Developing Data Handling Guidelines for Open-Source LLM Training in Compliance with Section 37 under Thailand's PDPA and Related Legal Provisions

การพัฒนาแนวปฏิบัติด้านการจัดการข้อมูลสำหรับการฝึก Open-Source LLM เพื่อให้สอดคล้องกับมาตรา 37 ของ PDPA และบทบัญญัติทางกฎหมายที่เกี่ยวข้อง

Nattakrit Kaewjiboon[1] and Peerapat Chokesuwattanaskul[1]
ณัฐกฤตย์ แก้วใจบุญ[1] และ พีรพัฒ โชคสุวัฒนสกุล[1]
[1]Master of Laws in Business Law (International Program), Faculty of Law, Chulalongkorn University
[1]หลักสูตรนิติศาสตรมหาบัณฑิต สาขากฎหมายธุรกิจ (หลักสูตรนานาชาติ) คณะนิติศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
Received: June 17, 2025
Revised: July 7, 2025
Accepted: July 7, 2025

Abstract

This study examines the application of Section 37 under Thailand's Personal Data Protection Act (PDPA) to the training of open-source Large Language Model (LLM), a context often characterized by decentralization and limited institutional oversight. Beyond doctrinal and comparative legal analysis, the research incorporates semi-structured interviews with open-source AI developers in Thailand to ground legal findings in real-world practices. Drawing from structural risk analysis and practitioner feedback, the study proposes a conceptual framework for LLM-based data handling guidelines. The framework presents a modular, role-sensitive approach accompanied by practical tables to assist data controllers in operationalizing Section 37. Designed to support resource-constrained environments, the proposed guidelines aim to support legally compliant, scalable, and responsible development of open-source LLM in Thailand.

Keywords: Open-Source LLM; PDPA Section 37; Data Handling; Data Security

บทคัดย่อ

งานวิจัยนี้ศึกษาการประยุกต์ใช้มาตรา 37 แห่งพระราชบัญญัติคุ้มครองข้อมูลส่วนบุคคล (PDPA) กับบริบทของการฝึกอบรมโมเดลภาษาแบบโอเพ่นซอร์ส (Open-Source Large Language Model: LLM) ซึ่งมักดำเนินการโดยบุคคลหรือกลุ่มผู้พัฒนาอิสระที่ไม่มีหน่วยงานกลางกำกับดูแลอย่างเป็นทางการ โดยเน้นวิเคราะห์ความเสี่ยงเชิงโครงสร้างและปัญหาการปฏิบัติตามกฎหมายในแต่ละขั้นตอนของการจัดการข้อมูลผ่านวิธีวิเคราะห์เชิงเอกสาร การเปรียบเทียบมาตรฐานสากล ตลอดจนการสัมภาษณ์เชิงคุณภาพกับผู้พัฒนาโมเดลภาษาแบบโอเพ่นซอร์สในประเทศไทย เพื่อสะท้อนข้อเท็จจริงในบริบทการปฏิบัติจริง จากผลการวิเคราะห์ ผู้วิจัยได้เสนอกรอบแนวคิดสำหรับแนวทางการจัดการ

ข้อมูลที่สอดคล้องกับมาตรา 37 ที่สามารถนำไปใช้ในการปฏิบัติจริงได้ แนวทางนี้มิได้มุ่งหมายให้เป็นแนวทางที่ไม่สามารถเปลี่ยนแปลงได้ หากแต่เป็นเครื่องมือสนับสนุนให้ผู้ควบคุมข้อมูลสามารถตีความและดำเนินการตามกฎหมายได้อย่างมีประสิทธิภาพและสอดคล้องกับความเป็นจริงทางเทคนิคของการพัฒนาโมเดลภาษาแบบโอเพ่นซอร์สในประเทศไทย

คำสำคัญ: โมเดลภาษาแบบโอเพ่นซอร์ส; PDPA มาตรา 37; แนวทางการจัดการข้อมูล; ความมั่นคงปลอดภัยของข้อมูล

Introduction

In recent years, the development of open-source LLM has expanded rapidly, enabling broader access to advanced artificial intelligence capabilities. However, this open and decentralized development approach also introduces complex data security challenges in relation to the collection, processing, and handling of personal data (Manchanda, Gupta, Majumder, Shridhar & Vig, 2024).

Although open-source LLM are rooted in technical design, their ability to process, generate, and potentially leak sensitive personal data requires an interdisciplinary approach. Legal-ethical scrutiny becomes essential in jurisdictions like Thailand where data protection laws are still evolving in response to emerging AI risks. In Thailand, the Personal Data Protection Act B.E. 2562 (PDPA) (Royal Thai Government Gazette, 2019) serves as principal legislation for safeguarding personal data, with Section 37 imposing specific obligations on data controllers to implement appropriate technical and organizational measures.

