

Literature Review on Mitigation Methods for Data Leakage in Large Language Models

Table of contents

Summary of research papers 3

 Paper 1: Prompt Leakage effect and mitigation strategies for multi-turn LLM Applications..... 3

 Paper 2: Defeating Cerberus: Concept-Guided Privacy-Leakage Mitigation in Multimodal Language Models 4

 Paper 3: Privacy-Aware Decoding: Mitigating Privacy Leakage of Large Language Models in Retrieval-Augmented Generation..... 5

 Paper 4: PATCH: Mitigating PII Leakage in Language Models with Privacy-Aware Targeted Circuit Patching 6

 Paper 5: Privacy-Aware Detection for Large Language Models Using a Hybrid BiLSTM-HMM Approach..... 7

 Paper 6: Investigating How Pre-training Data Leakage Affects Models’ Reproduction and Detection Capabilities 8

Critical Analysis 9

Connections 10

Conflicts between the findings..... 11

Synthesis Matrix for Literature Review..... 12

Detailed Technical Analysis: Theoretical Foundations and Empirical Evidence 15

 The Mechanics of Transformer Memorization and Data Exposure 15

 Advanced Attack Vectors: From Single-Turn to Agentic Multi-Turn 15

 Multimodal Expansion of Leakage Risk 16

 Inference-Time Probabilistic Mitigation 16

Mechanistic Interpretability: Surgical Defense	16
Real-Time Input Filtering and User-Centric Protection.....	17
Conclusion and Strategic Directions.....	17

Summary of research papers

Paper 1: Prompt Leakage effect and mitigation strategies for multi-turn LLM Applications

The rapid proliferation of Large Language Model (LLM) applications has elevated the system prompt—the foundational set of instructions that define a model's behavior, persona, and safety guardrails—into a form of valuable intellectual property. This research systematically investigates the vulnerability of these prompts to "extraction" or "leakage" attacks, particularly in complex, multi-turn interactions. The study posits that while single-turn attacks are relatively well understood, the iterative nature of modern chat-based AI agents introduces a sycophancy effect, where the model's desire to be helpful or align with the user's perceived intent can be weaponized to bypass initial safety filters.¹

The methodology employed by the researchers involves a comprehensive evaluation of 10 different LLMs, spanning both proprietary closed-source models and accessible open-source architectures. The experimental design focuses on four primary functional domains to ensure the generalizability of the findings. The core innovation of the study is the development of a multi-turn threat model that elevates the average Attack Success Rate (ASR) from a baseline of 17.7% in single-query scenarios to a staggering 86.2% in multi-turn interactions. This demonstrates that persistence and iterative reframing by an adversary can dismantle the security of even highly aligned models.¹

In terms of mitigation, the authors evaluated the effectiveness of seven black-box defense strategies. These strategies range from simple prompt-based prohibitions (e.g., instructing the model to "never reveal instructions") to more complex input-output filtering mechanisms. Furthermore, the study explored the utility of fine-tuning an open-source model specifically to recognize and resist prompt leakage attempts. The results indicated that while specific defenses can reduce ASR, there is a clear trade-off between the robustness of the defense and the model's general utility and helpfulness.¹

The primary limitations of this research are centered on the inherent "cat-and-mouse" game of black-box defenses. As attackers develop more sophisticated task-reframing techniques, static prompt-based filters may become obsolete. Additionally, the study's focus on system prompts means the findings do not directly address the leakage of training data or PII, which are governed by different memorization dynamics.¹

Paper 2: Defeating Cerberus: Concept-Guided Privacy-Leakage Mitigation in Multimodal Language Models

As Large Language Models evolve into multimodal systems capable of processing both text and imagery, the attack surface for data leakage expands significantly. This research addresses the risk of Personally Identifiable Information (PII) leakage in Vision Language Models (VLMs), where sensitive data is often embedded within visual inputs such as scanned documents, ID cards, or medical records. The authors identify a critical gap in current privacy research, which has focused almost exclusively on text-only architectures.⁵

The methodology introduces a "concept-guided mitigation" approach that operates without the need for traditional retraining or computationally expensive finetuning. By applying mechanistic interpretability to the model's internal hidden states, the researchers identified specific activation directions associated with the "concept" of PII. The proposed defense involves editing the model's weights to steer these internal representations toward refusal when PII-related tasks are identified in a multimodal context. To validate this, the researchers constructed several realistic multimodal PII datasets, including scanned document variants, to simulate real-world data exfiltration scenarios.⁵

