luke Zettlemoyer, and Armen Aghajanyan. Memorization without overfitting: Analyzing the training dynamics of large language models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=u3vEuRr08MT.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Hongjie Wang, Bhishma Dedhia, and Niraj K Jha. Zero-tprune: Zero-shot token pruning through leveraging of the attention graph in pre-trained transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16070–16079, 2024.

Yunhui Xia, Wei Shen, Yan Wang, Jason Klein Liu, Huifeng Sun, Siyue Wu, Jian Hu, and Xiaolong Xu. Leetcodedataset: A temporal dataset for robust evaluation and efficient training of code llms. *arXiv preprint arXiv:2504.14655*, 2025.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *International Conference on Learning Representations*, volume 2024, pp. 21875–21895, 2024.

Roy Xie, Junlin Wang, Ruomin Huang, Minxing Zhang, Rong Ge, Jian Pei, Neil Zhenqiang Gong, and Bhuwan Dhingra. ReCaLL: Membership inference via relative conditional log-likelihoods. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8671–8689, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.493. URL https://aclanthology.org/2024.emnlp-main.493/.

Pedram Zaree, Md Abdullah Al Mamun, Yue Dong, Ihsen Alouani, and Nael Abu-Ghazaleh. Attenmia: Llm membership inference attack through attention signals. *arXiv preprint arXiv:2601.18110*, 2026.

Jingyang Zhang, Jingwei Sun, Eric Yeats, Yang Ouyang, Martin Kuo, Jianyi Zhang, Hao Frank Yang, and Hai Li. Min-k%++: Improved baseline for pre-training data detection from large language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=ZGkfoufDaU.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. Opt: Open pre-trained transformer language models, 2022.

Xuanming Zhang. Generalization or memorization: Dynamic decoding for mode steering. In *Socially Responsible and Trustworthy Foundation Models at NeurIPS 2025*, 2025. URL https://openreview.net/forum?id=hLTdvMsDaV.

Kaijie Zhu, Jindong Wang, Jiaheng Zhou, Zichen Wang, Hao Chen, Yidong Wang, Linyi Yang, Wei Ye, Yue Zhang, Neil Gong, and Xing Xie. Promptrobust: Towards evaluating the robustness of large language models on adversarial prompts. In *Proceedings of the 1st ACM Workshop on Large AI Systems and Models with Privacy and Safety Analysis*, LAMPS '24, pp. 57–68, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400712098. doi: 10.1145/3689217.3690621. URL https://doi.org/10.1145/3689217.3690621.

A APPENDIX

A.1 RELATED WORK

Information Flow in Transformers. A foundational line of work models Transformer self-attention as a directed graph over tokens, enabling analysis of information flow across layers via layered DAG compositions (Abnar & Zuidema, 2020). Related mechanistic interpretability work further represents Transformers as computational graphs and extracts functional substructures via causal or graph-based methods (Conmy et al., 2023). These approaches primarily aim to interpret internal computation or information flow, whereas we use a graph-based abstraction of attention for diagnosing and intervening on security vulnerabilities in autoregressive LLMs.

Memorization, Privacy Leakage, and Membership Inference in LLMs. Prior work studies memorization and privacy leakage in models, typically through output-level queries.Decoding-time interventions such as contrastive decoding (Li et al., 2023) and TokenSwap (Prashant et al., 2026) modify token distributions, essentially performing output-space manipulation but do not intervene on the internal attention structure. A related line of work leverages cross-model signals and mechanistic interventions to improve model behavior. Gao et al. (2025) use attention discrepancies between large and small models to identify informative but under-emphasized tokens for improving knowledge transfer, while Dynamic Mode Steering (DMS) (Zhang, 2025) frames memorization as a failure of information compression and mitigates it via activation-space interventions. In contrast, we leverage cross-model signals and attention-structure–based interventions at inference time to target memorization (membership leakage) or model sensitivity to prompt perturbations.

