PriMod4AI: Lifecycle-Aware Privacy Threat Modeling for AI Systems using LLM

Gautam Savaliya*, Robert Aufschläger, Abhishek Subedi, Michael Heigl, Martin Schramm
Deggendorf Institute of Technology, Germany
{gautam.savaliya, robert.aufschlaeger, abhishek.subedi, michael.heigl, martin.schramm}@th-deg.de

***Abstract*—Artificial intelligence systems introduce complex privacy risks throughout their lifecycle, especially when processing sensitive or high-dimensional data. Beyond the seven traditional privacy threat categories defined by the LINDDUN framework, AI systems are also exposed to model-centric privacy attacks such as membership inference and model inversion, which LINDDUN does not cover. To address both classical LINDDUN threats and additional AI-driven privacy attacks, PriMod4AI introduces a hybrid privacy threat modeling approach that unifies two structured knowledge sources, a LINDDUN knowledge base representing the established taxonomy, and a model-centric privacy attack knowledge base capturing threats outside LINDDUN. These knowledge bases are embedded into a vector database for semantic retrieval and combined with system level metadata derived from Data Flow Diagram. PriMod4AI uses retrieval-augmented and Data Flow specific prompt generation to guide large language models (LLMs) in identifying, explaining, and categorizing privacy threats across lifecycle stages. The framework produces justified and taxonomy-grounded threat assessments that integrate both classical and AI-driven perspectives. Evaluation on two AI systems indicates that PriMod4AI provides broad coverage of classical privacy categories while additionally identifying model-centric privacy threats. The framework produces consistent, knowledge-grounded outputs across LLMs, as reflected in agreement scores in the observed range.**

***Index Terms*—Threat modeling, Retrieval-augmented generation (RAG), Data flow diagrams (DFDs), Large language models (LLMs).**

I. Introduction

AI systems are increasingly deployed across domains such as healthcare, mobility, finance, and public services, where they process sensitive personal data including biometric identifiers, behavioral patterns, and contextual information. Such processing introduces substantial privacy risks, and any misuse or leakage may violate core regulatory principles such as lawfulness, data minimization, and purpose limitation defined in Article 5 of the GDPR. Traditional privacy threat modeling frameworks, such as LINDDUN (Linking, Identifying, Non-repudiation, Detecting, Data Disclosure, Unawareness, Non-compliance) [1], provide a well established taxonomy for identifying data-centric privacy risks in conventional software systems. However, they do not fully account for the dynamic, iterative, and model-driven nature of modern AI pipelines.

Beyond these, modern AI systems introduce a distinct class of model-centric privacy attacks rooted in the behavior of trained models. Such attacks exploit memorization, overfitting, or unintended information leakage from learned representations [2], [3]. For example, a face-recognition model may inadvertently memorize training images, enabling membership inference [4], or reveal sensitive attributes through model inversion [5]. Beyond these, AI models are also vulnerable to attribute inference attacks that predict hidden personal traits, training data extraction attacks [2], [6] that recover specific records from generative and encoder–decoder architectures [7], shadow-model reconstruction techniques that approximate private datasets, and embedding-space leakage where vector representations disclose identifying or sensitive information [8], [9]. These risks manifest across the entire AI lifecycle, from data collection and preprocessing to training, deployment, inference, and continuous monitoring [10] as illustrated in Figure 1.

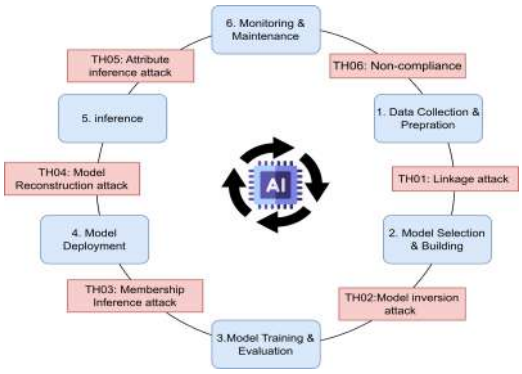

Fig. 1. A six-phase AI development lifecycle encompassing data collection, model building, training, deployment, inference, and continuous monitoring. The diagram maps distinct privacy risks (shown in red boxes) to their corresponding stages within the lifecycle.

Conventional privacy threat modeling techniques struggle to capture this dual landscape: LINDDUN-based approaches effectively identify classical privacy threats but overlook model-centric vulnerabilities, while automated extensions such as PILLAR [11] remain limited to the classical LINDDUN taxonomy and lack lifecycle awareness. This gap underscores the need for privacy threat-modeling methods that can jointly

*Gautam Savaliya is the corresponding author.

Workshop on LLM Assisted Security and Trust Exploration (LAST-X) 2026
27 February 2026, San Diego, CA, USA
ISBN 978-1-970672-05-3
https://dx.doi.org/10.14722/last-x.2026.23080
www.ndss-symposium.org

reason about both system-level and model-level privacy risks in AI systems.

The identified limitation underscores the need of lifecycle-aware methodology which are capable of addressing both traditional LINDDUN threats and emerging model-centric privacy attacks. PriMod4AI fulfills this need through a lifecycle-aware threat-identification pipeline that combines structured privacy knowledge bases with system metadata and employs LLM-driven retrieval-augmented generation to automate and explain privacy threat analysis. Motivated by this, we investigate the following research questions:

- **RQ1:** To what extent, PriMod4AI can extend traditional privacy threat-modeling approaches by jointly identifying data-centric LINDDUN threats and model-centric privacy attacks across the AI lifecycle?
- **RQ2:** How effectively can retrieval-augmented and knowledge-grounded LLM reasoning achieve reliable, consistent, and explainable privacy threat identification across multiple AI domains?

Motivated by these research questions, our work offers the following contributions:

1) **Dual structured privacy knowledge bases:** We develop two complementary knowledge sources: a LINDDUN knowledge base capturing the classical seven LINDDUN privacy threat categories, and a model-centric privacy attack knowledge base representing AI-driven threats such as membership inference, model inversion, and training data extraction.
2) **Data Flow-Specific Retrieval-Augmented Prompting:** We embed both knowledge bases into a vector database for semantic retrieval and combine the retrieved knowledge with system-level metadata derived from DFD. This enables PriMod4AI to generate context-rich, DF-specific prompts that provide grounded and lifecycle-aware threat reasoning.
3) **Explainable LLM-based threat identification:** Leveraging open-source LLMs, PriMod4AI produces structured threat assessments that integrate LINDDUN categories, AI-lifecycle stages, and explicit knowledge-source attribution, improving explainability and interpretability.

II. Related Works

A. Privacy Threat Modeling

LINDDUN is a well known established privacy threat-modeling methodology for software systems, offering a structured taxonomy of seven privacy threat categories [12]. Although several extensions, such as domain-specific refinements [13], improve its applicability, LINDDUN remains primarily design-time oriented and does not address privacy risks that emerge during model training, inference, or deployment. PLOT4AI provides an 86-threat catalog for AI technologies [14], yet its questionnaire-driven elicitation lacks lifecycle alignment and systematic mapping to system architecture. PILLAR automates LINDDUN analysis using LLMs [11], demonstrating the feasibility of LLM-assisted privacy threat modeling, but its reasoning is constrained to the classical taxonomy and does not incorporate model-centric threats or retrieval mechanisms. Together, these works establish the foundations for structured privacy analysis in software and data-processing systems.

B. AI-Specific Privacy Risks and Taxonomies

AI systems introduce privacy threats that extend beyond traditional data-flow centric frameworks. Membership inference attacks can reveal whether an individual's data was used during training [15], while model inversion techniques reconstruct sensitive attributes from model outputs [16]. Recent work further shows that large generative models can memorize and leak training data or personally identifiable information [2], underscoring risks that arise not only during data collection but also during training and deployment. Lifecycle analyses highlight the recurrence of privacy risks across evolving AI pipelines [2], [17]. Broader security oriented taxonomies such as ENISA's AI Threat Landscape and NIST's AI RMF [18], [19] catalog assets and attack surfaces, but provide limited granularity for privacy-specific risks and do not link them to LINDDUN or lifecycle stages. These findings highlight that modern AI systems introduce privacy threats that needs to be addressed.

C. Automated Threat Identification with LLMs

Recent research applies LLMs for automating security and privacy analysis. ThreatGPT [20] and ThreatModeling-LLM [21] use generative prompting and fine-tuning to automate STRIDE/NIST-based threat elicitation, while Auspex [22] and ThreatFinderAI [23] incorporate expert knowledge or knowledge graphs for asset-centric reasoning. However, these systems primarily target cybersecurity threats and lack explicit privacy taxonomies or lifecycle grounding. PILLAR [11] automates LINDDUN classification but relies solely on prompt instructions and cannot identify model-centric attacks such as inversion or membership inference. These approaches demonstrate the growing potential of LLMs for automating aspects of privacy and security assessment

D. Retrieval-Augmented Generation in Threat Modeling

Retrieval-Augmented Generation (RAG) has emerged as a powerful technique to enhance LLM-based analysis by grounding outputs in external knowledge sources, reducing hallucinations and improving factual accuracy [24]. In threat modeling, recent frameworks leverage RAG to automate elicitation processes. For instance, ThreatLens [25] integrates RAG with LLMs to generate threat models and test plans for hardware security verification, drawing from vulnerability databases. Similarly, a study shows that integrating retrieval mechanisms with generative models strengthens their ability to handle complex, information-dense queries. Such retrieval-augmented systems have been shown to provide more reliable and context-aware outputs, making them well-suited for domains where precision and up-to-date knowledge are

essential [26]. MoRSE [27], a cybersecurity chatbot, uses a mixture of RAG systems to provide comprehensive knowledge coverage across threat landscapes. These studies show that RAG can enhance domain reasoning in complex threat landscapes by grounding LLMs in structured external knowledge.