The key findings demonstrate that this steering mechanism achieves an average refusal rate of 93.3% across various PII-extraction tasks. Crucially, the mitigation was found to have a negligible impact on unrelated model performance, maintaining high accuracy on non-sensitive benchmarks such as general Document

Visual Question Answering (DocVQA). This suggests that "concept steering" can effectively isolate and suppress specific harmful behaviors within the latent space of the model without degrading its general linguistic or reasoning capabilities.⁵

However, the study is not without limitations. The evaluation was primarily conducted on open-source frameworks like LLaVA-Next, leaving the applicability of weight-space steering to massive, closed-source production models like GPT-4o or Gemini unverified. Furthermore, the reliance on synthetic repackaging of text-based PII into image formats may not fully capture the complexity of "in-the-wild" documents that feature significant OCR noise or unconventional layouts.⁵

Paper 3: Privacy-Aware Decoding: Mitigating Privacy Leakage of Large Language Models in Retrieval-Augmented Generation

Retrieval-Augmented Generation (RAG) has become the de facto standard for reducing hallucinations and providing up-to-date knowledge to LLMs. However, this architectural dependency introduces a severe privacy vulnerability: the model's response is conditioned on externally retrieved context, which may contain sensitive proprietary or private data. An adversary can craft queries to induce the model to emit these retrieved snippets verbatim. This research proposes "Privacy-Aware Decoding" (PAD) as a lightweight, inference-time solution to this problem.⁹

The PAD methodology involves the adaptive injection of calibrated Gaussian noise into the token logits during the decoding phase. Unlike static noise injection, PAD integrates confidence-based screening to identify "high-risk" tokens—those that exhibit a high logit margin and likely contain memorized or retrieved sensitive information. By modulating the noise scale based on prediction confidence and token entropy, the system balances privacy protection with linguistic quality. To provide a rigorous mathematical foundation, the authors employ a Rényi Differential Privacy (RDP) accountant to track and bound the cumulative privacy loss for each generated response.⁹

Experimental results across three real-world datasets—including Enron Mail and healthcare databases—show that PAD reduces verbatim private information leakage by 50% to 70% compared to baseline RAG systems. The defense is model-agnostic and requires no modification to the underlying retriever or the

model weights, making it highly scalable for diverse enterprise applications. The study confirms that PAD outperforms existing retrieval-phase defenses (like embedding perturbation) by targeting the generation phase where the leakage actually occurs.⁹

Significant limitations include the fact that PAD provides "fractional" DP protection, meaning that while high-risk spans are secured, an adversary might still exploit unprotected low-risk tokens to infer sensitive context through semantic similarity. Additionally, the heuristics used for sensitivity estimation—such as logit margins—may not capture every nuance of privacy risk, suggesting a need for more robust, smoothed sensitivity estimators in future iterations.¹³

Paper 4: PATCH: Mitigating PII Leakage in Language Models with Privacy-Aware Targeted Circuit Patching

This research explores the internal computational structures of transformers to address the problem of data memorization. The authors hypothesize that specific "circuits"—subgraphs of attention heads and MLP layers—are responsible for the storage and reproduction of PII. By targeting these circuits rather than the entire model, they aim to achieve a superior privacy-utility trade-off compared to global defenses like standard Differential Privacy (DP) training.¹⁴

The methodology, titled Privacy-Aware Targeted Circuit Patching (PATCH), utilizes mechanistic interpretability techniques, specifically Edge Attribution Patching with Integrated Gradients, to identify the precise internal edges that facilitate PII leakage. Once these circuits are mapped for various PII types (e.g., names, locations, email addresses), the researchers "patch" or edit the identified attention heads to suppress their influence during inference. This targeted approach allows the model to "forget" specific memorized sequences while retaining the rest of its parametric knowledge.¹⁴

The findings demonstrate that PATCH achieves a dramatic reduction in PII leakage recall, with improvements of up to 65% over baseline models. When combined with traditional DP mechanisms, the recall of residual leakage was reduced to as low as 0.01%. This suggests that the PII leakage circuits identified are persistent across different defense mechanisms, and direct editing is required for comprehensive mitigation. The research highlights that distinct attention heads are often responsible for different PII types, allowing for fine-grained control over what the model is allowed to disclose.¹⁴