While Section 37 articulates a comprehensive framework for data security, translating its general principles into operational practices remains a practical concern in open-source LLM projects, which are often developed by individuals or small teams operating without formal institutional support. Similar challenges have been observed in the AI governance context, where high-level principles such as trust and accountability, as recognized in frameworks such as the OECD AI

Principles (2019) and UNESCO's AI Ethics Guidelines (2021), face implementation difficulties in decentralized or under-resourced environments (Singh et al., 2024). Open-source initiatives often lack formal governance structures, increasing the risk of unintentional data breaches, insufficient documentation, or unclear accountability.

While Thailand's PDPA is the primary focus of this paper, the challenges it poses are part of a broader global tension. Around the world, privacy laws such as the GDPR in the EU, CPRA in California, and similar frameworks in Asia have raised questions about how decentralized, open-source AI development can meet compliance expectations without formal legal infrastructure.

This study addresses these concerns by evaluating practical compliance challenges under Section 37, synthesizing global and local regulatory insights, and proposing a context-aware, stage-based guideline tailored to the realities of open-source LLM development.

Objective

This research aims to develop a practical and context-aware data handling guideline that operationalizes Section 37 for open-source LLM training in Thailand.

Literature Review

1. Legal Frameworks for Data Protection

The GDPR, CCPA, and Thailand's PDPA establish clear and robust foundations for data protection, including principles of consent, data minimization, and security safeguards. Section 37 of the PDPA articulates specific duties for data controllers.

However, the core challenge in open-source LLM training lies not in the law itself but in its operationalization across fragmented development stages. Failure to implement safeguards at one stage can compromise compliance downstream, making it difficult to maintain the level of protection envisioned by legal standards.

2. AI Governance Standards and Technical Guidelines

AI governance standards such as the NIST AI Risk Management Framework (2024), ISO/IEC 42001, and the OECD AI Principles (2019) recognize the full lifecycle of AI systems and offer structured approaches to risk management, documentation, and accountability. These frameworks are conceptually aligned with the goals of data protection laws and offer valuable guidance.

However, their application often assumes formal governance structures and organizational resources. In open-source LLM development within the Thai context. These standards require contextual interpretation to avoid disproportionate burdens on developers. Adaptation is therefore essential to ensure that safeguards remain both effective and feasible.

3. Case Studies in Open-Source AI Compliance

Case studies such as BLOOM (BigScience Workshop, 2023), Flowise (Dark Reading, 2024), and LessLeak-Bench (Zhou et al., 2024) highlight recurring risks in open-source LLM development. These include challenges in dataset traceability, exposure of sensitive data through insecure deployment, and benchmark contamination.

This study examines such cases to identify critical vulnerabilities across the open-source LLM lifecycle and to inform the design of a context-specific compliance guideline under Section 37 of Thailand's PDPA.

4. Implementation Realities from Interviews

To verify and localize the risks identified in case studies, interviews were conducted with open-source developers in Thailand. While most participants were aware of their PDPA obligations, they faced practical challenges in applying safeguards due to limited tools, unclear processes, and constrained resources. Several participants emphasized the need for context-aware solutions, including simplified documentation templates and lightweight tools tailored to real-world workflows.

These insights confirm that awareness alone does not guarantee legal compliance. Without adaptable support structures, even well-intentioned developers remain exposed to compliance risks. The interview findings thus provide critical grounding for designing a modular and locally relevant guideline that aligns with Section 37 of Thailand's PDPA.

Despite growing attention to AI regulation and governance, existing literature provides limited guidance on how individual developers or decentralized teams can practically comply with Section 37 of Thailand's PDPA in open-source LLM training. This study addresses this gap by integrating legal analysis, technical standards, case evidence, and local interview insights to develop a context-sensitive compliance guideline.

Conceptual Framework

This study proposes a conceptual framework grounded in the principles of compliance-by-design, aiming to translate the abstract obligations of Section 37 of Thailand's PDPA into operational strategies suitable for decentralized AI development.

The framework consists of four components: (1) Role segmentation to clarify accountability across development roles; (2) Risk tiering by lifecycle stage to identify phas e-specific vulnerabilities; (3) Contextual legal interpre tation to align PDPA duties with project realities; and (4) Scalable safeguards to support proportio nate compliance based on model purpose, data sensitivity and local development practices.

Methodology

This study adopts a comparative qualitative methodology to explore how legal obligations under Section 37 of Thailand's PDPA can be operationalized in open-source LLM development. This approach is appropriate given the interpretive nature of legal analysis and the need to contextualize compliance practices across diverse development environments.

Data were collected through three sources: (1) Doctrinal and comparative analysis of legal instruments and AI standards (e.g., PDPA, GDPR, NIST AI RMF, ISO/IEC 42001); (2) Case studies of open-source LLM projects (BLOOM, Flowise, LessLeak-Bench); and (3) Semi-structured interviews with Thai developers directly involved in data handling or model training.