Although existing research provides valuable foundations across classical privacy taxonomies, AI-specific threat analyses, LLM-assisted modeling, and RAG-based reasoning, these efforts remain fragmented and insufficient for lifecycle-aware privacy-by-design. LINDDUN and its refinements focus on design-time, data-centric harms and do not capture threats emerging during model training or inference. AI-oriented taxonomies offer detailed accounts of model-centric attacks but lack integration with DFD semantics or privacy engineering workflows. Automated approaches such as PILLAR are the closest to our objectives, yet they remain limited to the classical LINDDUN space and cannot identify model-centric risks such as membership inference, inversion, or training-data leakage. Thus, they only partially align with the requirements of modern AI systems. Likewise, RAG-based threat modeling frameworks primarily address general cybersecurity vulnerabilities rather than privacy-specific harms, and do not incorporate structured knowledge bases or lifecycle mappings. These limitations collectively motivate PriMod4AI, which uniquely combines retrieval-augmented prompting, dual structured knowledge bases (LINDDUN and AI-specific threats), and DFD-level metadata to enable lifecycle-aware, explainable, and domain-consistent privacy threat identification across both classical and AI-specific threat spaces.

III. Proposed Method

This section outlines the methodological framework developed for automated, lifecycle-aware privacy threat modeling in AI systems. The overall architecture of the proposed framework is illustrated in Figure 2. By integrating domain knowledge and system-level representations, the framework enables contextual, LLM-driven threat reasoning. The methodology comprises five phases, beginning with Knowledge Base Construction, followed by DFD Representation of the AI System, Retrieval-Augmented Prompt Generation, and LLM Integration and Inference, and concluding with Structured Output Generation.

A. Knowledge Base Construction

The first phase of the proposed framework involves constructing two complementary knowledge bases to enable automated, context-aware privacy threat identification.

LINDDUN Knowledge Base (LINDDUN_KB): The official LINDDUN taxonomy, originally provided as a set of hierarchical PDF descriptions, was fully converted into a structured JSON-based knowledge base covering all seven LINDDUN privacy threat categories and their subnodes. Using NLP-assisted extraction followed by manual refinement, we preserved the complete hierarchy, including examples, criteria, impacts, and additional context for each threat node. Representing the taxonomy in JSON offers two key advantages: (i) it enables consistent and fine-grained retrieval of relevant threat information during prompt construction, and (ii) it allows LLMs to consume semantically structured content rather than raw PDF text, reducing ambiguity and improving grounding.

AI Model-Centric Privacy Attack Knowledge Base (AI_Privacy_KB): To extend coverage beyond traditional software privacy domains, an AI_Privacy_KB was developed through a structured, transparent literature review tailored to privacy risks unique to AI and ML. The review covered publications from 2016–2025 and examined peer-reviewed venues (*IEEE Xplore*, *ACM Digital Library*, *SpringerLink*) alongside screened preprints from *arXiv* to capture emerging threats not yet formally published. Searches used keyword combinations of *AI privacy*, *model inversion*, *membership inference*, *privacy attacks*, and *threat modeling*, with inclusion criteria requiring that a source (i) described AI-lifecycle-specific privacy risks, (ii) articulated mitigation, governance, or regulatory implications (e.g., GDPR, AI Act), and (iii) provided adequate technical detail for structured threat encoding. After filtering and deduplication, about 30 unique peer-reviewed and regulatory sources remained. Each identified threat was reviewed, mapped to its closest LINDDUN dimension, and encoded in a JSON schema capturing its `name`, `description`, `attack vector`, `AI lifecycle stage`, and `source reference`, ensuring comprehensive and taxonomy-aligned coverage. The complete list of reviewed sources and corresponding threat mappings is provided in Appendix-D.

B. DFD Representation of the AI System

To support structured privacy reasoning within the PriMod4AI pipeline, the graphical DFD of the target AI system is first transformed into comprehensive textual and semantic representation. This process begins with metadata extraction, where system documentation, component descriptions, and signal specifications are analyzed to identify functional units, their interconnections, and the type and sensitivity of data exchanged between them. The analysis produces a detailed inventory of: (i) external entities, (ii) processes and processing functions, (iii) data stores, (iv) data flows, and (v) trust boundaries delineating privacy-relevant domains.

```
"data_flows": [
  {
    "id": "DF1",
    "source": "E1",
    "destination": "P1",
    "data_type": "camera images/video",
    "sensitive_info": "visual scene data",
    "description": "Transfer of camera data to sensor
        fusion.",
    "lifecycle_stage": "Data Collection to Data
        Processing"
  }
]
```

Listing 1. JSON representation of Data Flow DF1 (Camera → Sensor Fusion), illustrating how PriMod4AI encodes data-flow metadata including source, destination, data type, sensitivity, and lifecycle stage for downstream threat-analysis prompting and knowledge retrieval.

Each data flow is then formalized into a JSON-based representation that captures its source, destination, data type,

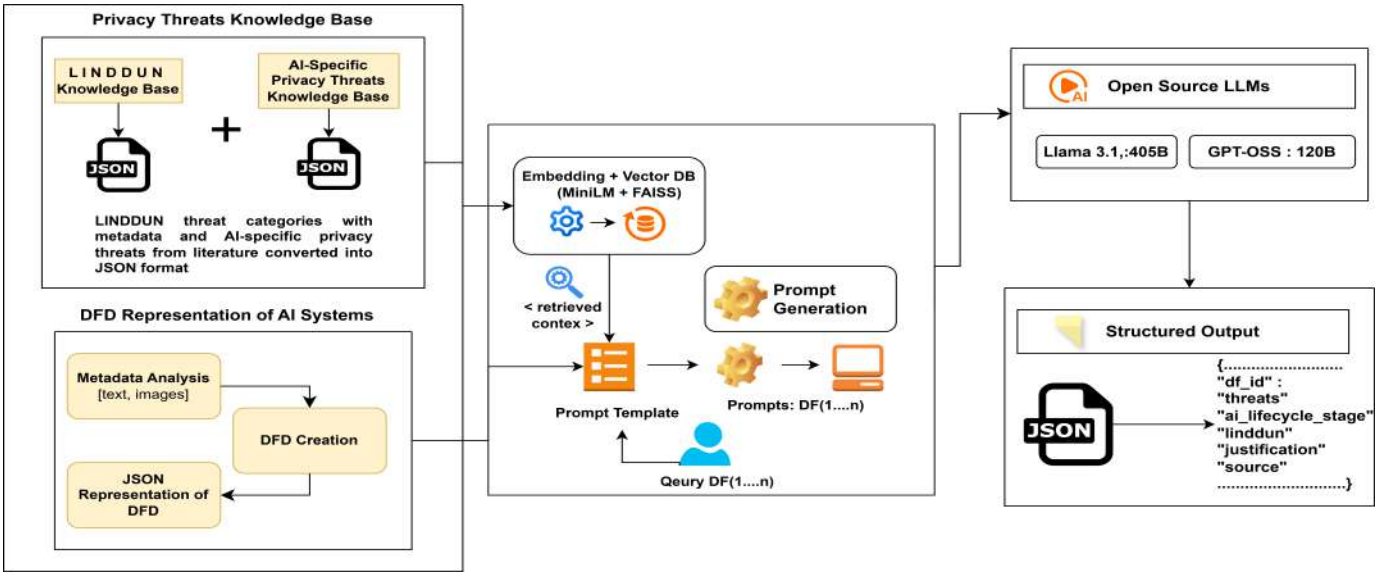

Fig. 2. Architecture of the proposed PriMod4AI framework for automated privacy threat modeling in AI systems. The framework integrates a LINDDUN + AI-specific privacy threat knowledge base, DFD representation of AI systems, and open-source LLMs for prompt-based threat identification, producing structured JSON outputs.

sensitivity classification, functional description, and associated AI lifecycle stage. An example snippet for Data Flow DF1 (Camera → Sensor Fusion), taken from the AI system used in our experimentation and illustrated in Figure 4, is provided in Listing 1. These structured representations form the basis for generating DF-specific prompts and for retrieving relevant knowledge during later stages of the PriMod4AI pipeline.

C. Retrieval-Augmented Prompt Generation

This phase generates a structured prompt for each data flow by combining the JSON-encoded DFD metadata with relevant knowledge retrieved from the LINDDUN knowledge base and AI_Privacy_KB. A base prompt template is then assembled with this information, producing DF–specific prompts that blend system context with taxonomy-grounded and model-centric threat knowledge.

a) Base Prompt Template: The framework employs a structured base prompt template that defines the LLM's analytic role, incorporates data-flow metadata, and specifies the required JSON output schema. An abbreviated version is shown in Listing 2, the full template is included in Appendix-A.

```
Analyze the following Data Flow {df_id} and identify
     privacy threats using retrieved knowledge from
    the LINDDUN_KB and AI_Privacy_KB.

### Data Flow:
{source} -> {destination}, {data_type}, {
    sensitive_info}
### Knowledge Context:
{context}
Return a JSON object with: name, justification,
    linddun_category,
ai_lifecycle_stage, and source.
```

Listing 2. Abbreviated base prompt template guiding LLM-driven privacy threat identification. The template encodes analytic instructions, data-flow metadata, and output schema constraints.

Serving as the foundation of PriMod4AI's threat-identification process, this template standardizes how system metadata and retrieved knowledge are combined, ensuring consistent, explainable, and reproducible outputs across all data flows.

b) Per-DF Prompt Construction: For each data flow (DF_i), a composite prompt (P_i) is instantiated from the base template by injecting DF-specific metadata including `id`, `source`, `destination`, `data_type`, `sensitive_info`, and `lifecycle_stage` into predefined placeholders. This ensures that each prompt is precisely aligned with the semantics of its corresponding data flow while preserving a uniform structure across the pipeline.