The primary drawbacks are the model-specific nature of circuit discovery; the circuits identified in Llama-2-7B may not correspond to those in Mistral-7B, necessitating a custom discovery phase for every model architecture. Furthermore, the researchers caution that precision increases in non-targeted model areas

must be carefully monitored to ensure that patching PII circuits does not inadvertently degrade the model's reasoning capabilities in complex, non-sensitive tasks.¹⁴

Paper 5: Privacy-Aware Detection for Large Language Models Using a Hybrid BiLSTM-HMM Approach

Recognizing that users often inadvertently share sensitive data with LLMs, this research proposes a lightweight, real-time detection system that operates at the user-input stage. This proactive approach aims to quantify privacy risks before the data is transmitted to an LLM provider's servers, thereby addressing the root cause of many inadvertent data breaches.¹⁸

The methodology involves a hybrid pipeline that combines Bidirectional Long Short-Term Memory (BiLSTM) networks with Hidden Markov Models (HMM). The BiLSTM module is designed to capture contextual dependencies and identify sensitive information patterns within user queries. To overcome the limitation that neural networks often fail to provide rigorous probability estimates for privacy risk, the authors integrated an HMM using the Forward algorithm. This allows the system to quantify the probability that a sequence of tokens represents a privacy breach, providing actionable feedback to the user.¹⁸

The researchers reported high performance across diverse metrics, including 99.94% precision and recall on synthetic datasets generated via a novel Predefined and Sensitive Labeling (PSL) technique. On real-world datasets, the system achieved 99.99% accuracy regardless of the sentence structure. Crucially, the average execution time was recorded at 35.46 milliseconds, satisfying the strict latency requirements for practical deployment as a browser extension or API middleware.¹⁸

One significant limitation of this approach is its reliance on predefined sensitive patterns. While the BiLSTM improves upon rule-based regex systems, it may still struggle with "fuzzy" PII or information that only becomes sensitive through complex contextual inference. Additionally, the system operates as an external filter; while it protects the user, it does not mitigate the inherent vulnerabilities within the LLM itself, such as training-data memorization or prompt injection attacks that might elicit previously stored data.¹⁸

Paper 6: Investigating How Pre-training Data Leakage Affects Models' Reproduction and Detection Capabilities

This study addresses the fundamental relationship between data leakage in pre-training corpora and the subsequent behaviors of LLMs. As models are trained on massive, uncured web-crawled datasets, they inadvertently ingest personal information, copyrighted texts, and benchmark data. The researchers conduct an experimental survey to clarify how the volume of this leaked data influences both the model's reproduction of that data and our ability to detect it.²²

The methodology utilizes sampling from massive training sets and identifies leaked instances using a combination of regular expressions, web searches, and database lookups. The authors analyzed the reproduction rates of several open-source LLMs across different leakage targets. They also evaluated the effectiveness of existing detection methods, such as LOSS, PPL/zlib, and Min-K%, under varying conditions. A key focus of the study was the use of few-shot learning to enhance detection performance, providing the model with specific examples of what constitutes "leaked" versus "generated" content.²²

The experiments reveal a critical insight: LLMs frequently generate outputs containing leaked information even when the volume of such data in the training set is minimal. Furthermore, there is an inverse relationship between the quantity of leaked instances and the ease of detection—the fewer the leaked instances, the more difficult they become to identify. However, the researchers demonstrated that few-shot learning could significantly mitigate this detection bottleneck, particularly in "low-leakage" scenarios where standard zero-shot detectors often fail.²²

The primary drawbacks identified include the practical impossibility of comprehensively covering every possible leakage instance across the entire volume of training data due to resource constraints. The study was also limited to open-source models where pre-training data is publicly available, leaving the behaviors of closed-source giants like GPT-4 or Claude largely speculative in this specific context. Finally, the researchers noted that providing specific examples for supervision is critical, making high-quality labeled data a potential performance bottleneck for real-world detection systems.²²

Critical Analysis

The current landscape of data leakage mitigation for Large Language Models is characterized by a transition from heuristic-based external filtering toward intrinsic, model-aware interventions. While early efforts focused on simple input/output sanitization, the papers reviewed here demonstrate that robust security requires a deep understanding of the model's internal mechanics, specifically its tendency to memorize rare sequences.²⁴