The data were analyzed using thematic analysis to identify recurring risks and developer responses, with themes mapped to the open-source LLM lifecycle to inform guideline design. Trustworthiness was ensured through triangulation across legal texts, case studies, and interviews, along with reflective participant validation.

This study hypothesizes that developing supplementary data handling guidelines or future semi-automated tools tailored to open-source LLM training will enable individual developers and small-scale actors, as data controllers, to more effectively implement the security measures required under Section 37 of Thailand's PDPA. By transforming abstract legal mandates into accessible practices, such frameworks can help overcome compliance barriers in decentralized and resource-limited AI development contexts.

Population and Sample

The sample included six participants, which was determined sufficient based on the principle of thematic saturation. Participants were selected using purposive sampling, with inclusion criteria as follows: (1) active involvement in open-source LLM development within the past two years; (2) direct experience in data processing, training, or deployment stages; and (3) familiarity with the PDPA or comparable data protection requirements.

This sample size and selection strategy aimed to balance depth and diversity, capturing a range of developer roles, project scales, and implementation practices within Thailand's open-source AI landscape.

Research Tools

This study employed two primary tools. First, doctrinal legal analysis was used to interpret statutory obligations under Thailand's PDPA, while thematic analysis was applied to AI governance frameworks and case studies to extract recurring risk factors and mitigation approaches.

Second, the interview questions were designed based on supervisory guidance to elicit developers' experiences regarding three core areas: (1) challenges faced during open-source LLM development, (2) strategies used to address those challenges, and (3) unmet compliance needs. The protocol was reviewed for clarity and contextual alignment before use.

Data Collection

Data was collected between January and June 2025 through two methods. First, a literature review examined relevant legal provisions, AI standards, and case studies to identify theoretical risks and best practices. Second, semi-structured interviews were conducted online via video call, phone call, and written responses with six open-source developers. Participants were selected based on their direct involvement in at least one stage of the open-source LLM lifecycle to ensure insights were grounded in real-world compliance practices under the PDPA.

All participants gave informed consent prior to the interviews. To maintain confidentiality, all identifying information was anonymized, and data was securely stored throughout the research process.

Data Analysis: Core Obligations of Section 37 of the PDPA

Section 37 of Thailand's PDPA outlines comprehensive duties for data controllers, including the implementation of appropriate security measures, maintenance of data accuracy, breach notification, and oversight of third-party processors (Royal Thai Government Gazette, 2019). While these requirements are clear in principle, their application in open-source LLM development requires contextual interpretation in settings where roles are decentralized and resources are limited.

Practical Risks in Open-Source LLM Training vs Section 37 Protections

1. Data Collection: Scraped datasets often contain personal data without verified consent (Wang et al., 2024). Although Section 37(1) prohibits unlawful collection, applying this provision to retrospective consent verification in large-scale datasets poses challenges.

2. Data storage: In decentralized projects, shared drives or cloud platforms are often used for collaboration. These introduce risks of unauthorized access if encryption or access logs are missing. Although Section 37(2)– (3) requires encryption and access control, it offers limited technical guidance when personal data is later embedded in model weights (Ayyamperumal & Ge, n.d.).

3. Pretraining & Fine-Tuning: These stages can unintentionally embed personal data into model parameters when public datasets or third-party pretrained weights are used without sufficient filtering (Carlini et al., 2023; Singh et al., 2024). Section 37(4)– (5) governs lawful data use and deletion, but enforcing these duties becomes difficult once sensitive data is internalized by the model.

4. Deployment & inference: Open-source LLM can generate outputs mimicking real individuals or leaking data via adversarial prompts. Section 37 imposes general security duties but does not clarify enforcement at the inference stage (Royal Thai Government Gazette, 2019).

This study distinguishes between model developers, who train or fine-tune models, and model deployers, who release them via APIs or applications. Each role carries distinct legal

and technical responsibilities. Developers are expected to implement pre-deployment safeguards such as output filtering and watermarking while deployers must oversee user interfaces, access control, and usage monitoring. By interpreting Section 37 through a role-based and stage-specific lens, this paper aims to operationalize its obligations within decentralized AI pipelines. In doing so, it supports the application of "appropriate measures" in environments that lack centralized oversight.

Evidence from Case Studies and Interviews

1. Data Collection and Storage

The Flowise incident revealed how weak storage protections in decentralized tools can expose sensitive data, as demonstrated by unsecured vector databases lacking credential checks (Dark Reading, 2024). Meanwhile, BLOOM, a 176-billion-parameter multilingual open-source LLM developed by the BigScience Workshop, encountered challenges in fully tracing data provenance and ensuring consistent consent across its globally distributed contributions (BigScience Workshop, 2023).

