UNCASE: Privacy-Safe Synthetic Conversational Data for Fine-Tuning LLMs in Regulated Industries

Mariano Morales[1] and UNCASE Team[1]

[1]UNCASE AI `https://uncase.md`

Technical Whitepaper — Version 2.0
March 2026

Abstract

Fine-tuning large language models (LLMs) for regulated industries—healthcare, finance, legal, manufacturing—requires domain-specific conversational training data that is both *high-quality* and *free of personally identifiable information* (PII). We present **UNCASE** (Unbiased Neutral Convention for Agnostic Seed Engineering), an open-source framework that implements the *Synthetic Conversation Seed Framework* (SCSF): a five-layer pipeline that (1) strips PII from real conversations to produce anonymized structural blueprints called *seeds*, (2) generates thousands of synthetic conversations guided by these seeds, (3) evaluates quality across nine metrics—including LLM-as-Judge semantic fidelity and embedding drift—with hard rejection thresholds, and (4) exports certified training data in eleven fine-tuning formats with full tool-call support. The system includes five compliance profiles (HIPAA, GDPR, SOX, LFPDPPP, EU AI Act), an adversarial input shield, 150 curated domain seeds, 30 industry-specific tools, and enterprise infrastructure with 106 REST API endpoints, JWT authentication, audit logging, and observability. UNCASE is designed so that real data never reaches the generation or training stages, providing privacy guarantees by architecture rather than by post-hoc anonymization. We describe the framework design, quality assurance methodology, compliance mechanisms, and deployment options.

Keywords: synthetic data, conversational AI, fine-tuning, LoRA, privacy, PII, regulated industries, LLM-as-Judge, differential privacy

1 Introduction

Large language models have demonstrated remarkable capabilities across natural language tasks [2, 18, 21]. However, deploying these models in regulated industries—healthcare, finance, legal services, and manufacturing—requires domain-specific fine-tuning on conversational data that reflects the terminology, decision-making patterns, and regulatory constraints of each sector.

This creates a fundamental tension: the conversations needed for fine-tuning contain sensitive personal data that cannot be used directly due to regulations such as HIPAA [19], GDPR [7], SOX [20],

and the EU AI Act [6]. Existing approaches—manual anonymization, rule-based scrubbing, template-based generation, and generic LLM prompting—each fail to simultaneously preserve domain fidelity, guarantee zero PII exposure, and provide auditable quality assurance.

We introduce **UNCASE** (Unbiased Neutral Convention for Agnostic Seed Engineering), an open-source framework built on the *Synthetic Conversation Seed Framework* (SCSF). The key insight is that real conversations should never be used directly for generation or training. Instead, UNCASE extracts anonymized structural blueprints—*seeds*—that capture the conversational patterns, domain knowledge, and dialog flow without retaining any PII. These seeds then guide the generation of entirely new synthetic conversations that are structurally faithful to the originals but contain no real personal data.

Contributions. This paper describes the following:

1. A five-layer pipeline architecture where PII removal occurs *before* any generation or processing, providing privacy by design rather than post-hoc anonymization (Section 3).
2. A nine-metric quality evaluation system including two semantic evaluators—LLM-as-Judge rubric scoring and embedding drift detection—with hard rejection thresholds that prevent low-quality data from entering training (Section 3.3).
3. An adversarial input protection module (PromptShield) that detects prompt injection, jailbreak attempts, and PII solicitation across five threat categories (Section 4).
4. Five frozen compliance profiles for HIPAA, GDPR, SOX, LFPDPPP, and the EU AI Act, each specifying PII categories, differential privacy budgets, retention policies, and quality thresholds (Section 5).
5. A production-ready implementation with 106 REST API endpoints, parallel pipeline orchestration, 150 curated domain seeds, 30 industry tools, 11 export formats, and GPU deployment automation (Section 7).

2 Related Work

Synthetic data generation. Approaches to synthetic data range from statistical methods [14] and GANs [15] to LLM-based generation [10]. Most focus on tabular or structured data; conversational data presents additional challenges in maintaining dialog coherence, role consistency, and multi-turn factual fidelity.

Privacy-preserving ML. Differential privacy [1, 5] provides formal guarantees for training data protection. Federated learning [12] keeps data distributed. PII detection systems such as Microsoft Presidio [13] and spaCy [8] offer named entity recognition for anonymization. UNCASE combines PII detection with architectural separation: real data is transformed into abstract seeds before any LLM processing occurs.

LLM evaluation. LLM-as-Judge [22] uses language models to evaluate generated text quality. ROUGE [11] measures surface-level overlap. Type-token ratio (TTR) quantifies lexical diversity [17]. UNCASE combines lexical, structural, and semantic evaluation into a unified quality gate with mandatory thresholds.