A primary strength across these studies is the move toward formal privacy guarantees, such as the application of Rényi Differential Privacy in the PAD framework.⁹ This provides a mathematical boundary that empirical testing alone cannot offer. Furthermore, the exploration of mechanistic interpretability (e.g., PATCH and concept-guided steering) marks a significant advancement in "precision security," allowing developers to suppress specific harmful behaviors without the global utility degradation often associated with traditional DP-SGD training.⁵

However, significant flaws remain, particularly regarding the scalability and generalizability of these solutions. Many of the most effective mitigations (e.g., weight editing or circuit patching) are model-specific and computationally expensive to implement for models with billions of parameters. There is also a notable absence of unified evaluation criteria; some studies prioritize "refusal rates" while others focus on "verbatim extraction rates," making direct comparisons difficult. Moreover, the "privacy-utility trade-off" remains a persistent challenge, as aggressive noise injection or circuit suppression can inadvertently degrade the model's reasoning capabilities or creative fluency.⁷

From a methodological standpoint, the reliance on synthetic datasets or small-scale open-source models is a recurring limitation. While these allow for controlled experimentation, they may not accurately reflect the "emergent" memorization behaviors observed in massive production-scale models. Future research must bridge this gap by developing scalable, model-agnostic defenses that can handle the heterogeneous nature of data leakage in real-world, multi-modal applications.⁵

Connections

The analyzed research reveals five critical connections that unify the diverse approaches to LLM data leakage mitigation:

1. **The Centrality of Memorization:** All studies recognize that the fundamental cause of data leakage is the transformer architecture's propensity to memorize high-frequency or rare token sequences during the optimization process. Whether addressing prompt leakage, RAG context, or training data, the underlying vulnerability remains the same.²²
2. **Contextual Dependency:** Several papers (Paper 1 and 3) highlight that privacy is not a static property of the data but is highly dependent on the interaction context. Multi-turn dialogues and retrieval-augmented prompts significantly increase the risk of eliciting sensitive data compared to single-shot interactions.¹
3. **The Move Toward Inference-Time Defense:** Given the prohibitive cost of retraining massive LLMs, there is a clear consensus on the necessity of inference-time or post-training interventions. Solutions like PAD, PATCH, and concept steering all focus on modifying model behavior during the generation phase.⁵
4. **The Utility-Security Balancing Act:** A recurring theme is the careful calibration required to maintain model performance. Every mitigation strategy, from Gaussian noise to circuit editing, is evaluated against its impact on standard utility benchmarks like MT-Bench or DocVQA.⁵
5. **The Need for Sophisticated Evaluation:** Traditional regex-based PII detection is increasingly viewed as inadequate. The research points toward the need for semantic-aware evaluation, utilizing LLMs-as-judges or complex probabilistic models (like HMMs) to quantify risk.¹⁸

Conflicts between the findings

Despite the shared goals, several areas of conflict exist within the research findings:

1. **Centralized vs. Decentralized Defense:** There is a strategic disagreement between systems that mitigate leakage at the user/client level (Paper 5) and those that modify the model or decoding process at the server level (Paper 1, 3, 4). Decentralized filters offer better protection for user queries but fail to address internal model vulnerabilities.¹¹
2. **Mathematical vs. Empirical Guarantees:** A major divide exists between the use of Differential Privacy, which offers formal mathematical bounds but often degrades utility, and empirical methods like weight steering or circuit patching, which offer better utility but lack formal guarantees.⁵
3. **Weight Editing vs. Logit Modification:** Researchers are divided on whether to modify the model's parameters (Paper 2 and 4) or its output distribution (Paper 3). Parameter modification is more permanent but riskier in terms of stability, while logit modification is lightweight but may be bypassable through "clever" prompting.⁵
4. **General vs. Targeted Mitigation:** Some studies support a general approach to reducing memorization across the board, while others (like PATCH) argue that leakage is tied to specific circuits that should be surgically removed. The conflict lies in whether "privacy" can be isolated from "knowledge" within the transformer architecture.¹⁴
5. **Detection Efficacy in Low-Leakage Scenarios:** Findings in Paper 6 suggest that low leakage rates are the most difficult to detect, potentially contradicting the assumption in other papers that minor interventions are sufficient for low-risk scenarios. This implies that "rare" data might actually be the most vulnerable.²²