Interview insights from six participants revealed that four out of six emphasized storage and dataset concerns as a primary compliance risk. A data collector working on a legal chatbot reported using only public court rulings, which minimized PDPA risk and eliminated the need for extensive data cleaning. However, they noted that other projects often substituted personal information with mock data to reduce liability. A model trainer described using Vaultwarden, a secure vault platform, to protect sensitive data files when working with collaborators. These responses illustrate how developers' roles and resource availability shape their data protection strategies during collection and storage.

2. Training

Three out of six participants discussed training-stage risks, particularly those inherited from upstream datasets or base models. A model trainer explained that although their dataset had been verified and cleaned, they had encountered other open-source base models containing embedded personal data, which could introduce PDPA violations even when fine-tuning was compliant. A data collector highlighted that hallucinations often stemmed from noisy or unfiltered training data. These findings underscore the importance of both dataset curation and understanding model provenance during the training stage.

3. Deployment

The LessLeak-Bench study demonstrated how evaluation benchmarks can be unintentio nally contaminated by training data, leading to privacy leaks through model outputs, and empha sizing the need for traceability and monitoring in post-deployment phases (Zhou et al., 2024).

Five out of six interviewees raised concerns about deployment-stage risks. An enterprise AI trainer noted threats from insider misuse and recommended access controls and auditable outputs. A data cleaning specialist stressed the need to guard against external attacks. A participant in AI system governance highlighted the high cost of retroactive data deletion, suggesting tools such as OneTrust. Even an endpoint user adopted safeguards, using mock data in prompts to avoid memorization of sensitive information. These concerns show that deployment risks span across roles and require targeted safeguards.

From Hypothesis to Practice: Comparing Expected and Actual PDPA Risks in open-source LLM Training

While early legal analysis anticipated that decentralized development would pose compliance challenges under Section 37 of the PDPA, interview findings confirm this with added nuance.

Legal concerns around consent and data protection were broadly confirmed in practice, though the severity of risks varied across projects. Some developers encountered delays in implementing deletion systems due to limited resources, while others worked with anonymized or public domain data where risks were minimal. In storage and pretraining, tools such as credential managers were used in some cases, but automation for enforcing user rights remained limited. This highlights that while encryption supports confidentiality, proactive filtering and minimization at the collection stage are often more feasible than retroactive deletion. During fine-tuning and deployment, some projects required manual filtering to address personal data exposure, whereas others operated in low-risk, internal-use settings. These patterns suggest that rigid, one-size-fits-all compliance measures may overburden low-risk projects, and a risk-tiered approach would better reflect the realities of open-source LLM development.

Overall, the interviews support the thesis argument: Section 37 provides a strong foundation, but developers need practical, role-adapted tools and scalable guidelines.

Comparative Analysis of Legal Frameworks and Technical Standards for Open-Source LLM Governance in Relation to Case Studies and Real-World Insights

The legal frameworks such as Thailand's PDPA (Section 37), the GDPR, and the CCPA provide foundational obligations for data protection. These laws define principles such as lawful processing, data minimization, access control, breach notification, and user rights management (Royal Thai Government Gazette, 2019; European Parliament & Council, 2016; California State Legislature, 2018). While they differ in scope and enforcement style, all three frameworks aim to uphold privacy and accountability in data handling across the AI lifecycle. Their strength lies in setting normative baselines and rights-based duties for developers, processors, and platform operators.

To translate these legal expectations into actionable steps, global and national organizations have released technical standards and governance guidelines. The NIST AI Risk Management Framework promotes layered safeguards and continuous monitoring; ISO/IEC 42001 provides system-level governance protocols; and the OECD AI Principles encourage transparency and fairness across AI use cases (NIST, 2024; BSI, 2023; OECD, 2019). In Thailand, TDPG 3.0, PDPC guidelines, and ETDA's AI Governance Framework mirror these values by recommending ethical AI practices, proportional safeguards, and risk-based security planning (PDPC, 2022; ETDA, 2023). These standards serve as bridges between legal rules and technical implementation, helping developers understand how to meet compliance obligations in practice.

However, most of these frameworks assume centralized governance, formal oversight, and clear lines of accountability. This condition differs in open-source LLM development. For example, the BLOOM project, a multilingual LLM developed by hundreds of volunteers worldwide, struggled to uniformly verify data provenance or ensure informed consent across its distributed dataset contributions (BigScience Workshop, 2023). Similarly, the Flowise incident revealed how the absence of centralized deployment controls can result in credential-free access to sensitive vector databases (Dark Reading, 2024). Interviewees also confirmed these structural gaps. One participant noted that internal misuse remained possible even in enterprise deployments, while another emphasized that decentralized teams often lacked retroactive data deletion systems or budget for tools. A data collector mentioned that while their own legal chatbot used reliable court rulings, many community-led projects relied on scraped content without filtering or contributor agreements. These examples illustrate that even with good intentions, the lack of formal control structures makes it difficult to apply traditional legal duties and technical standards uniformly across open-source projects. As such, operationalizing legal compliance in this space requires a more flexible, context-aware approach.