Prompt Brittleness and Robustness of LLMs. A third line of work studies the brittleness of LLMs under small or semantically minor prompt perturbations, where minor changes can lead to disproportionately large changes in model outputs, motivating adversarial prompting and jailbreak-style attacks that exploit sensitivity in input space. He & Wang (2025) formulates prompt generation as transformations over external semantic graphs, which are then converted into adversarial inputs to induce jailbreak behavior. Dong et al. (2026) constructs a token-level attention graph from a single forward pass and uses graph structural features over this internal DAG to train a classifier for jailbreak detection at inference time. In contrast, we leverage internal attention dynamics as a graph-based abstraction for both training-free intervention (ATTENTIONSWAP) and surrogate-guided adversarial prompting (BOTDROP).

A.2 BACKGROUND

Membership Inference Attack. Assuming an underlying model, e.g., the LLM, as a function $f : \mathcal{X} \to \mathcal{Y}$, such that a plausible set of textural prompts given as input to an LLM and its textual or numerical answer could exemplify the function's domain and range, and $f \in \mathcal{F}$, where $\mathcal{F}$ is the set of model functions. Once f is trained on a dataset $\mathcal{D} \subset \mathcal{X} \times \mathcal{Y}$, the objectives of MIA is to design a scoring function $h : \mathcal{X} \times \mathcal{Y} \times \mathcal{F} \to [0, 1]$ that observes the input prompt $x \in \mathcal{X}$ to the pre-trained model function and its answer as $y = f(x) \in \mathcal{Y}$, and classifies $(x, y') \in \mathcal{D}$, if $h(x, y; f) \geq \tau$, called a member, and $(x, y') \notin \mathcal{D}$ otherwise, called non-member, where $y' \in \mathcal{Y}$ could be any groundtruth answer or be absent. Accordingly, a defense mechanism against MIA aims to reduce the attacker's prediction performance.

A.3 DETAILS OF BOTDROP IMPLEMENTATION

Algorithm 1 outlines an $O(N^2)$ dynamic programming approach to find the widest paths from a source node s to all downstream targets $t > s$ in a single topological pass, where N denotes the total number of tokens in the sequence.

A.4 EXPERIMENTAL SETTINGS

Table 4 summarizes the datasets and metrics used across our evaluation settings.

Algorithm 1 Single-Source Widest Path (SSWP)

Require: Layer-averaged attention matrix $\bar{A}$; source node $s \in \{1, \ldots, N\}$
$C \leftarrow [0, 0, \ldots, 0]$ ▷ Capacity array
$C[s] \leftarrow \infty$
$P \leftarrow [-1, -1, \ldots, -1]$ ▷ Parent array
for $u = s$ **to** $N - 1$ **do**
  for $v = u + 1$ **to** N **do**
    $t \leftarrow \min(C[u],\ \bar{A}[v, u])$
    if $t > C[v]$ **then**
      $C[v] \leftarrow t$
      $P[v] \leftarrow u$
    end if
  end for
end for
return Capacities C, Parent Tree P

Table 4: Datasets and Metrics used for Evaluation.

	Task	Datasets	Metrics
Privacy Axis	Memorization reduction	LeetCode, SmolTalk2 (Memo) ARC, WinoGrande, OpenBook, SciQ, MMLU (Util)	ROUGE-L, Levestein, Frac EM Accuracy
	MIA defense	MIMIR, WikiMIA	AUROC Cosine Sim
Robustness Axis	Prompt perturbation	GSM8K, StrategyQA, ZebraLogic 2x2	Accuracy

A.4.1 MIA Datasets

To evaluate our approach, we use WikiMIA (Shi et al., 2024) and MIMIR (Duan et al., 2024), two standard benchmarks that provide gold-standard membership labels for pre-training data detection. WikiMIA establishes membership ground truth based on model-knowledge cutoff timestamps derived from Wikipedia text. It is partitioned into four sequence-length splits (32, 64, 128, and 256) and evaluates both original verbatim texts and ChatGPT-paraphrased variants. In contrast, MIMIR is generated out of the Pile dataset and spans seven diverse domains. MIMIR is significantly more challenging for attackers because it draws member and non-member samples from the same dataset and uses n-gram filtering to maximize textual similarity. Please find the datasets' statistics in Table 5.