Synthesis Matrix for Literature Review

Study / Paper	Authors	Year	Research Design	Key Findings/Results	Methodology	Limitations
Prompt Leakage in Multi-turn LLM	Agarwal et al.	2024	Experimental	Multi-turn ASR reaches 86.2% vs 17.7% in single-	Evaluated 10 LLMs across 4 domains; tested 7 black-box	Trade-off between robustness and utility; black-box
				turn; sycophancy effect is the primary driver of leakage.	defenses; fine-tuned open-source models for refusal.	defenses can be bypassed by adaptive reframing. 1

Defeating Cerberus (Multimodal PII)	Zhang et al.	2025	Experimental	Achieved 93.3% average refusal rate for multimodal PII extraction with minimal impact on DocVQA performance.	Concept-guided mitigation via weight steering; identification of PII internal state concepts; multimodal dataset creation.	Evaluated only on open-source frameworks (LLaVA-Next); reliance on synthetic image-text composites. ⁵
Privacy-Aware Decoding (PAD)	Wang et al.	2025	Experimental	Substantial reduction in private information leakage (50-70%) while providing response-level $(\epsilon, \delta)_{\text{DP}}$ guarantees.	Adaptive Gaussian noise injection into logits; confidencebased screening; Rényi Differential Privacy accounting.	Only fractional token protection; sensitivity heuristics may not capture all latent risks. ⁹
PATCH (Circuit)	Hughes et al.	2025	Experimental	Targeted circuit editing reduced PII	Mechanistic interpretability	Discovery phase is computationally

Patching)				recall by 65%; combined with DP, recall dropped to 0.01%.	(Edge Attribution Patching); identification of PII-specific attention heads; targeted editing.	complex; circuits are architecture-specific and require custom discovery. ¹⁴
Hybrid BiLSTM-HMM Detection	Abbasalizadeh & Narain	2025	Experimental	99.99% accuracy on real-world datasets with 35.46ms latency; detects 18 types of sensitive information.	Hybrid pipeline (BiLSTM for context + HMM Forward for probability); Predefined Sensitive Labeling (PSL).	Pattern recognition dependency; does not mitigate internal model vulnerabilities like prompt injection. ¹⁸

Pre-training Data Leakage Survey	Kaneko & Baldwin	2025	Experimental Survey	LLMs generate leaked data even with minimal training instances; low leakage is harder to detect but helped by few-shot.	Sampling of training data; ID via regex and web search; evaluation of LOSS/Min-K% detectors; few-shot supervision.	Limited to opensource models with accessible data; resource intensive; data labeling is a bottleneck. ²²
---	------------------	------	---------------------	---	--	---

Detailed Technical Analysis: Theoretical Foundations and Empirical Evidence

The Mechanics of Transformer Memorization and Data Exposure

The architecture of the Large Language Model, specifically the multi-head self-attention mechanism within the Transformer, is fundamentally optimized for the compression and retrieval of information. This process, while enabling high levels of linguistic competence, inadvertently creates "parametric memory" of specific training instances. Research indicates that sequences that are highly repetitive or uniquely formatted (such as credit card numbers or system prompts) are disproportionately likely to be memorized.²⁴ The "long-tail" of data distribution in web-crawled corpora often contains sensitive PII, which the model can emit verbatim if prompted with a specific "trigger" or prefix.²²

Advanced Attack Vectors: From Single-Turn to Agentic Multi-Turn

Traditional security evaluations focused on single-shot prompt injection, where an attacker attempts to override the system instructions in a single prompt. However, as demonstrated in the research by Agarwal et al. (2024), the multi-turn interaction model significantly expands the attack surface. In these scenarios, an adversary can use the conversation history to build "trust" or context that gradually steers the model away from its safety guardrails. This "sycophancy" effect

is particularly dangerous in agentic workflows where the LLM is connected to external tools or databases, as prompt leakage can serve as a precursor to unauthorized data access or system exploitation.¹

Multimodal Expansion of Leakage Risk

The introduction of vision and audio modalities into LLMs (creating MLLMs) introduces a cross-modal leakage vector. For example, a Vision Language Model might correctly refuse a text prompt asking for a social security number but willingly extract it if presented with an image of a tax form. This is because the safety alignment often occurs primarily in the text domain. The work on "Concept-Guided Mitigation" (Zhang et al., 2025) highlights that the model's internal representations of "sensitive concepts" can be identified and suppressed across modalities, providing a unified defense strategy that is more robust than simple OCR-based text filtering.⁵