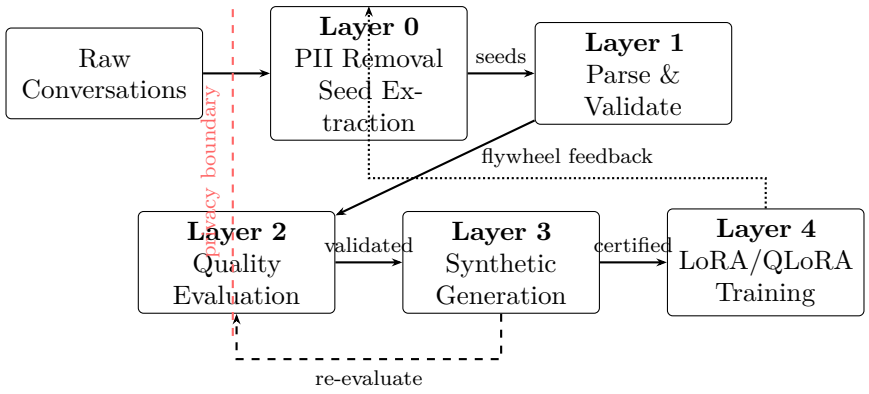

Figure 1. SCSF five-layer pipeline. The privacy boundary (dashed red line) ensures raw data never propagates to generation or training. The re-evaluation loop between Layers 2 and 3 enables feedback-augmented generation. The flywheel provides continuous improvement.

Fine-tuning for domain specialization. LoRA [9] and QLoRA [4] enable efficient fine-tuning of large models. Training data quality directly impacts model performance [3, 23]. UNCASE provides a complete pipeline from raw conversations to certified training data in formats compatible with all major model architectures.

3 The SCSF Architecture

The Synthetic Conversation Seed Framework (SCSF) is a five-layer pipeline where each layer has a single responsibility and communicates via validated Pydantic v2 schemas. The architecture enforces a critical invariant: *raw conversational data never propagates beyond Layer 0.*

3.1 Layer 0: Privacy & Seed Engine

Layer 0 is the zero-trust boundary. It performs two operations:

PII detection. A dual-strategy scanner combines (a) nine regex heuristics for structured identifiers (email, phone, SSN, CURP, RFC, credit card, IP address, IBAN) and (b) Microsoft Presidio [13] with spaCy NER [8] for context-dependent entities (person names, locations, dates, medical licenses, bank accounts, passport numbers). Detected PII is replaced with category tokens (e.g. `[PERSON]`, `[EMAIL]`), producing 14 distinct anonymization categories.

Seed extraction. After anonymization, the engine extracts structural metadata into a `SeedSchema v1` object:

- **Roles**: Participant identities and functions (e.g. “physician”, “patient”).
- **Domain**: Industry vertical classification.
- **Objective**: Inferred conversational purpose.
- **Factual parameters**: Domain constraints, restrictions, expected behaviors.
- **Expected flow**: Logical progression of conversation steps.
- **Tone**: Formal, technical, empathetic, etc.

Table 1. Quality metrics with mandatory thresholds. Gate metrics (marked with †) cause immediate rejection regardless of other scores.

Metric	Threshold	Gate	Description
ROUGE-L	≥ 0.65		Structural coherence with seed
Factual Fidelity	≥ 0.90		Domain fact accuracy
Lexical Diversity (TTR)	≥ 0.55		Vocabulary richness
Dialogic Coherence	≥ 0.85		Inter-turn logical consistency
Tool Call Validity	≥ 0.90		Tool schema correctness
Semantic Fidelity	≥ 0.60		LLM-as-Judge rubric score
Embedding Drift	≥ 0.40		Seed–conversation similarity
Privacy Score	$= 0.00$	†	Zero residual PII
Memorization	< 0.01	†	Extraction attack rate

The seed is a structural blueprint—not data. It contains no PII and no verbatim content from the original conversation.

3.2 Layer 1: Parser & Validator

Accepts four input formats: WhatsApp exports (`chat.txt`), CSV transcripts (call center format), JSON/JSONL objects, and webhook payloads (real-time CRM ingestion). All parsing produces validated Pydantic v2 models with automatic type coercion, constraint checking, and descriptive error messages. Format auto-detection eliminates manual configuration.

3.3 Layer 2: Quality Evaluator

Every generated conversation is scored against nine mandatory metrics with hard thresholds (see Table 1). No conversation enters the training pipeline unless it passes all gates.