Formally, given a base prompt P_0 and metadata set m_i, the composite prompt is defined as:

$$P_i = f(P_0, m_i), \tag{1}$$

where f denotes the template-filling function.

After instantiation, each P_i is enriched with retrieved knowledge via RAG before being provided to the LLM. This modular separation between the static template and DF-specific instantiation improves scalability by allowing new data flows to be analyzed without redesigning prompts while also enhancing reproducibility, since each generated prompt is deterministically constructed from explicit metadata and a fixed template.

c) Retrieval-Augmented Generation: To provide factual grounding and reduce hallucination, PriMod4AI integrates a RAG pipeline that couples semantic retrieval with LLM-based reasoning. Both the LINDDUN_KB and AI_Privacy_KB are transformed into structured JSON and embedded as dense vectors using the `MiniLM-L6-v2` [28] model from the Sentence Transformers family. The resulting embeddings are indexed

using FAISS [29] for efficient nearest-neighbor retrieval. To maintain concept-level granularity, the knowledge bases are segmented via a recursive text-splitting strategy, ensuring that each node of the taxonomy remains independently retrievable.

For each instantiated prompt P_i, the textual description of the corresponding data flow denoted as d_i and obtained directly from the DFD metadata, is encoded into a query vector q_i. The retriever then selects the top-k ($k = 7$) most semantically relevant knowledge fragments:

$$S_i = \text{top-}k(R(q_i, K)), \tag{2}$$

where K denotes the embedded knowledge corpus and R is the retrieval function.

The retrieved fragment set S_i is inserted into the prompt's <context> section, after which the remaining instruction block of the base template is appended. Formally, the augmented prompt is expressed as:

$$\tilde{P}_i = \text{assemble}(P_i, S_i), \tag{3}$$

where assemble denotes the template-aware integration of (i) DF-specific metadata, (ii) retrieved knowledge, and (iii) task instructions.

D. Open-Source LLM Integration

The next stage of the framework processes retrieval-enhanced prompts using open-source LLMs for automated privacy threat identification. To ensure transparency and reproducibility, only openly available models were employed via the OllamaLLM interface. Two variants were used: GPT-OSS (120B)[1] optimized for structured reasoning, and LLaMA 3.1 (405B)[2]. offering enhanced contextual precision. Each augmented prompt $\tilde{P}_i$, containing system metadata and retrieved knowledge fragments, is supplied to the selected model for inference in schema-constrained JSON mode, ensuring a uniform output structure (df_id, identified_threats). Decoding parameters (e.g., temperature, top-p) are fixed across models to maintain comparability. This setup supports interchangeable inference and cross-model evaluation of reasoning consistency, coverage, and reproducibility.

E. Structured Output Generation

The final stage produces a structured output summarizing all identified privacy threats per DF. For each DF, the LLM returns a standardized JSON record with threat names, LINDDUN categories, AI lifecycle stages, and source references. As shown in Listing 3, this uniform format supports consistency, evaluation, and automated visualization, while enabling comparison and reproducibility across DFs. this uniform format supports consistency, evaluation, and automated visualization, while enabling comparison and reproducibility across DFs.

[1]Ollama, "gpt-oss," Ollama Library. [Online]. Available: https://ollama.com/library/gpt-oss. [Accessed: Dec. 3, 2025].

[2]AI at Meta, Jul. 23, 2024. [Online]. Available: https://ai.meta.com/blog/meta-llama-3-1/. [Accessed: Dec. 3, 2025].

```
{
"df_id": "DF5",
"identified_threats":[ {
   "name": "Unencrypted Data Transfer",
   "justification": "Sensitive biometric data
       may be exposed if transferred without
       encryption.",
   "linddun_category": "Disclosure of
       information",
   "ai_lifecycle_stage": "Inference/Storage",
   "source": "LINDDUN" },
   {
   "name": "Model Inversion Attack",
   "justification": "Stored embeddings could
       be exploited to reconstruct facial
       traits.",
   "linddun_category": "Disclosure of
       information",
   "ai_lifecycle_stage": "Inference",
   "source": "AI_PRIVACY_KB" }]}
```

Listing 3. The structured JSON output for Data Flow 5 (DF5), derived from Figure 3, is generated by LLaMA 3.1 within PRIMOD4AI and applies the unified threat-encoding schema to an internal AI system flow.

F. Algorithmic Pipeline for Threat Identification

Algorithm 1 PriMod4AI: Retrieval-Augmented LLM-Based Privacy Threat Identification

Require: LINDDUN_KB $\mathcal{K}_L$, AI_Privacy_KB $\mathcal{K}_A$, DFD JSON $\mathcal{D}$, base prompt template $\mathcal{P}$; embedding model M_{emb}; FAISS index $\mathcal{V}$ (initially empty); LLM M_{LLM}; retriever top-k (here $k = 7$).

Ensure: Result list $\mathcal{R}$ of zero or more JSON objects of the form {"df_id": ..., "identified_threats": [...]}.

$\mathcal{R} \leftarrow [\,]$ {Initialize result list}
Build knowledge index:
Split $\mathcal{K}_L$ and $\mathcal{K}_A$ into concept-level text chunks.
for each chunk c in $\mathcal{K}_L \cup \mathcal{K}_A$ **do**
  $e \leftarrow M_{\text{emb}}(c)$
  Insert e into FAISS index $\mathcal{V}$
end for
Load inputs:
Parse $\mathcal{D}$ into the set of data flows $\{DF_j\}$ and load the base prompt template $\mathcal{P}$.
for each data flow DF_j in $\{DF_j\}$ **do**
  Extract DF metadata m_j (source, destination, data_type, sensitive_info, lifecycle_stage) and textual description d_j.
  Construct query vector $Q_j \leftarrow M_{\text{emb}}(d_j)$.
  Retrieve relevant fragments: $S_j \leftarrow \text{top-}k(R(Q_j, \mathcal{V}))$.
  Form composite prompt $X_j \leftarrow \text{assemble}(\mathcal{P}, m_j, S_j)$.
  Generate JSON output $Y_j \leftarrow M_{\text{LLM}}(X_j)$.
  Validate Y_j against the expected schema; optionally apply repair or re-prompting if invalid.
  Append Y_j to $\mathcal{R}$.
end for
Store results: Write $\mathcal{R}$ to outputs.

IV. Experimental Setup

This section describes the implementation setup and experimental procedure used to validate PriMod4AI. Two cross-domain AI-based systems, a Face Authentication System and an Autonomous Driving System were selected due to their distinct privacy sensitivities and heterogeneous data modalities. The implementation outlines how PriMod4AI was executed end-to-end, detailing the experimental environment and computational resources used during inference. Table I summarizes the complete environment configuration employed across all experiments.

TABLE I
HARDWARE AND SOFTWARE CONFIGURATION USED FOR IMPLEMENTING AND EXECUTING PRIMOD4AI.

Component	Specification / Description
Host System	Windows 11; Virtual Machine
CPU	AMD Ryzen 5 5625U (VM configuration); 16 GB RAM
GPU Hardware	NVIDIA A100 80 GB PCIe (MIG profile: 3g.40 GB)
Environment	Python 3.11; `OllamaLLM` inference interface
Model 1	LLaMA 3.1 (405B), open-source dense decoder model
Model 2	GPT-OSS (120B), open-source LLM for structured reasoning
Decoding Parameters	Temperature = 0.7; top-p = 0.9; max tokens = 1024

A. Use Cases

1) AI-based Face Authentication System: The first system represents a typical biometric verification pipeline that processes facial images, embeddings, and identity records. The dataset and reference architecture were adapted from the open-source PILLAR repository[3] ensuring compatibility with standard LINDDUN threat definitions. The DFD of the system is shown in Figure 3.

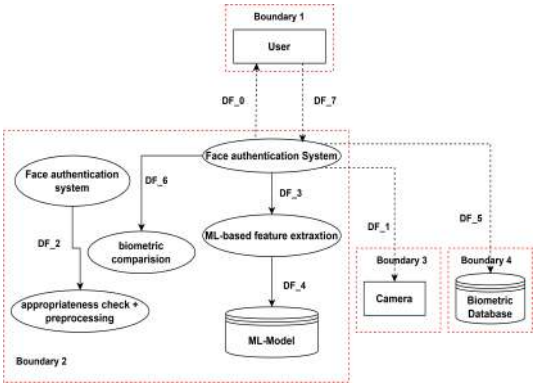

Fig. 3. DFD of the AI-based Face Authentication System, adapted from the open-source PILLAR repository

The architecture consists of a camera (data source), preprocessing and feature extraction modules, a biometric comparison component, and storage units including databases and ML models. Each data flow (DF1–DF7) was analyzed to capture its source, destination, data type, sensitivity, and corresponding AI lifecycle stage.

2) Autonomous Driving System: The second system represents a generalized autonomous driving architecture derived from the government-funded JUST BETTER DATA (JBData) research project[4]. Due to the scale and complexity of the original system, and to ensure confidentiality, the architecture has been abstracted and generalized for experimental use. While specific implementation details are omitted, the core structural characteristics, sensor modalities, and representative data flows are preserved to maintain realism and suitability for privacy analysis.As shown in Figure 4, the system integrates multi-sensor perception, fusion modules, planning and control components, and cloud-assisted synchronization processes. The architecture comprises fourteen data flows (DF1–DF14) spanning multiple AI lifecycle stages, providing a realistic and comprehensive testbed for evaluating lifecycle-aware privacy threat identification methods.