The comparative review shows a mismatch between current frameworks and open-source LLM realities. Legal instruments like Thailand's PDPA, GDPR, and CCPA offer strong foundations but assume centralized control and defined accountability. Technical standards such as NIST AI RMF and ISO/IEC 42001 translate these laws into practice yet still rely on formal oversight and stable roles. However, case studies and interviews show that open-source LLM operate in fragmented, role-fluid environments with ad hoc safeguards and unclear responsibilities. This gap calls for compliance models that are modular, context-aware, and tailored to decentralized development settings.

Designing Practical Guidelines for Section 37 Compliance

To build a practical compliance guideline for Section 37, this framework was developed by integrating legal reasoning, international standards, technical case studies, and interview insights. The next sections outline how each input was translated into modular strategies across the open-source LLM development stages.

1. Operational Challenges in Contex tualizing Section 37

Open-source LLM projects differ in the types of data they process, the potential impact of their outputs and in team structure and operational capacity. For example, some projects are led by individual developers using pre-cleaned public data such as anonymized court rulings, posing minimal privacy risk. Others involve community-scale contributions or startup-led models that rely on mixed or unverified datasets, increasing exposure to hidden risks. Section 37 of Thailand's PDPA provides flexible mandates, but without contextual guidance, its enforcement may either burden low-risk projects unnecessarily or leave high-risk scenarios under protected. This guideline prioritizes safeguards based on the actual risk posed by the data and model purpose, offering a more scalable and proportional approach to compliance.

2. Rethinking "Appropriate Measures" for Open-Source LLM Contexts

The term "appropriate measures" under Section 37 is deliberately flexible but difficult to operationalize in decentralized open-source environments. In practice, fixed checklists often prove ineffective, either too burdensome for low-risk projects or too vague for higher risk ones. Drawing from global standards and developer interviews, this research proposes interpreting "appropriateness" based on the model's purpose, data sensitivity, and development workflow. Effective safeguards should be traceable, scalable, and adaptable, focusing on real impact rather than rigid compliance rituals.

3. Objective-Sensitive Approaches and Use Cases

Risks in open-source LLM depend not only on data but also on the model's purpose. Section 37 of the PDPA does not provide mechanisms to scale safeguards by use-case sensitivity. This research proposes aligning safeguards with the model's function and risk profile, drawing from NIST, OECD, and Thai guidelines to support proportionate controls.

Empirical evidence reinforces this approach. A developer of a legal chatbot confirmed their use of anonymized public data, with minimal privacy risk and sufficient protection through dataset documentation and issue tracking. By contrast, interviewees working with sensitive or user-generated data highlighted challenges such as the absence of filtering, lack of auto-deletion, and misconfigured deployments, as seen in the Flowise case.

These insights justify distinguishing between general-use models, which carry low privacy risk and require basic safeguards, and high-risk models, which process sensitive or dynamic inputs and demand stronger protections such as logging and deployment monitoring. This distinction enables compliance strategies under Section 37 to match real-world use cases.

4. Supporting Tools and Practical Techniques

To improve applicability for individual developers and small teams, this guideline categorizes tools into three functional types that correspond to key duties under Section 37 of Thailand's PDPA.

First, compliance support tools assist in enforcing core privacy safeguards, such as Presidio for PII detection or Zero Trust Architectures for securing access, aligning with Section 37(1) on lawful data handling.

Second, documentation assist tools facilitate traceability and internal accountability, including datasheet templates, model cards, commit messages, and issue trackers, thus supporting Section 37(2) by creating verifiable records of data practices.

Finally, audit and risk monitoring tools provide oversight during model deployment and use, with mechanisms like rate-limiters, audit loggers, and adversarial testing, which reflect Section 37(3)'s emphasis on response-readiness and post-deployment safeguards (Fernandez & Brazhuk, 2022; Zhang et al., 2024).

By classifying tools in this manner, the framework enables developers to select safeguards that match their model's purpose and risk profile without relying on formal infrastructure, thereby embedding compliance-by-design into everyday development workflows. This approach minimizes implementation burden and remains accessible to all team sizes, from individual contributors to institutional projects.

5. Toward a Compliance-by-Design Framework for Open-Source LLM

This framework offers a compliance-by-design approach that aligns with Section 37 by embedding safeguards directly into everyday development workflows. It adopts a flexible structure where controls scale based on model purpose and risk level, ranging from commit logs for low-risk legal bots to tools for high-risk applications. Developers can leverage familiar platforms such as GitHub or HuggingFace for logging dataset provenance, using issue trackers, and managing credentials without centralized oversight. These embedded practices foster traceability and legal accountability in decentralized settings. Looking forward, semi-automated tools such as LLM-based monitors could support real-time risk flagging and documentation assistance. While this raises ethical questions around visibility and metadata use, it signals a shift toward participatory co-governance. Ultimately, the framework bridges Section 37 with the realities of open-source AI by prioritizing adaptability, real-world workflows, and lightweight safeguards.