Composite score. The overall quality score uses the minimum across all non-gate dimensions, gated by strict privacy and memorization constraints:

$$Q = \begin{cases} \min(r, f, d, c, t, s, e) & \text{if } p = 0 \ \wedge \ m < 0.01 \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where r is ROUGE-L, f is factual fidelity, d is lexical diversity, c is dialogic coherence, t is tool call validity, s is semantic fidelity, e is embedding drift, p is privacy score, and m is memorization rate. Using the minimum rather than the mean ensures no dimension can be weak while others compensate.

3.3.1 Semantic Fidelity (LLM-as-Judge)

A cost-efficient LLM (default: Claude Haiku) grades each conversation on a structured rubric across four dimensions, each scored on a 1–5 scale:

1. **Factual fidelity** (weight: 0.35): Adherence to seed domain constraints.
2. **Logical coherence** (weight: 0.30): Dialog flow rationality.
3. **Role consistency** (weight: 0.20): Participant characterization.

4. **Naturalness** (weight: 0.15): Conversational realism.

The weighted average is normalized from $[1, 5]$ to $[0, 1]$ via $s = (w - 1)/4$. The metric falls back to a neutral score of 0.5 when the LLM API is unavailable, ensuring the pipeline operates offline.

3.3.2 Embedding Drift

Measures cosine similarity between the seed specification and the generated conversation using vector embeddings. Two backends are supported: (a) provider-agnostic LLM embeddings via LiteLLM (e.g. `text-embedding-3-small`), and (b) a TF-IDF fallback that requires no API access. High similarity indicates topical fidelity; low similarity signals semantic drift from the seed specification.

3.3.3 Tool Call Validator

For conversations involving tool use, a five-dimensional validator checks: hallucinated tool names, missing required arguments, unknown arguments, type mismatches, and sequence ordering (exact, subset, or partial order matching against the seed's expected tool sequence).

3.4 Layer 3: Synthetic Generator

The generator uses LiteLLM as a provider-agnostic interface supporting seven LLM providers: Anthropic (Claude), OpenAI, Google (Gemini), Groq, Ollama, vLLM, and any OpenAI-compatible endpoint.

Smart retry. The generator implements an escalating retry strategy: (1) first attempt uses `response_format=json_object` when supported; (2) on JSON format failure, retries without structured output; (3) on each retry, temperature is increased by a configurable step (default: +0.1) to escape degenerate patterns. JSON extraction uses direct parsing and markdown code block extraction—no fragile regex bracket-matching.

Feedback-augmented generation. When a conversation fails quality evaluation, Layer 2 produces specific feedback identifying which metrics failed and by how much. The generator incorporates this feedback as additional prompt instructions in the next attempt, creating a self-correcting loop.

Parallel orchestration. The pipeline orchestrator uses `asyncio.gather()` with semaphore-based concurrency control (configurable, default: 10 concurrent operations) to process seeds, generation, and evaluation in parallel while respecting LLM rate limits.

3.5 Layer 4: LoRA Training Pipeline

The final layer transforms certified synthetic data into trained model adapters using HuggingFace Transformers + PEFT [9] with MLflow experiment tracking. The system exports to eleven fine-tuning formats (Table 2), each with full tool-call training data support.

GPU deployment is automated with scripts supporting auto-detection of GPU type (A40, A6000, A100, H100, RTX 5090), tensor parallelism for multi-GPU serving, LoRA merge pipelines, vLLM inference serving, and optional Cloudflare Tunnel for public access.

Table 2. Supported fine-tuning export formats with tool-call support.

Format	Template	Tools	Target Models
ChatML	`chatml`	✓	GPT-4, Qwen, Yi
Llama 3/4	`llama`	✓	LLaMA 3, 3.1, 4
Qwen 3	`qwen`	✓	Qwen 3, Qwen 2.5
Mistral	`mistral`	✓	Mistral, Mixtral
Nemotron	`nemotron`	✓	NVIDIA Nemotron
Harmony	`harmony`	✓	Cohere Command R+
Moonshot	`moonshot`	✓	Kimi
MiniMax	`minimax`	✓	MiniMax
OpenAI API	`openai_api`	✓	Any OpenAI-compatible
Alpaca	`alpaca`	—	Instruction-tuned

4 Privacy & Adversarial Protection

4.1 Privacy by Architecture

UNCASE does not attempt to "anonymize data well enough." Instead, it ensures that **real data never reaches the generation or training stages**:

1. Raw conversations enter Layer 0.
2. Dual-layer PII detection identifies all personal data (14 categories).
3. Category tokens replace real data (`[PERSON]`, `[EMAIL]`, etc.).
4. Structural metadata is extracted into a seed—a blueprint, not data.
5. Layer 3 creates entirely new conversations from the blueprint.
6. Layer 2 re-scans generated output, rejecting anything with residual PII.