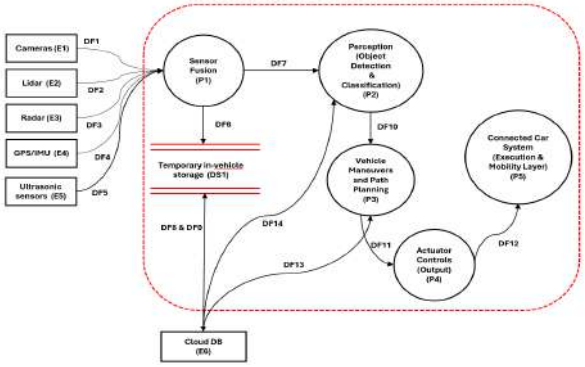

Fig. 4. DFD of the autonomous driving system showing key processes, data stores, and data flows (DF1–DF14).

V. Evaluation

The evaluation assesses the performance and reliability of PriMod4AI using a two-layer structure, summarized in Table II. The layered design reflects the dual nature of PriMod4AI's outputs, which include both classical LINDDUN threats and AI-driven, model-centric privacy attacks.

A. Layer A: Classical LINDDUN Threat Identification (Comparison with PILLAR)

In this evaluation layer, we assess PriMod4AI, implemented using both the GPT-OSS and LLaMA 3.1 models within the classical LINDDUN privacy-threat space for both examined systems. Traditional LINDDUN modeling defines seven canonical threat categories (Linkability, Identifiability, Non-repudiation, Detectability, Disclosure of Information, Unawareness, and Non-compliance). To understand how PriMod4AI aligns with this taxonomy, we compare its category-

[3]PILLAR: LINDDUN Privacy Threat Modeling Using LLMs. [Online]. Available: https://github.com/stfbk/PILLAR (Accessed: Nov. 5, 2025).

[4]just better DATA: Effiziente und hochgenaue Datenerzeugung für KI-Anwendungen im Bereich autonomes Fahren. [Online]. Available: https://www.justbetterdata.de/konzept/ (Accessed: Nov. 5, 2025).

TABLE II
OVERVIEW OF EVALUATION LAYERS AND ASSOCIATED METRICS.

Layer	Metrics Used
Layer A: Classical LINDDUN Space	1) Category Coverage 2) PILLAR-Recall 3) Jaccard Similarity
Layer B: Cross-Model Agreement (LINDDUN + AI)	1) Cohen's κ • P_o • PABAK 2) Robustness Coefficient

level outputs against those produced by PILLAR, an LLM-based tool that automates the original LINDDUN workflow. PILLAR is not treated as a ground-truth baseline, but rather as a point of comparison for classical LINDDUN reasoning. The tool systematically generates analysis across all seven LINDDUN categories for each data flow for LINDDUN Pro method.

a) Experimental Setup for PILLAR: We executed PILLAR using its official Streamlit web application[5] with the GPT-4 Turbo model (OpenAI API), temperature 0.7. The application requires structured input describing the system, data types, data transformations, actors, and data governance properties. For questions relating to data retention and deletion, we used a standardized organizational data policy. PILLAR then generated threats for all seven LINDDUN categories for each data flow (DF), after which a human analyst performed the relevance filtering step described above. PriMod4AI (LLaMA 3.1) and PriMod4AI (GPT-OSS) were evaluated against the filtered PILLAR results using the classical LINDDUN framework and the following metrics:

- **Category Coverage (%) [30]:**
  Measures how many of the seven LINDDUN categories are detected by a threat-identification model m (PILLAR or PriMod4AI) for a specific data flow d in the system's DFD.
  Let $\mathcal{L} = \{\ell_1, \ldots, \ell_7\}$ denote the LINDDUN category set. The coverage is defined as:
  $$\mathrm{Cov}_m^{(d)} = \frac{\sum_{\ell \in \mathcal{L}} \mathbf{1}_{\text{threat}}(\ell, m, d)}{|\mathcal{L}|}. \tag{4}$$
  Here, $\mathbf{1}_{\text{threat}}(\ell, m, d)$ equals 1 if model m identifies at least one threat in category ℓ for data flow d, and 0 otherwise.
- **PILLAR-Recall (Category-Based) [31], [32]:** Assesses backward compatibility by computing the fraction of (data-flow, category) pairs identified by PILLAR that are also identified by PriMod4AI. A recall of 1.0 indicates perfect reproduction of PILLAR's LINDDUN coverage.
- **Jaccard Similarity (Per DF) [33]:**
  Measures set-similarity between PriMod4AI and PILLAR on a per-data-flow basis:
  $$J_m(d) = \frac{|C_m(d) \cap C_{\text{PILLAR}}(d)|}{|C_m(d) \cup C_{\text{PILLAR}}(d)|}. \tag{5}$$
  We report both per-DF scores and the average across all DFs.

[5]https://pillar-ptm.streamlit.app/, (Accessed: Nov. 5, 2025)

B. Layer B: Overall Privacy Threat Space and Cross-Model Agreement Analysis

PriMod4AI identifies both classical LINDDUN threats and AI-driven, model-centric privacy attacks, forming an extended privacy threat space for which no benchmarks or expert-annotated ground truth exist. As direct accuracy-based evaluation is therefore infeasible. Layer B assesses the reliability and factual consistency of PriMod4AI's outputs by measuring the agreement between two LLMs (GPT-OSS and LLaMA3.1) across both evaluation systems and across all identified threat types. To ensure consistent agreement computation across models, threat names extracted from the structured JSON outputs are normalized using a preprocessing pipeline comprising: (i) lowercasing and lemmatization, (ii) stopword and punctuation removal, and (iii) token-set normalization. After preprocessing, semantically similar labels are merged using token-set Jaccard similarity, clustering two threat names when their similarity exceeds $\tau = 0.20$, a commonly used threshold balancing precision and recall for near-duplicate textual concepts.

Three complementary statistical measures are then applied:

1) **Observed Agreement (P_o) [34]:** Represents the raw proportion of semantic threat clusters on which both models (GPT-OSS and LLaMA3.1) agree, either by jointly predicting the presence or the absence of a threat. It serves as the foundational quantity from which the other agreement metrics are derived.
2) **Cohen's κ** [34]: A chance-corrected measure of agreement that quantifies how consistently the two models identify the same semantic threat clusters beyond what would be expected by random coincidence. For observed agreement P_o and expected agreement P_e, Cohen's κ is defined as:
   $$\kappa = \frac{P_o - P_e}{1 - P_e}. \tag{6}$$
   Values above 0.75 are typically interpreted as indicating substantial to near-perfect agreement.
3) **PABAK (Prevalence-Adjusted Bias-Adjusted Kappa)** [35]: A modified form of Cohen's κ designed for imbalanced binary data, where most threat clusters are absent. To correct the instability of κ under low-prevalence conditions, PABAK adjusts the estimate using only observed agreement:
   $$\text{PABAK} = 2P_o - 1. \tag{7}$$
   This yields a more robust agreement measure in sparse threat-identification settings.

VI. Results and Discussion

A. Layer A: Classical LINDDUN Threat Space

Layer A evaluates PriMod4AI within the classical LINDDUN threat space by comparing its outputs to the PILLAR output as shown in Table III. Across both AI systems, PriMod4AI achieves moderate and acceptable agreement with PILLAR, as reflected by its PILLAR-Recall scores and Jaccard overlaps. The GPT-OSS variant consistently shows the closest alignment with PILLAR, achieving higher recall and broader category coverage, whereas the LLaMA3.1 variant provides a more compact and selective set of categories, resulting in slightly lower coverage but still maintaining reasonable overlap. The results also suggest that both PriMod4AI variants remain stable across heterogeneous system architectures, including the more complex multi-flow Autonomous Driving system. Overall, Layer A shows that PriMod4AI remains compatible with classical LINDDUN reasoning while enabling structured, lifecycle-aware threat assessment. Appendix -B details per-flow LINDDUN analyses, reporting category coverage and identified threats for PILLAR, PriMod4AI (GPT-OSS), and PriMod4AI (LLaMA 3.1).

B. Layer B: Combined Threat Space and Model-Centric Analysis

Layer B evaluates the consistency of PriMod4AI across different LLM variants by assessing cross-model agreement within the combined (LINDDUN + AI-specific) threat space. Since no ground-truth dataset exists for this domain, reliability is examined through inter-model metrics reported in Table III. Across both systems, the agreement scores between PriMod4AI (GPT-OSS) and PriMod4AI (LLaMA3.1) fall within a moderate to substantial range, indicating that the two models produce broadly comparable threat sets despite differences in wording or granularity. The observed agreement (P_o) and PABAK values suggest that both variants follow similar decision patterns rather than diverging arbitrarily, while Cohen's κ show that the overlap between their outputs is meaningful but not identical. To summarize stability across system architectures, the normalized robustness coefficient R indicates that PriMod4AI achieves moderate and consistent cross-model reliability, with $R = 0.7018$ for the face authentication system and $R = 0.6117$ for the autonomous driving system, reflecting stable but not exceptionally high robustness.Model-centric privacy risks are detailed in Appendix -C, including their threat families and canonical categorization.

VII. Conclusion and Future Work

This work introduced PriMod4AI, a lifecycle-aware privacy threat-modeling framework that unifies structured privacy knowledge with retrieval-augmented LLM reasoning for automated, explainable privacy risk analysis in AI systems.

RQ1 examined whether PriMod4AI can extend traditional privacy threat modeling by identifying both LINDDUN threats and AI-specific, model-centric risks. The results indicate that PriMod4AI maintains compatibility with classical LINDDUN outputs while additionally detecting a range of AI-driven threats not captured by PILLAR. This suggests that the framework can broaden the assessed privacy threat space in a structured manner.