A.4.2 More Experimental Details

Across all MIA experiments, we fix the third layer of all architectures for swapping attention weights. We use the full set of samples for the WikiMIA datasets and 2000 out of 20000 samples from the "ngram_7_0.2" split of the MIMIR dataset, which contains 1000 member and 1000 non-member samples. We repeat all experiments across 5 random seeds to assess statistical significance. Furthermore, from four WikiMIA sample splits with prompt lengths of 32, 64, 128, and 256 tokens, we use the 32-length split for the main results. This split is also used across all ablation studies except for ablation on the effect of prompt length, as in Figure 1. Regarding the BotDrop evaluation, we use the attention values at the thirteenth layer for GSM8K and StrategyQA and those at the second layer for ZebraLogic. All baselines are implemented based on the original papers and codes. We used NVIDIA-L40S GPUs and NVIDIA-GeForce-RTX-3090 GPUs.

A.5 More MIA Results

We present the MIA results for WikiMIA-32 in Table 6. AttS maintains a strong balance between utility and privacy by preserving semantic similarity while lowering membership leakage across different attacks. We have also included the visualization of the results of Tables 2 and 6 in Figures 2 and 3 respectively.



Figure 2: Utility privacy trade-off on MIMIR across four MIA attacks. Points on the Pareto frontier are highlighted. Note that the the x-axis is inverted. Lower AUROC indicates better privacy defense, and higher Cosine Similarity indicates better semantic preservation.

Table 5: Dataset statistics used in the MIA experiments.

Dataset	Total	Members	Nonmembers
WikiMIA-32	776	387	389
WikiMIA-64	542	284	258
WikiMIA-128	250	139	111
WikiMIA-256	82	51	31
MIMIR Pile-CC	2000	1000	1000

A.6 More Ablation Studies

A.6.1 MIA And Model Size

The literature on MIA and LLM memorization has widely discussed the effect of the model's parameter count (model size) as a significant factor determining the extent of memorization and the model's vulnerability to leaking information once exposed to attackers. Following up on these studies, we conduct two ablation studies on AttS-aux and AttS-rand to examine the effect of model size. We use WikiMIA-32 for these experiments. First, we fix the main model (Pythia-6.9b) and evaluate performance across auxiliary models with different parameter counts (all from the Pythia family). Next, we do the opposite experiment by fixing the auxiliary model (Pythia-70m) and varying the size of the main model. The results are shown in Figure 4, where the top panel corresponds to the former experiment and the bottom one belongs to the latter.

In general, our defense methods exhibit a noticeable performance drop against larger target models, consistent with prior findings that larger models tend to memorize more. In contrast, their performance remains largely unchanged across different auxiliary model sizes. This is an encouraging observation, as it suggests that a large target model can be effectively defended using a lightweight auxiliary model, substantially reducing computational overhead and making AttS highly scalable.



Figure 3: Utility privacy trade-off on WikiMIA-32 across four MIA attacks. Points on the Pareto frontier are highlighted. Note that the the x-axis is inverted. Lower AUROC indicates better privacy defense, and higher Cosine Similarity indicates better semantic preservation.

Table 6: AttS achieves a balanced utility–privacy trade-off on WikiMIA-32, preserving semantic similarity while reducing membership leakage across attacks. Best is bold and second-best is underlined. Lower AUROC, higher Cos., and higher Cos./AUROC are better.