Result: A Context-Aware Compliance Framework for Open-Source LLM

To translate Section 37 of Thailand's PDPA into actionable steps for open-source LLM developers, this study introduces a streamlined, context-aware compliance guideline based on four key stages of the open-source LLM training pipeline: data collection, storage, model training (pretraining and fine-tuning), and deployment. The framework emphasizes modular safeguards that align with the purpose and risk profile of each model, addressing the realities of individual developers and small teams.

Each stage is structured around three analytical dimensions: the compliance objective under Section 37 (e.g., lawful handling, internal control, response-readiness), the risk of non-compliance (e.g., improper consent, unsecured access, output misuse), and the appropriate tool types. Tools are grouped into three functional categories: compliance support (e.g., PII detection, credential control), documentation assist (e.g., datasheets, model cards, commit logs), and audit and risk monitoring (e.g., rate-limiters, audit logs, adversarial testing). These tools are chosen for their adaptability and ease of integration into everyday development workflows.

To support contextual compliance, this guideline differentiates between general-use and high-risk models based on three core criteria: (1) the source and sensitivity of training data, (2) the model's intended function, and (3) the manner in which user interactions are processed. These criteria are grounded in empirical findings from case studies and semi-structured interviews with developers involved in open-source LLM training. For instance, a developer of a legal chatbot confirmed the use of solely anonymized public legal data without retaining user prompts, representing a general-use model with minimal privacy risks. In contrast, other interviewees reported challenges in managing prompt-based user data, citing the absence of auto-deletion systems and user rights controls. The Flowise case further illustrates how misconfigured deployments can expose sensitive vector databases (Dark Reading, 2024), reinforcing the need to distinguish between static and dynamic data interactions. These insights inform the categorization framework used in this guideline, allowing Section 37 compliance strategies to align with each model's real-world risk profile.

1. Stage 1: Data Collection

This stage governs how data is sourced and selected before being used for model training. Under Section 37(1), data controllers must collect personal data in a lawful and fair manner, ensuring that sensitive or identifiable information is handled responsibly from the outset.

General-use models operate on vetted, non-sensitive public data. For instance, a legal chatbot project confirmed that all training data was drawn from anonymized court rulings and public legal repositories. Since the datasets do not contain personal identifiers and the purpose is educational, the risk profile is low. In this context, basic documentation assist tools (e.g., source listings or simple markdown logs) are sufficient to demonstrate compliance. However, developers should ensure that public data sources are trustworthy, properly vetted, and free from identifiable personal information to avoid unintentional violations of Section 37(1).

By contrast, high-risk models may integrate collected or user-generated data during fine-tuning, significantly increasing compliance risks. In such cases, developers should employ documentation assist tools, such as Datasheets for Datasets, to record dataset provenance, consent assumptions, and curation processes. Compliance support tools such as Presidio should be applied to filter out sensitive or irrelevant content, and where feasible, mock data may be substituted during testing or validation to minimize exposure to real data.

2. Stage 2: Data Storage

This stage concerns how training datasets and derived information are securely stored and accessed throughout the development lifecycle. Under Section 37(1) of Thailand's PDPA, data controllers are responsible for protecting personal data against unauthorized access or use, while Section 37(3) requires the maintenance of accurate and verifiable records. Both obligations apply throughout the storage phase for high-risk or long-term development projects.

General-use models that rely on static, non-sensitive datasets, lightweight storage practices may suffice. For example, developers working on public legal chatbot projects stored anonymized datasets in encrypted local folders or restricted cloud drives. In collaborative settings, additional controls such as role-based access permissions (e.g., GitHub Teams or shared folder restrictions) and changelog tracking can enhance accountability. Documentation assist tools such as markdown logs and dataset versioning, combined with basic encryption solutions such as VeraCrypt or Google Drive with permission controls, help fulfill legal expectations under Section 37 without adding unnecessary complexity for low-risk applications.

High-risk models, by contrast, often store evolving datasets or user-generated content that may contain personal or sensitive data, requiring compliance-by-design strategies. Developers should implement access control mechanisms using compliance support tools such as GitHub Teams or protected branches to segment responsibilities. For traceability, audit and risk monitoring tools such as MLflow can log data access and training activities. Projects using cloud infrastructure should adopt security tools such as AWS KMS to enforce encryption-at-rest and in-transit. Furthermore, compliance support platforms such as OneTrust can automate data retention policies and deletion workflows, ensuring lifecycle governance and minimizing long-term exposure across development iterations.