RQ2 investigated whether retrieval-augmented and knowledge-grounded prompting supports consistent and reliable threat identification. The cross-model agreement metrics reported in our evaluation show moderate to substantial alignment between the GPT-OSS and LLaMA3.1 variants, indicating that PriMod4AI produces generally stable and repeatable threat assessments across heterogeneous system architectures.

These findings demonstrate that structured knowledge integration and RAG-based prompting provide a practical way to enhance coverage and reproducibility in privacy threat modeling for AI systems.PriMod4AI thus demonstrates that structured knowledge integration and RAG-based prompting can significantly enhance the rigor, coverage, and reproducibility of privacy threat modeling for the AI systems. Future work will focus on several practical extensions. First, the current evaluation is limited by the absence of expert-validated ground truth for AI-driven, model centric privacy threats so incorporating expert feedback loops could strengthen external validity. Second, the static knowledge bases may be expanded into lightweight, updateable knowledge repository to better track emerging privacy risks and regulatory changes. Third, integrating simple mitigation guidance for the identified threats would improve the framework's usability in real development settings. Additionally, extending the pipeline to support incremental updates or domain-specific modules may further enhance adaptability across diverse AI architectures.

Acknowledgement

The research leading to these results is funded by the German Federal Ministry for Economic Affairs and Energy within the project "just better DATA - Effiziente und hochgenaue Datenerzeugung für KI-Anwendungen im Bereich autonomes Fahren". The authors would like to thank the consortium for the successful cooperation.

References

[1] K. Wuyts, L. Sion, and W. Joosen, "LINDDUN GO: A Lightweight Approach to Privacy Threat Modeling," in Proc. 2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), Genoa, Italy, Sept. 2020, pp. 302–309, doi: 10.1109/EuroSPW51379.2020.00047.

[2] N. Carlini, F. Tramèr, E. Wallace, M. Jagielski, A. Herbert-Voss, K. Lee, A. Roberts, T. Brown, D. Song, Ú. Erlingsson, A. Oprea, and C. Raffel, "Extracting Training Data from Large Language Models," in Proc. 30th USENIX Security Symposium (USENIX Security 21), Aug. 2021, pp. 2633–2650.

[3] Y. Liu, J. Huang, Y. Li, et al., "Generative AI model privacy: a survey," Artificial Intelligence Review, vol. 58, p. 33, 2025. [Online]. Available: https://doi.org/10.1007/s10462-024-11024-6

[4] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in Proc. 2017 IEEE Symp. Security and Privacy (SP), San Jose, CA, USA, May 2017, pp. 3–18, doi: 10.1109/SP.2017.41.

[5] Y. Zhang, R. Jia, H. Pei, W. Wang, B. Li, and D. Song, "The secret revealer: Generative model-inversion attacks against deep neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 253–261.

TABLE III

SUMMARY OF EVALUATION RESULTS ACROSS THE TWO ANALYTICAL LAYERS: LAYER A (CLASSICAL LINDDUN THREAT IDENTIFICATION, COMPARED WITH PILLAR) AND LAYER B (OVERALL PRIVACY THREAT SPACE AND CROSS-MODEL AGREEMENT ANALYSIS). EACH SECTION INCLUDES AN INTERPRETATION COLUMN HIGHLIGHTING THE PRIMARY OBSERVATIONS.

Layer A: Classical LINDDUN Threat Identification (Comparison with PILLAR)					
System	**Model**	**PILLAR-Recall**	**Avg. Coverage**	**Avg. Jaccard**	**Interpretation**
Face Authentication	PILLAR	–	71.4%	–	Reference tool providing rule-based LINDDUN outputs used for comparison after analyst filtering.
	PriMod4AI (GPT-OSS)	85.0%	82.1%	0.642	High alignment with PILLAR and broad category coverage, indicating strong classical LINDDUN consistency.
	PriMod4AI (LLaMA)	77.5%	73.2%	0.617	Moderate overlap with PILLAR with more selective category usage, reflecting a more compact reasoning style.
Autonomous Driving	PILLAR	–	82.6%	–	Produces full-category threat suggestions requiring analyst filtering; used as comparative reference.
	PriMod4AI (GPT-OSS)	85.2%	81.1%	0.726	Strong agreement with PILLAR and stable category coverage across complex multi-flow system.
	PriMod4AI (LLaMA)	80.2%	71.3%	0.686	Good overlap with PILLAR while exhibiting greater divergence in category selection across flows.
Layer B: Overall Privacy Threat Space and Cross-Model Agreement Analysis					
System	**Model Pair**	κ	P_o	**PABAK**	**Interpretation**
Face Authentication	GPT-OSS ↔ LLaMA	0.7455	0.7818	0.5782	Moderate agreement levels, indicating generally consistent and largely non-hallucinatory reasoning..
Autonomous Driving	GPT-OSS ↔ LLaMA	0.69	0.715	0.43	Moderate cross-model consistency across diverse system architectures.

[6] N. Carlini, J. Hayes, M. Nasr, M. Jagielski, V. Sehwag, F. Tramèr, B. Balle, D. Ippolito, and E. Wallace, "Extracting Training Data from Diffusion Models," in 32nd USENIX Security Symposium (USENIX Security 23), Anaheim, CA, 2023, pp. 5253–5270.

[7] B. Hilprecht, M. Härterich, and D. Bernau, "Monte carlo and reconstruction membership inference attacks against generative models," *Proceedings on Privacy Enhancing Technologies*, 2019.

[8] T. Miura, T. Shibahara, and N. Yanai, "Megex: Data-free model extraction attack against gradient-based explainable ai," in *Proceedings of the 2nd ACM Workshop on Secure and Trustworthy Deep Learning Systems*, 2024, pp. 56–66.

[9] J. Niu, P. Liu, X. Zhu, K. Shen, Y. Wang, H. Chi, Y. Shen, X. Jiang, J. Ma, and Y. Zhang, "A survey on membership inference attacks and defenses in machine learning," *Journal of Information and Intelligence*, vol. 2, no. 5, pp. 404–454, 2024.

[10] S. Shahriar, S. Allana, S. M. Hazratifard, and R. Dara, "A Survey of Privacy Risks and Mitigation Strategies in the Artificial Intelligence Life Cycle," IEEE Access, vol. 11, pp. 61829–61854, 2023. [Online]. Available: https://doi.org/10.1109/ACCESS.2023.3287195

[11] M. Mollaeefar, A. Bissoli, D. Van Landuyt, and S. Ranise, "PILLAR: LINDDUN Privacy Threat Modeling Using LLMs," in *Proceedings of the 2025 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW),* 2025, pp. 278–286, doi: 10.1109/EuroSPW67616.2025.00038.

[12] M. Deng, K. Wuyts, R. Scandariato, B. Preneel, and W. Joosen, "A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements," Requirements Engineering, vol. 16, no. 1, pp. 3–32, Mar. 2011, doi: 10.1007/s00766-010-0115-7.

[13] K. Wuyts, D. Van Landuyt, A. Hovsepyan, and W. Joosen, "Effective and efficient privacy threat modeling through domain refinements," in Proc. 33rd Annu. ACM Symp. Appl. Comput. (SAC), Pau, France, Apr. 2018, pp. 1175–1178, doi: 10.1145/3167132.3167414.

[14] I. Barberá, "PLOT4AI: Practical Library Of Threats 4 Artificial Intelligence," PLOT4AI, 2025. [Online]. Available: https://plot4.ai/library ([plot4.ai](https://plot4.ai/library))

[15] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in Proc. 2017 IEEE Symp. Security and Privacy (SP), San Jose, CA, USA, May 2017, pp. 3–18, doi: 10.1109/SP.2017.41.

[16] Z. Zhou, J. Zhu, F. Yu, X. Li, X. Peng, T. Liu, and B. Han, "Model inversion attacks: A survey of approaches and countermeasures," arXiv preprint arXiv:2411.10023, 2024. [Online]. Available: https://arxiv.org/abs/2411.10023

[17] K. Chen, X. Zhou, Y. Lin, S. Feng, L. Shen, and P. Wu, "A Survey on Privacy Risks and Protection in Large Language Models," arXiv preprint arXiv:2505.01976, 2025. [Online]. Available: https://arxiv.org/abs/2505.01976

[18] E. Tabassi, Artificial Intelligence Risk Management Framework (AI RMF 1.0), NIST AI 100-1, National Institute of Standards and Technology, Gaithersburg, MD, Jan. 2023. [Online]. Available: hhttps://doi.org/10.6028/NIST.AI.100-1

[19] European Union Agency for Cybersecurity (ENISA), "Artificial Intelligence Cybersecurity Challenges," ENISA, Nov. 2023. [Online]. Available: https://www.enisa.europa.eu/publications/artificial-intelligence-cybersecurity-challenges

[20] M. Gupta, C. Akiri, K. Aryal, E. Parker, and L. Praharaj, "From ChatGPT to ThreatGPT: Impact of Generative AI in Cybersecurity and Privacy," IEEE Access, vol. 11, pp. 80218–80245, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:259316122

[21] T. Wu, S. Yang, S. Liu, D. Nguyen, S. Jang, and A. Abuadbba, "ThreatModeling-LLM: Automating Threat Modeling using Large Language Models for Banking System," *arXiv e-prints*, 2024, p. arXiv–2411.

[22] A. Crossman, A. R. Plummer, C. Sekharudu, D. Warrier, and M. Yekrangian, "Auspex: Building threat modeling tradecraft into an artificial intelligence-based copilot," *arXiv e-prints*, 2025, p. arXiv–2503.