Defense	Min-K			Min-K++			RECALL			Neighbor		
	AUROC	Cos.	Cos./AUROC	AUROC	Cos.	Cos./AUROC	AUROC	Cos.	Cos./AUROC	AUROC	Cos.	Cos./AUROC
Tok-rand	**0.64**$^{\pm 0.01}$	0.92$^{\pm 0.00}$	1.43$^{\pm 0.01}$	0.68$^{\pm 0.01}$	0.92$^{\pm 0.00}$	1.35$^{\pm 0.01}$	**0.69**$^{\pm 0.05}$	0.92$^{\pm 0.00}$	1.33$^{\pm 0.10}$	**0.59**$^{\pm 0.00}$	0.92$^{\pm 0.00}$	1.56$^{\pm 0.01}$
Top-rand	0.65$^{\pm 0.00}$	0.86$^{\pm 0.00}$	1.33$^{\pm 0.01}$	**0.67**$^{\pm 0.00}$	0.86$^{\pm 0.00}$	1.28$^{\pm 0.01}$	0.70$^{\pm 0.03}$	0.86$^{\pm 0.00}$	1.24$^{\pm 0.05}$	**0.59**$^{\pm 0.00}$	0.86$^{\pm 0.00}$	1.46$^{\pm 0.01}$
AttS-rand	0.65$^{\pm 0.00}$	0.94$^{\pm 0.00}$	1.45$^{\pm 0.01}$	0.68$^{\pm 0.00}$	0.94$^{\pm 0.00}$	1.39$^{\pm 0.01}$	0.71$^{\pm 0.07}$	0.95$^{\pm 0.00}$	**1.34**$^{\pm 0.13}$	**0.59**$^{\pm 0.00}$	0.95$^{\pm 0.00}$	1.61$^{\pm 0.01}$
AttS-aux	0.66$^{\pm 0.00}$	**0.96**$^{\pm 0.00}$	**1.46**$^{\pm 0.00}$	0.68$^{\pm 0.00}$	**0.96**$^{\pm 0.00}$	**1.41**$^{\pm 0.00}$	0.74$^{\pm 0.07}$	**0.96**$^{\pm 0.00}$	1.32$^{\pm 0.13}$	**0.59**$^{\pm 0.00}$	**0.96**$^{\pm 0.00}$	**1.64**$^{\pm 0.00}$

A.6.2 MIA And Targeted Swapping Layer

Two important settings of our method are the number of layers at which the attention edge weights are swapped and the positions of these layers within the model's architecture. Therefore, we monitor AttS-aux's performance across a wide range of settings on the WikiMIA-32 dataset. Broadly, we consider three categories for the position where the swapping layers are located: the first layers, the middle layer, and the last layers of the architecture. Next, we change the number of layers in $\{1, 5, 10\}$. The results are provided in Figure 5 for two attack methods.

The most obvious, and expected, observation is that the more layers across which attention weights are swapped, the less encoded information remains in the hidden states, according to the Cos. metric. However, looking at the AUROC of the attacker, this semantic information loss due to a larger number of targeted swapping layers is accompanied by better defense capability. This behavior is similar to what was observed previously in the ablation studies on prompt length (see Figure 1).

The other intuitive finding is that swapping attention edge weights at the last layers of the transformer model yields a higher marginal gain in defense performance than at other positions. For both attack methods, inferring whether a sample point is a member is harder—lower AUROC—when the attention weights are swapped right before computing the token prediction logits.

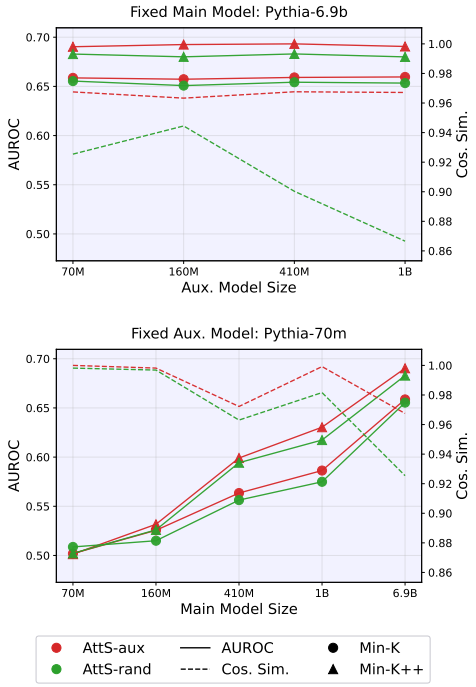


Figure 4: ATTS is largely insensitive to auxiliary model size when the main model is fixed (top), but degrades on larger main models when the auxiliary model is fixed (bottom), enabling scalable defense with lightweight auxiliaries.