3. Stage 3: Pre-training & Fine-tuning

This stage encompasses the model's core training and adaptation processes. Under Section 37(1) and (2) of Thailand's PDPA, data controllers are required to implement safeguards that prevent unauthorized use or disclosure of personal data in contexts where training data may be memorized or reproduced by the model.

General-use models rely on static, anonymized datasets sourced from verified public repositories. For example, models trained on ThaiMOE or other pre-vetted corpora can mitigate compliance risks through manual review and controlled versioning. Developers may utilize documentation assist tools such as dataset cards or spreadsheet logs to record dataset history, while compliance support tools such as Superfilter or rule-based token masking can help remove residual personal data before training begins. Although these models present low exposure risks, optional audit and risk tools can be adopted to track output drift or data leakage over time.

High-risk models, by contrast, involve dynamic adaptation or personalized responses, during fine-tuning. These models face greater risks of hallucination, memorization, or unintended disclosure of training data. Developers should therefore employ documentation assist tools such as Datasheets for Datasets to record dataset provenance, filtering rationale, and legal basis for reuse. Filtering tools such as Presidio must be applied prior to training to identify sensitive phrases or patterns. In addition, RLHF (Reinforcement Learning from Human Feedback) logging mechanisms can be used as audit and risk tools to monitor unexpected behavioral shifts or repeated reproduction of sensitive prompts. Studies (e.g., Carlini et al., 2023) have shown that fine-tuned LLM are more susceptible to leaking memorized text, making red-teaming or pre-deployment simulation exercises crucial before public use. Where appropriate, mock data may be used during early-stage tuning but should not replace legal due diligence for real datasets.

4. Stage 4: Deployment

Deployment is the final and most public-facing stage of the LLM pipeline, where each model must balance accessibility with accountability. Under Section 37(1) and (3) of Thailand's PDPA, data controllers are obligated to implement technical and organizational measures that prevent misuse, restrict unauthorized access, and document interactions for traceability.

For general-use models, which typically operate in low-risk domains and rely on static or non-interactive outputs, developers may adopt lightweight compliance support tools such as static content filters (e.g., profanity-check) and structured prompt templates to reduce the likelihood of generating inappropriate responses. Deployment on platforms with built-in access disclaimers and permission-controlled endpoints such as Hugging Face Spaces or Google Colab with restricted access

High-risk models involve dynamic user prompts, increasing risks of misuse, prompt injection, and hallucination. To address these challenges, developers should combine preventive and reactive safeguards. Compliance support tools such as Perspective API can moderate toxic outputs, while rate-limiting tools such as FastAPI-limiter help manage usage volume. Role-based access control systems such as Firebase Auth or AWS IAM policies restrict high-privilege functions to verified users. Audit and risk tools such as Wazuh or custom log monitoring pipelines should be used to detect anomalies and support incident reviews. Additionally, invisible

watermarking tools such as DeepMind's SynthID embed traceable identifiers in outputs, reinforcing post-deployment accountability. These practices reflect research by Andrus et al. (2024), which emphasizes structured logging and policy-aligned oversight to enable auditability under Section 37(3). In cloud environments, agentless endpoint monitoring frameworks such as the one proposed by Ghaleb et al. (2019) offer scalable activity tracking without intrusive agents. Finally, the level of deployment control should be proportionate to the use context. tricter access controls may suffice for internal deployments, while external deployments require scalable safeguards such as API key gating and automated output tracking to meet PDPA compliance expectations.

Discussion

This study presents a practical framework that translates the legal language of Section 37 of Thailand's PDPA into actionable safeguards tailored to the realities of open-source LLM development. By organizing legal responsibilities across five training stages and mapping them with concrete tools, the framework bridges the gap between abstract compliance obligations and real-world development practices. It highlights that many open-source routines can be repurposed to meet data protection standards without excessive burden.

The framework supports a compliance-by-design mindset, promoting accountability and traceability throughout the AI lifecycle. While it is grounded in Thailand's legal context, the structure reflects universal principles that align with broader data protection regimes such as the GDPR, indicating its potential for international adaptation. However, further cross-jurisdictional research is needed to generalize these findings and test operational feasibility in varied institutional settings.

Despite these constraints, the modular nature of the framework allows for flexible implementation and future enhancement. As automated compliance tools based on open-source LLM themselves, this guideline may serve as a foundational reference for designing scalable, risk-aware governance systems across jurisdictions.

Recommendation

Open-source developers should adopt stage-based safeguards that align with their model's purpose and risk level. Existing tools can be repurposed to support PDPA compliance within the workflow. While optional, each measure in the guideline is grounded in empirical research and should be adjusted based on project needs.