[23] J. Von der Assen, A. Huertas, J. Sharif, C. Feng, G. Bovet, and B. Stiller, "ThreatFinderAI: Automated Threat Modeling Applied to LLM System Integration," in Proceedings of the 2024 20th International Conference on Network and Service Management (CNSM), 2024, pp. 1–3, doi: 10.23919/CNSM62983.2024.10814632.

[24] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive NLP tasks," in

Proceedings of NeurIPS 2020, Red Hook, NY, USA: Curran Associates Inc., 2020, Art. no. 793, pp. 1–16.*

[25] D. Saha, H. Al Shaikh, S. Tarek, and F. Farahmandi, "Special session: ThreatLens: LLM-guided threat modeling and test plan generation for hardware security verification," in *2025 IEEE 43rd VLSI Test Symposium (VTS)*, 2025, pp. 1–5.

[26] I. Elsharef, Z. Zeng, and Z. Gu, "Facilitating threat modeling by leveraging large language models," in *Proceedings of the Workshop on AI Systems with Confidential Computing*, San Diego, CA, USA, Feb. 2024.

[27] M. Simoni, A. Saracino, V. P, and M. Conti, "MoRSE: Bridging the gap in cybersecurity expertise with retrieval augmented generation," in *Proceedings of the 40th ACM/SIGAPP Symposium on Applied Computing*, 2025, pp. 1213–1222.

[28] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, and M. Zhou, "Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers," *Advances in neural information processing systems*, vol. 33, pp. 5776–5788, 2020.

[29] J. Johnson, M. Douze, and H. Jégou, "Billion-Scale Similarity Search with GPUs," *IEEE Transactions on Big Data, vol. 7, no. 3, pp. 535–547, 2021.* [Online]. Available: https://doi.org/10.1109/TBDATA.2019.2921572

[30] W. Xiong and R. Lagerström, "Threat modeling – A systematic literature review," *Computers & Security*, vol. 84, pp. 53–69, Jul. 2019, doi: 10.1016/j.cose.2019.03.010.

[31] C. D. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval. USA: Cambridge University Press, 2008. [Online]. Available: https://www.cambridge.org/core/books/introduction-to-information-retrieval/5980753ED7E3036C8B83D4321F55C09B

[32] C. C. Aggarwal and C. Zhai, "A Survey of Text Classification Algorithms," in *Mining Text Data*, C. Aggarwal and C. Zhai, Eds. Boston, MA: Springer, 2012, pp. 163–222.

[33] S. Fletcher and M. Z. Islam, "Comparing sets of patterns with the Jaccard index," Australasian Journal of Information Systems, vol. 22, 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:54807073

[34] J. Cohen, "A Coefficient of Agreement for Nominal Scales," Educational and Psychological Measurement, vol. 20, pp. 37–46, 1960. [Online]. Available: https://api.semanticscholar.org/CorpusID:15926286

[35] T. Byrt, J. Bishop, and J. B. Carlin, "Bias, prevalence and kappa," Journal of Clinical Epidemiology, vol. 46, no. 5, pp. 423–429, 1993. [Online]. Available: https://doi.org/10.1016/0895-4356(93)90018-V

APPENDIX

A. Base Prompt Template

The PriMod4AI framework relies on a structured and retrieval-augmented prompt design that standardizes how LLMs perform privacy threat analysis, as reflected in the complete base prompt template shown in Listing 4, ensuring uniform incorporation of retrieved context, lifecycle stages, and DFD attributes, thereby enabling systematic and well-grounded threat identification across AI systems.

The prompt template is intentionally modular and role-based, enabling the LLM to combine system-level metadata, lifecycle context, and domain knowledge from both the LINDDUN taxonomy and the AI-specific privacy threat knowledge base. The template enforces a controlled reasoning process by (i) explicitly describing the data flow under analysis, (ii) injecting only the relevant subset of the knowledge bases retrieved for that flow, The template also operationalizes dual-mode reasoning. First, it directs the LLM to apply the seven LINDDUN categories (Linkability, Identifiability, Non-repudiation, Detectability, Disclosure of Information, Unawareness, and Non-compliance) to the given data flow. Second, it supplements this with AI-specific, model-centric threat reasoning (e.g., membership inference, model inversion, model extraction, training-data leakage) sourced from the AI Privacy KB. When an AI threat does not directly map onto a LINDDUN category, the template requires the model to provide a justified mapping, thereby aligning emerging AI privacy risks with classical taxonomic structures.

```
You are an expert privacy threat analyst
    specializing in AI systems. Your task is to
    analyze the provided Data Flow (DF) and identify
     all relevant privacy threats.
### Data Flow Context
- ID: {df_id}
- Source: {source}
- Destination: {destination}
- Data Type: {data_type}
- Sensitive: {sensitive}
- Sensitive Info: {sensitive_info}
- AI Lifecycle Stage: {lifecycle_stage}
### Knowledge Base (retrieved context)
{context}
### Threat Identification Rules (IMPORTANT)
1. Use BOTH:
  - your own expertise in privacy, LINDDUN, and AI-
      specific threats, AND
  - the retrieved context text above.
  The context is guidance, not a mandatory
      constraint.
2. Identify only those threats that are **logically
    possible** for THIS DF,
  based strictly on:
  - source and destination,
  - data_type,
  - sensitivity,
  - sensitive_info,
  - lifecycle_stage.
  Do NOT invent threats or include threats that
      contradict the DF.
3. LINDDUN threats:
  - Include a LINDDUN threat ONLY if it genuinely
      applies.
  - If `sensitive` = true, at least one LINDDUN
      threat is expected unless the DF truly poses
      no risk.
4. AI-specific threats:
  Include an AI-specific threat **only if this DF
      involves. Do NOT include an AI-specific threat
       if none logically apply.
5. Avoid repeating identical threats across DFs
    unless the same threat clearly applies
  to the same scenario.
6. For each identified threat, include:
  - "name": a clear and specific threat name.
  - "justification": explain why this threat applies
      , referencing DF-specific fields
    (source, destination, data_type, sensitive_info,
        lifecycle_stage).
  - "linddun_category": LINDDUN category.
  - "ai_lifecycle_stage": the lifecycle stage where
      the threat occurs
  - "source": "LINDDUN" for traditional privacy
      threats,
          "AI_PRIVACY_KB" for AI-specific threats.
```

Listing 4. Base prompt template used for automated privacy threat identification. The prompt integrates LINDDUN knowledge, AI-specific privacy risks, and structured system metadata to elicit grounded and machine-readable reasoning.

By integrating structured retrieval, explicit lifecycle grounding, and strict output constraints, this template enables reproducible, explainable, and cross-model-consistent privacy threat identification. It prevents unconstrained generative behavior, reduces hallucinations, and ensures that every reported threat is explicitly tied to a knowledge source and a lifecycle stage.

It prevents unconstrained generative behavior, reduces hallucinations, and ensures that every reported threat is explicitly tied to a knowledge source and a lifecycle stage.

B. Layer A: Classical LINDDUN Threat Space (LINDDUN Category Coverage Across Data Flows)

To evaluate the breadth of classical privacy threat identification, Tables IV and V summarize, for each data flow in the studied systems, the set of LINDDUN categories detected by the evaluated methods. Rather than listing each category in a dedicated column, the tables present the activated category set together with its overall count. This compact representation enables a clear, flow-by-flow comparison among the three approaches: PILLAR, as an automated LINDDUN analysis tool, and the two PriMod4AI variants powered by GPT-OSS and LLaMA-3.1.

The results illustrate characteristic patterns in how each method engages with the LINDDUN taxonomy. The PriMod4AI models frequently identify a broad range of categories for individual data flows, reflecting their capacity to generate diverse, context-sensitive threat hypotheses. PILLAR, in turn, produces outputs that reflect its structured analytical process, providing consistent detection aligned with its underlying LINDDUN mapping logic. These differences do not indicate superiority of one approach over another; rather, they highlight the methodological distinctions between a deterministic LINDDUN engine and LLM-driven reasoning processes that are capable of exploring a wider interpretation space.

Taken together, the tables provide a unified view of category-level activation across the evaluated systems and methods. This presentation supports an assessment of how comprehensively each approach engages with the classical LINDDUN framework, without presupposing which style of analysis is preferable.

C. Layer B: Combined Threat Space and Model-Centric Analysis

Unlike classical LINDDUN threats, model-centric privacy risks originate from the behaviour of trained models and do not correspond to specific data flows or system components. These risks typically arise from interactions with model parameters, training distributions, latent representations, or generative capabilities, and therefore require an analysis framework that extends beyond the structural boundaries of a DFD. To account for these phenomena, PriMod4AI extracts AI-specific threat descriptions generated by the LLMs and consolidates them into a set of canonical model-centric threat categories through clustering.

Because GPT-OSS and LLaMA-3.1 often describe conceptually similar attacks using different surface forms, for example “AI-Generated Misinformation Using Location Data” and “AI-Fabricated Location Misinformation”, so clustering step is essential for producing a unified taxonomy. This process ensures that semantically equivalent threats across systems (face authentication and autonomous driving) and across model variants are represented consistently, even when the LLMs use divergent wording.

The resulting canonical categories capture well-established families of model-centric privacy risks, such as membership inference attacks, model inversion attacks, reconstruction and leakage mechanisms, embedding or template exposure, dataset replication, and model-generated misleading outputs. In the subsequent analysis, we report how many such canonical categories each PriMod4AI variant identifies in each system and examine their overlap.

Table VI reports the number of distinct threat expressions and their clustered canonical categories identified for each system. PriMod4AI (GPT-OSS) consistently identifies a wider range of model-centric threats, whereas the LLaMA variant returns more compact but semantically aligned sets. The analysis demonstrates that PriMod4AI extends privacy reasoning beyond traditional design-time taxonomies by uncovering behavioural vulnerabilities specific to machine learning models.