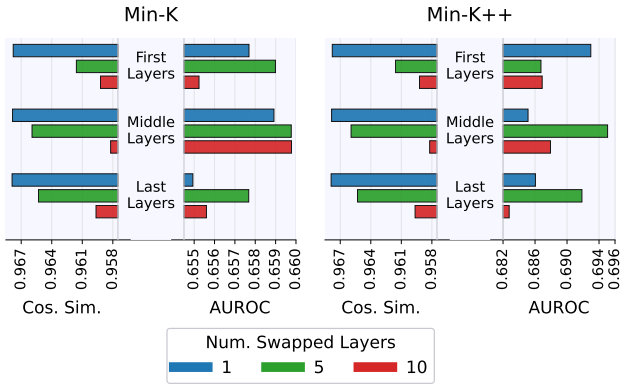


Figure 5: ATTS-AUX trades semantic preservation for stronger defense as more layers are swapped, with last-layer swaps yielding the largest AUROC reduction across attacks on WikiMIA-32.

A.6.3 MIA AND MODEL ARCHITECTURE

Although we propose ATTS-AUX and ATTS-RAND for defense against MIA with only the assumption that the main and auxiliary models share the same tokenizer, we are interested in how these methods perform when we use a different family of architectures. To achieve this, we choose three standard LLM architectures, namely Pythia (Biderman et al., 2023b), OPT (Zhang et al., 2022), and GPT-Neo (Gao et al., 2021).

Table 7 includes the empirical results of this experiment, which shows that the general pattern observed in Tables 6 and 2 keeps appearing with the use of other LLM architectures: ATTS-AUX and ATTS-RAND attain a better tradeoff of defense success rate and downstream task performance preservation measured by similarity in encoded semantics before and after defense. Moreover, one can see that, compared with TOK-RAND and TOP-RAND, our method ATTS not only has less se-

mantic information deviation, but also offers better defense performance when the architecture has fewer parameters.

Table 7: Main and auxiliary model type ablation.

Model Pair	Defense	Min-K		Min-K++	
		AUROC	Cos.	AUROC	Cos.
Main: Pythia-6.9B Aux: Pythia-160M	TOK-RAND	**0.64**$^{\pm 0.01}$	0.92$^{\pm 0.00}$	0.68$^{\pm 0.01}$	0.92$^{\pm 0.00}$
	TOP-RAND	0.65$^{\pm 0.00}$	0.86$^{\pm 0.00}$	**0.67**$^{\pm 0.00}$	0.86$^{\pm 0.00}$
	ATTS-RAND	0.65$^{\pm 0.00}$	0.94$^{\pm 0.00}$	0.68$^{\pm 0.00}$	0.94$^{\pm 0.00}$
	ATTS-AUX	0.66$^{\pm 0.00}$	**0.96**$^{\pm 0.00}$	0.68$^{\pm 0.00}$	**0.96**$^{\pm 0.00}$
Main: OPT-6.7B Aux: OPT-125M	TOK-RAND	**0.61**$^{\pm 0.00}$	0.99$^{\pm 0.00}$	**0.64**$^{\pm 0.00}$	0.99$^{\pm 0.00}$
	TOP-RAND	0.62$^{\pm 0.00}$	0.99$^{\pm 0.00}$	0.65$^{\pm 0.00}$	0.99$^{\pm 0.00}$
	ATTS-RAND	0.62$^{\pm 0.00}$	**1.00**$^{\pm 0.00}$	0.65$^{\pm 0.00}$	**1.00**$^{\pm 0.00}$
	ATTS-AUX	0.62$^{\pm 0.00}$	**1.00**$^{\pm 0.00}$	0.65$^{\pm 0.00}$	**1.00**$^{\pm 0.00}$
Main: GPT-Neo-2.7B Aux: GPT-Neo-125M	TOK-RAND	**0.64**$^{\pm 0.00}$	0.98$^{\pm 0.00}$	**0.67**$^{\pm 0.00}$	0.98$^{\pm 0.00}$
	TOP-RAND	**0.64**$^{\pm 0.00}$	0.98$^{\pm 0.00}$	**0.67**$^{\pm 0.00}$	0.98$^{\pm 0.00}$
	ATTS-RAND	**0.64**$^{\pm 0.00}$	**1.00**$^{\pm 0.00}$	**0.67**$^{\pm 0.00}$	**1.00**$^{\pm 0.00}$
	ATTS-AUX	**0.64**$^{\pm 0.00}$	**1.00**$^{\pm 0.00}$	**0.67**$^{\pm 0.00}$	**1.00**$^{\pm 0.00}$