Regulators should clarify enforcement expectations under the PDPA, provide practical compliance examples, and develop mechanisms to support trustworthy AI development in resource-constrained settings.

Policymakers are encouraged to support flexible legal guidance that enables open-source innovation while maintaining accountability. This framework offers a practical approach to interpreting Section 37 in decentralized AI development.

Researchers can expand on this study by evaluating tool effectiveness, identifying compliance gaps, and testing the framework's applicability under different regulatory regimes such as the GDPR.

References

Andrus, M., Jia, A., Jia, R., Koh, P. W., Kummerfeld, J. K., Narayanan, A., & Zhang, J. (2024). *Towards accountable foundation models through auditable model outputs*. arXiv. Doi: https://doi.org/10.48550/arXiv.2504.15585

Ayyamperumal, S. G., & Ge, L. (n.d.). *Current state of LLM risks and AI guardrails*. Carnegie Mellon University. Doi: https://doi.org/10.48550/arXiv.2406.12934

Big Science Workshop. (2023). *Bloom: A 176B-parameter open-access multilingual language model.* arXiv. Doi: https://doi.org/10.48550/arXiv.2211.05100

British Standards Institution. (2023). *Webinar: ISO/IEC 42001–AI management system standard overview*. BSI Group.

California State Legislature. (2018). *California Consumer Privacy Act of 2018 (CCPA), Cal. Civ. Code § 1798.100–1798.199.* Retrieved from https://leginfo.legislature.ca.gov

Carlini, N., Jagielski, M., Tang, L., Tramèr, F., Zhang, C., & Wallace, E. (2023). *Extracting training data from diffusion models*. arXiv. Doi: https://doi.org/10.48550/arXiv.2301.13188

Dark Reading. (2024). *Hundreds of LLM servers expose corporate, health, and other online data.* Retrieved from https://www.darkreading.com/application-security/hundreds-of-llm-servers-expose-corporate-health-and-other-online-data

European Parliamentary Research Service. (2020). *The impact of the General Data Protection Regulation (GDPR) on artificial intelligence. European Parliament.* Retrieved from https://www.europarl.europa.eu/thinktank/en/document/EPRS_STU(2020)641530

Fernandez, E. B., & Brazhuk, A. (2022). *A critical analysis of Zero Trust Architecture (ZTA)*. SSRN. Doi: https://doi.org/10.2139/ssrn.4210104

Ghaleb, A., Traore, I., & Ganame, K. (2019). *A generic agentless endpoint framework for security monitoring of cloud computing endpoints. In 2019 IEEE Conference on Communications and Network Security (CNS) (pp. 1–9).* Doi: https://doi.org/10.1109/CNS.2019.8802828

Government of Thailand. (2019). *Personal Data Protection Act, B.E. 2562 (2019)*. Royal Thai Government Gazette.

Manchanda, S., Gupta, K., Majumder, B. P., Shridhar, K., & Vig, L. (2024). *The open-source advantage in large language models.* arXiv. Doi: https://doi.org/10.48550/arXiv.2412.12004

National Institute of Standards and Technology. (2024). *Artificial intelligence risk management framework: Generative artificial intelligence profile (NIST AI 600-1)*. U.S. Department of Commerce. Doi: https://doi.org/10.6028/NIST.AI.600-1

Organisation for Economic Co-operation and Development (OECD). (2019). *OECD recommendation on artificial intelligence.* Retrieved from https://legalinstruments.oecd.org/en/instruments/OECD-LEGAL-0449

Personal Data Protection Committee (PDPC). (2022). *Guidelines on personal data protection measures, 2022*. Royal Thai Government Gazette.

Royal Thai Government Gazette. (2019). *Personal Data Protection Act B.E. 2562 (PDPA).* Retrieved from https://www.ratchakitcha.soc.go.th/DATA/PDF/2562/A/069/T_0052.PDF

Singh, S., Singhania, P., Ranjan, A., Kirchenbauer, J., Geiping, J., Wen, Y., Jain, N., Hans, A., Shu, M., Tomar, A., Goldstein, T., & Bhatele, A. (2024). *Democratizing AI: Open-source scalable LLM training on GPU-based supercomputers*. arXiv. Doi: https://doi.org/10.48550/arXiv.2502.08145

Wang, Z., Zhong, W., Wang, Y., Zhu, Q., Mi, F., Wang, B., Shang, L., Jiang, X., & Liu, Q. (2024). *Data management for training large language models: A survey*. arXiv. Doi: https://doi.org/10.48550/arXiv.2312.01700

Zhou, X., Weyssow, M., Widyasari, R., Zhang, T., He, J., Lyu, Y., Chang, J., Zhang, B., Huang, D., & Lo, D. (2024). *LessLeak-Bench: A first investigation of data leakage in LLMs across 83 software engineering bench marks.* arXiv. Doi: https://doi.org/10.48550/arXiv.2502.06215