D. AI Model-Centric Privacy Attack Knowledge Base (AI_Privacy_KB)

The AI_Privacy_KB presented in this appendix provides the complete catalogue of AI-specific, model-centric privacy threats derived from the systematic literature review described in the main paper. The review covered 30 peer-reviewed publications, standards, and authoritative reports published between 2016 and 2025, each offering documented evidence of privacy attacks or vulnerabilities in modern AI systems.

A simplified example of the encoding format is shown below Listing5:

```
{
  "threatId": "6",
  "privacyThreatName": "Inference Attacks on Model
      Outputs",
  "flowType": "Output flow",
  "aiLifecycleStage": "Inference",
  "shortDescription": "Attackers infer sensitive
      data from the model's predictions or outputs."
      ,
  "privacyThreatJustification": "Can reveal personal
       information, even if the data was anonymized
      or protected during training.",
  "reference": {
    "type": "article",
    "title": "Membership Inference Attacks Against
        Machine Learning Models",
    "authors": ["R. Shokri",
         "Marco Stronati",
         "Congzheng Song",
         "Vitaly Shmatikov"],
    "journal": "2017IEEE Symposium on Security and
        Privacy (SP)",
    "year": "2016",
    "pages": "3-18",
    "url": ""https://api.semanticscholar.org/CorpusID
        :10488675""
  }
}
```

Listing 5. Example entry from the `AI_Privacy_KB` illustrating the standardized JSON schema used for all model-centric privacy threats.

While the main text outlines the construction methodology and the integration of this knowledge base within PriMod4AI,

TABLE IV

LAYER-A LINDDUN THREATS IDENTIFIED FOR EACH DATA FLOW (DF0–DF7) IN THE FACE AUTHENTICATION SYSTEM, COMPARING PRIMOD4AI (GPT-OSS AND LLAMA VARIANTS) WITH PILLAR. EACH CELL LISTS THE THREATS FOLLOWED BY THE NUMBER OF LINDDUN CATEGORIES IDENTIFIED (IN BRACKETS). THREAT ABBREVIATIONS: L = LINKABILITY, I = IDENTIFIABILITY, DI = DISCLOSURE OF INFORMATION, DT = DETECTABILITY, U = UNAWARENESS, NR/NR = NON-REPUDIATION, NC/NC = NON-COMPLIANCE.

Data Flow	PriMod4AI (GPT-OSS)	PriModAI(Llama3.1)	PILLAR
DF0	L, I, D, DD, U, Nc **[6]**	L, I, D, DD, U, Nc **[6]**	L, I, DD **[3]**
DF1	L, I, D, DD, U, Nc **[6]**	L, I, D, DD, Nr, U, Nc **[7]**	L, I, D, DD **[4]**
DF2	L, I, U, Nc **[4]**	L, I, DD, U, Nc **[5]**	L, I, D, DD, Nr, Nc **[6]**
DF3	L, I, D, Nr, DD, U, Nc **[7]**	L, I, D, DD, Nr, U, Nc **[7]**	L, I, D, DD, Nr, Nc **[6]**
DF4	L, I, Nr, U, Nc **[5]**	L, I, D, DD, Nr, U, Nc **[7]**	L, I, D, DD, Nr, Nc **[6]**
DF5	L, I, DD, Nr, U, Nc **[6]**	L, I, DD, Nr, U, Nc **[6]**	L, I, D, DD, Nr, Nc **[6]**
DF6	L, I, DD, Nc **[4]**	L, I, DD, U, Nc **[5]**	L, I, D, DD, Nc **[5]**
DF7	D, DD, Nr **[3]**	D, Nr, U **[3]**	L, I, D, Nr **[4]**

TABLE V

LINDDUN THREATS IDENTIFIED FOR EACH DATA FLOW (DF1–DF14) IN THE AUTONOMOUS DRIVING SYSTEM, COMPARING PRIMOD4AI (GPT-OSS AND LLAMA VARIANTS) WITH PILLAR. EACH CELL LISTS THE THREATS FOLLOWED BY THE NUMBER OF LINDDUN CATEGORIES IDENTIFIED (IN BRACKETS). THREAT ABBREVIATIONS: L = LINKABILITY, I = IDENTIFIABILITY, DI = DISCLOSURE OF INFORMATION, DT = DETECTABILITY, U = UNAWARENESS, NR = NON-REPUDIATION, NC = NON-COMPLIANCE.

Data Flow	PriMod4AI (GPT-OSS)	PriMod4AI (LLaMA3.1)	PILLAR
DF1	L, I, NC, DI, U, NR **[6]**	U, I, DI, NC, L, NR **[6]**	L, I, DI, U, NR **[5]**
DF2	L, DT, DI, U, NC **[5]**	DI, L, DT, U, NC **[5]**	L, I, DI, U, NR **[5]**
DF3	L, DT, DI, U, NC **[5]**	U, DT, L, DI, NC **[5]**	L, I, DI, U, NR **[5]**
DF4	L, I, NR, DT, DI, U, NC **[7]**	L, I, DT, DI, U, NC **[6]**	L, I, DI, U, NR **[5]**
DF5	L, DT, DI, U, NC **[5]**	U, DT, NC, DI, L **[5]**	L, I, DI, U, NR **[5]**
DF6	L, I, NR, DT, DI, U, NC **[7]**	I, L, NR, DI, U, DT, NC **[7]**	L, I, NR, U, NC, DI, DT **[7]**
DF7	L, I, DT, DI, U **[5]**	I, L, DI **[3]**	L, I, DI, U, NR **[5]**
DF8	L, I, NR, DT, DI, U, NC **[7]**	DI, L, NC, I, DT **[5]**	L, I, NR, U, NC, DI **[6]**
DF9	L, I, DT, DI, U, NC **[6]**	DI, I, NC, L, DT, U **[6]**	L, I, NR, U, NC, DI **[6]**
DF10	L, I, DT, DI, U **[5]**	U, DT, L, DI **[4]**	L, I, NR, U, NC, DI **[6]**
DF11	L, I, DT, DI, U, NC **[6]**	DI, NC, L, DT, I **[5]**	L, I, NR, U, NC, DI, DT **[7]**
DF12	L, NR, DT, DI, U, NC **[6]**	L, DI, U, NR, DT **[5]**	L, I, NR, U, NC, DI, DT **[7]**
DF13	L, I, NC, DI, U **[5]**	DI, L, U, I, NC **[5]**	L, I, NR, U, NC, DI **[6]**
DF14	L, I, NC, DI, DT, U **[6]**	DI, L, U, I, NC **[5]**	L, I, DI, U, NR **[5]**

TABLE VI

MODEL-CENTRIC THREAT COVERAGE ACROSS SYSTEMS. CANONICAL CATEGORIES REPRESENT CLUSTERS OF SEMANTICALLY RELATED AI-SPECIFIC PRIVACY THREATS IDENTIFIED BY EACH MODEL.

System	Model	Canonical Categories
Autonomous Driving	PriMod4AI (GPT-OSS)	11
	PriMod4AI (LLaMA)	7
Face Authentication	PriMod4AI (GPT-OSS)	9
	PriMod4AI (LLaMA)	5

this appendix presents the final, structured representation of all identified threats.

Each entry in the AI_Privacy_KB corresponds to a single threat extracted from the reviewed sources. To ensure consistency and reproducibility, all threats were encoded using a unified JSON schema that captures (i) a unique threat identifier, (ii) the normalized threat name, (iii) the associated AI lifecycle stage, (iv) the relevant flow type used in PriMod4AI's reasoning, (v) a concise short description of the attack mechanism, (vi) a justification of its privacy relevance, and (vii) the full bibliographic reference of the originating publication.

All threats in this appendix were processed through a unified pipeline including extraction, terminology normalization, deduplication, and harmonization of lifecycle and flow annotations, ensuring coherent and comparable representations across diverse privacy attack types. This standardized representation also enables direct integration of the knowledge base into automated reasoning workflows and LLM based analysis components within PriMod4AI. Moreover, maintaining a consistent schema simplifies auditing and validation of the collected threats and supports incremental updates as new research emerges.

Tables VII and VIII present the complete AI_Privacy_KB in its final tabular form, consolidating threats from all 30 reviewed publications. This appendix serves as the authoritative source for AI specific threats used in PriMod4AI and supports transparency, reproducibility, auditing, and future extension of the knowledge base. Each threat is normalized into a consistent representation that includes a concise description and traceable references to the original literature, ensuring that the mapping process remains verifiable. The structured format also enables systematic integration of additional threats in future revisions and facilitates automated processing within the proposed framework. Furthermore, presenting the threats in a unified schema helps ensure comparability across different AI lifecycle stages and strengthens the interpretability of the resulting threat analysis.

TABLE VII
THIS TABLE LISTS AI-SPECIFIC PRIVACY THREATS EXTRACTED FROM REVIEWED AND STANDARDIZED SOURCES, PRESENTED IN THEIR SIMPLIFIED AI_PRIVACY_KB FORMAT.