Inference-Time Probabilistic Mitigation

To provide mathematically rigorous privacy, methodologies like Privacy-Aware Decoding (PAD) utilize the principles of Differential Privacy (DP). Standard DP-SGD (Stochastic Gradient Descent with DP) is applied during the training phase, but it is notoriously difficult to scale to models with hundreds of billions of parameters due to the significant impact on gradient convergence and final model utility. PAD shifts this paradigm to the inference phase, where noise is injected not into the gradients, but into the output logits. By using Rényi Differential Privacy, researchers can accurately track the "privacy budget" consumed by each word generated, ensuring that the model never releases more information than the specified privacy policy allows.⁹

Mechanistic Interpretability: Surgical Defense

The cutting edge of LLM security lies in mechanistic interpretability—the study of how internal weights and activations map to specific behaviors. The PATCH framework (Hughes et al., 2025) exemplifies this by treating the LLM as a modular system rather than a black box. By identifying specific "attention heads" that act as "name-movers" or "PII-reproducers," developers can surgically disable or modify only those components. This minimizes the "collateral damage" to the model's performance in other areas, such as coding or creative writing, which might otherwise be degraded by broader privacy measures.¹⁴

Real-Time Input Filtering and User-Centric Protection

While internal model defenses are critical, the "last mile" of privacy is the user. The hybrid BiLSTM-HMM approach (Abbasalizadeh & Narain, 2025) provides a defense-in-depth layer that informs the user of potential risks before data ever leaves their local environment. By combining the contextual understanding of recurrent networks with the rigorous risk quantification of Hidden Markov Models, this system provides a transparent "privacy score" for every interaction. This dual approach—protecting the model and empowering the user—is increasingly seen as essential for achieving compliance with regulations like GDPR and the EU AI Act.¹⁸

Conclusion and Strategic Directions

The synthesis of contemporary research on LLM data leakage mitigation indicates that no single solution is sufficient to address the complex threat landscape. Instead, a multi-layered security architecture is required, integrating:

1. **Input-Stage Filtering:** Real-time detection systems (BiLSTM-HMM) to catch inadvertent PII sharing by users.¹⁸
2. **Inference-Time Probabilistic Defenses:** Decoding strategies like PAD that enforce formal privacy guarantees through Gaussian noise and Rényi DP.⁹
3. **Internal Structural Alignment:** Mechanistic interpretability techniques like PATCH and concept-guided steering to identify and suppress the specific internal circuits responsible for memorization and reproduction of sensitive data.⁵
4. **Agentic Guardrails:** Robust hardening of system prompts through shield-appending and multi-turn testing to protect intellectual property and prevent adversarial reconnaissance.¹

Future research must focus on the automation of these defenses, particularly the "circuit discovery" phase, to allow for scalable application across the rapidly evolving ecosystem of foundation models. Moreover, as LLMs are integrated into more complex, multi-agent systems, the study of "emergent leakage"—where non-sensitive information from multiple sources is combined to infer sensitive attributes—will become the next frontier of privacy engineering in the generative AI era.³³

Reference

- 1- Prompt Leakage effect and mitigation strategies for multi-turn LLM Applications, <https://aclanthology.org/2024.emnlp-industry.94/>
- 2- Defeating Cerberus: Concept-Guided Privacy-Leakage Mitigation in Multimodal Language Models - arXiv, <https://arxiv.org/html/2509.25525v1>
- 3- Privacy-Aware Decoding: Mitigating Privacy Leakage of Large Language Models in Retrieval-Augmented Generation -arXiv.org, <https://arxiv.org/abs/2508.03098>
- 4- PATCH: Mitigating PII Leakage in Language Models with Privacy-Aware Targeted Circuit Patching - ResearchGate, https://www.researchgate.net/publication/396374186_PATCH_Mitigating_PII_Leakage_in_Language_Models_with_Privacy-Aware_Targeted_Circuit_Patching
- 5- Privacy-Aware Detection for Large Language Models Using a Hybrid BiLSTM-HMM Approach - IEEE Xplore, <https://ieeexplore.ieee.org/iel8/6287639/6514899/11077118.pdf>
- 6- Investigating How Pre-training Data Leakage Affects Models' Reproduction and Detection Capabilities - ACL Anthology, <https://aclanthology.org/2025.emnlp-main.1201/>