ID	Threat	Short description	Reference
1	Data Quality Compromise	In processing flows during data cleaning and preprocessing, accidental or intentional degradation of data quality leads to unreliable models that mishandle sensitive data.	C. Sillaber, C. Sauerwein, A. Mussmann, and R. Breu, "Data Quality Challenges and Future Research Directions in Threat Intelligence Sharing Practice," in *Proc. 2016 ACM Workshop on Information Sharing and Collaborative Security (WISCS '16)*, 2016, pp. 65–70.
2	Label Tampering	In processing flows during data labeling, malicious modification of labels causes misclassification that can leak or misrepresent sensitive attributes.	R. Sharma, G. K. Sharma, and M. Pattanaik, "Adversarial Label Flipping Attack on Supervised Machine Learning-Based HT Detection Systems," in *Proc. 2024 IEEE Int. Symp. Circuits and Systems (ISCAS)*, 2024, pp. 1–5.
3	Inference Attacks on Model Outputs	In output flows during inference, attackers analyze model predictions to infer sensitive data that may have been used for training.	R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership Inference Attacks Against Machine Learning Models," in *Proc. 2017 IEEE Symp. Security and Privacy (SP)*, 2017, pp. 3–18.
4	Adversarial Attacks	In model-related flows during training and inference, carefully crafted perturbations cause misclassification and can enable privacy and security violations.	X. Yuan, P. He, Q. Zhu, and X. Li, "Adversarial Attacks and Defenses in Deep Learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 9, pp. 2805–2824, 2019.
5	Model Inversion Attacks	In model-related flows during inference, adversaries exploit model outputs to reconstruct sensitive features or entire training records.	M. Fredrikson, S. Jha, and T. Ristenpart, "Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures," in *Proc. ACM Conf. Computer and Communications Security (CCS)*, 2015.
6	Data Poisoning Attacks	In data collection and model-related flows during data collection and training, adversaries tamper with data to introduce backdoors or targeted errors.	E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How To Backdoor Federated Learning," in *Proc. Int. Conf. Artificial Intelligence and Statistics (AISTATS)*, 2020, pp. 2938–2948.
7	Membership Inference Attacks	In model-related and output flows during training and inference, attackers test whether specific records were part of the training dataset.	R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership Inference Attacks Against Machine Learning Models," in *Proc. 2017 IEEE Symp. Security and Privacy (SP)*, 2017, pp. 3–18.
8	Risk of Non-Compliance	Across data collection, processing, and deployment flows, AI systems fail to meet privacy regulations such as the GDPR, leading to legal and trust risks.	S. Shahriar, S. Allana, S. M. Hazratifard, and R. Dara, "A Survey of Privacy Risks and Mitigation Strategies in the Artificial Intelligence Life Cycle," *IEEE Access*, vol. 11, pp. 61829–61854, 2023.
9	AI-Assisted Hacking	In processing flows during inference, AI tools generate or support cyberattacks, lowering the barrier to sophisticated intrusions.	"From ChatGPT to ThreatGPT: Impact of Generative AI in Cybersecurity and Privacy," *IEEE Access*, 2023.
10	Deep Leakage from Gradients	In model-related flows during centralized or federated training, shared gradients leak enough information to reconstruct training examples.	L. Zhu, Z. Liu, and S. Han, "Deep Leakage from Gradients," in *Proc. 33rd Int. Conf. Neural Information Processing Systems (NeurIPS)*, 2019.
11	Model Extraction via Prediction APIs	In output flows during deployment and inference, black-box querying of prediction APIs is used to clone models and their behavior.	F. Tramèr, F. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, "Stealing Machine Learning Models via Prediction APIs," in *Proc. USENIX Security Symp.*, 2016.
12	Training Data Extraction from LLMs	In output flows during inference and deployment, adversaries craft prompts that cause LLMs to reveal training data.	N. Carlini *et al.*, "Extracting Training Data from Large Language Models," in *Proc. 30th USENIX Security Symp.*, 2021.
13	Side-Channel Attacks	In model-related flows during deployment, adversaries exploit timing, power, or other side channels to infer secrets without direct access to model data.	B. I. Priya, P. V. R. D. P. Rao, and D. V. L. Parameswari, "Shielding secrets: developing an enigmatic defense system with deep learning against side channel attacks," *Discov. Sustain.*, vol. 5, art. 249, 2024, doi: 10.1007/s43621-024-00455-4.
14	Reconstruction Attacks	In model-related and output flows during data processing and deployment, attackers reconstruct private datasets using model outputs and auxiliary information.	S. Shahriar, S. Allana, S. M. Hazratifard, and R. Dara, "A Survey of Privacy Risks and Mitigation Strategies in the Artificial Intelligence Life Cycle," *IEEE Access*, vol. 11, pp. 61829–61854, 2023, doi: 10.1109/ACCESS.2023.3287195.
15	Exfiltration via Cyber Means	In data collection, training, and deployment flows, attackers use network or physical compromise to steal datasets and model weights	"A Survey on Privacy Attacks Against Digital Twin Systems in AI-Robotics," *arXiv*, 2024. [Online]. Available: https://arxiv.org/abs/2406.18812

TABLE VIII
THIS TABLE LISTS AI-SPECIFIC PRIVACY THREATS EXTRACTED FROM REVIEWED AND STANDARDIZED SOURCES, PRESENTED IN THEIR SIMPLIFIED AI_PRIVACY_KB FORMAT.

ID	Threat	Short description	Reference
16	AI-Generated Misinformation	During output flows at deployment and inference, AI systems generate false or misleading content that can harm reputations and privacy.	H.-P. Lee, Y.-J. Yang, T. S. von Davier, J. Forlizzi, and S. Das, "Deepfakes, Phrenology, Surveillance, and More! A Taxonomy of AI Privacy Risks," *arXiv preprint arXiv:2310.07879*, 2023. [Online]. Available: https://arxiv.org/abs/2310.07879
17	AI-Enabled Social Engineering	During output flows at deployment and inference, AI generates highly personalized phishing or manipulation messages that increase the risk of disclosing sensitive data.	H.-P. Lee *et al.*, same as ID 1 above.
18	Critical Data Removal	In processing flows during data cleaning and preprocessing, selective removal of critical data points distorts model behavior and may indirectly expose personal information.	V. Bazarevsky *et al.*, "BlazeFace: Sub-millisecond Neural Face Detection on Mobile GPUs," *arXiv preprint arXiv:1907.05047*, 2019.
19	Biased Data Processing	In processing flows during data preprocessing, biased transformations amplify unfairness and can result in discriminatory handling of personal data.	L. E. Celis, V. Keswani, and N. K. Vishnoi, "Data preprocessing to mitigate bias: A maximum entropy based approach," *arXiv preprint arXiv:1906.02164*, 2020.
20	Data Leakage During Preprocessing	In processing and model-related flows across preprocessing, training, and testing, poor handling of notebooks and pipelines causes unintended exposure of sensitive information.	C. Yang *et al.*, "Data Leakage in Notebooks: Static Detection and Better Processes," *arXiv preprint arXiv:2209.03345*, 2022.
21	Privacy Leakage During Monitoring	In output flows during monitoring, logs, metrics, or traces recorded for performance tracking inadvertently expose sensitive data.	M. Jegorova *et al.*, "Survey: Leakage and Privacy at Inference Time," *arXiv preprint arXiv:2107.01614*, 2022.
22	Poisoning Attacks	In model-related flows during training, attackers inject malicious samples into datasets to corrupt models and enable privacy-relevant misbehavior.	C. Sitawarin *et al.*, "A Survey on Data Poisoning Attacks in Machine Learning," *arXiv preprint arXiv:2301.05412*, 2023.
23	Data Extraction Attacks	In model-related and output flows during inference, adversaries design queries to extract sensitive training data from model outputs.	N. Carlini *et al.*, "Extracting Training Data from Large Language Models," *arXiv preprint arXiv:2012.07805*, 2021.
24	Ethical and Societal Risks	In model-related flows during deployment and inference, AI systems may impact societal norms, autonomy, and rights, creating systemic privacy harms.	"Privacy Risks of General Purpose AI Systems: A Foundation for Investigation Practitioner Perspectives," *arXiv preprint arXiv:2407.02027*, 2024.
25	LLM Data Leakage	In output flows during inference, large language models inadvertently reveal private, proprietary, or training data in generated responses.	"A Survey on Privacy Attacks Against Digital Twin Systems in AI-Robotics," *arXiv preprint arXiv:2406.18812*, 2024.
26	Synthetic Data Inference	In data collection and model-related flows during processing and inference, attackers use synthetic data statistics to re-identify individuals from the original dataset.	C. Zhang, "State-of-the-Art Approaches to Enhancing Privacy Preservation of Machine Learning Datasets: A Survey," *arXiv preprint arXiv:2404.16847*, 2025.
27	Unauthorized AI Tool Usage	Across all flows and stages, employees use unapproved AI tools, bypassing security controls and data governance.	Forbes Technology Council, "Emerging AI Threats To Navigate In 2025 And Beyond," *Forbes*, Feb. 2025. [Online]. Available: https://www.forbes.com/councils/forbestechcouncil/2025/02/12/emerging-ai-threats-to-navigate-in-2025-and-beyond/
28	API-Based Model Stealing	In output flows during deployment, repeated queries to model APIs are used to reconstruct or approximate proprietary models containing privacy-sensitive patterns.	SecureSustain, "International AI Safety Report 2025 – Security & Sustainability," 2025. [Online]. Available: https://securesustain.org/report/international-ai-safety-report-2025/
29	Re-identification of Anonymized Data	In processing and inference stages, AI techniques re-link anonymized data with external sources, re-identifying individuals.	"Artificial Intelligence and Privacy: Examining the Risks and Potential Solutions," *Artificial Intelligence*, 2024. [Online]. Available: https://www.researchgate.net/publication/378545816
30	Malicious Code Generation	In output flows during inference, AI systems generate malware, ransomware, or exploit scripts that can be used to compromise privacy.	"From ChatGPT to ThreatGPT: Impact of Generative AI in Cybersecurity and Privacy," *IEEE Access*, 2023, doi: 10.1109/ACCESS.2023.3300381.