The synthetic conversations *never contained real PII*—they were generated from anonymized blueprints. This is architecturally distinct from the generate-then-anonymize approach common in other systems.

4.2 PromptShield: Adversarial Input Detection

The `PromptShield` module scans all inputs before they reach any LLM, detecting five categories of adversarial content:

1. **Prompt injection**: Override attempts (e.g. "ignore previous instructions").
2. **Jailbreak**: Roleplay-based safety bypasses.
3. **System prompt extraction**: Requests to reveal internal configuration.
4. **Toxic content**: Requests for harmful instructions.
5. **PII solicitation**: Attempts to bypass anonymization (e.g. "use real names").

The module operates in three modes: `audit` (log only), `warn` (log + flag), and `block` (reject). An optional LLM-backed classifier provides enhanced detection for sophisticated attacks that evade regex patterns. Confidence thresholds are configurable per category.

4.3 Privacy Gateway

The LLM Gateway intercepts all messages to and from external LLM providers, scanning for PII in real time. All provider API keys are Fernet-encrypted at rest in the database. The gateway

Table 3. Compliance profiles with PII category counts, differential privacy budgets, and data retention policies.

Profile	PII	ε	Retention	Key Requirements
HIPAA	21	≤ 3.0	7 years	RBAC + MFA, BAA, Safe Harbor
GDPR	17	≤ 5.0	1 year	Right to erasure, DPIA, portability
SOX	9	≤ 5.0	7 years	Audit trail, segregation of duties
LFPDPPP	10	Optional	1 year	ARCO rights (Mexico)
EU AI Act	7	≤ 5.0	—	Risk classification, Art. 11 docs

ε: differential privacy budget for DP-SGD during fine-tuning [1]. Lower values provide stronger privacy guarantees but may reduce model utility.

Table 4. Domain coverage with curated seed counts and scenario template distribution across skill levels and edge cases.

Domain	Seeds	Scenarios	Example Topics
Automotive Sales	50	12 + 5 edge	Financing, trade-ins, fleet sales
Medical Consult.	50	10 + 3 edge	Symptom triage, prescriptions
Finance Advisory	50	10 + 4 edge	Portfolio review, KYC/AML
Legal Advisory	—	8 + 3 edge	Case intake, conflict of interest
Industrial Support	—	8 + 3 edge	Equipment diag., safety incident
Education Tutoring	—	8 + 3 edge	Concept explanation, exam prep

supports three privacy modes matching the PromptShield configuration.

5 Compliance Framework

Five regulatory frameworks are implemented as **frozen dataclass configurations**—immutable, version-controlled, and auditable:

Each profile specifies: (a) which PII categories must be detected and removed, (b) the differential privacy epsilon budget, (c) data retention periods with auto-deletion schedules, (d) quality metric thresholds (stricter for higher-risk domains), and (e) required access control mechanisms.

6 Domain Coverage

UNCASE ships with **150 curated conversation seeds** across three primary domains and **56 scenario templates** across six industry verticals (Table 4).

Each scenario template defines intent, skill level (basic/intermediate/advanced), expected tool sequence, flow steps, edge-case flags, and weighted random selection probabilities for controlling scenario distribution in batch generation.

Domain tools. Each industry includes five built-in tools (30 total) that simulate real-world integrations—e.g. inventory search, patient history lookup, portfolio analysis—enabling generation

Table 5. Codebase metrics as of March 2026.

Metric	Value	Metric	Value
Python source files	203	Curated seeds	150
Python LOC	36,638	Scenario templates	56
Frontend components	132	Domain tools	30
Frontend LOC	62,200	Export formats	11
API endpoints	106	Official plugins	6
API routers	24	Compliance profiles	5
Pydantic models	93	Docker services	8
SQLAlchemy models	18	Quality metrics	9
Alembic migrations	13	SDK wrapper classes	6
Test functions	1,160		

of conversations with realistic tool-calling patterns.

7 Implementation

7.1 System Scale

7.2 API Architecture

The backend exposes 106 REST endpoints across 24 routers built on FastAPI [16] with async PostgreSQL (asyncpg + SQLAlchemy). Key endpoint categories include: authentication (JWT with RBAC), seed management, generation, evaluation, template rendering, tool registration, LLM provider routing, data connectors (WhatsApp, webhook, CSV, JSONL), E2B cloud sandboxes, plugin management, pipeline orchestration, background jobs, knowledge base, usage metering, audit logging, and cost tracking.

7.3 Enterprise Infrastructure

Authentication. JWT access + refresh token pairs with configurable expiration, three-tier RBAC (admin/developer/viewer), organization-scoped data isolation, Fernet-encrypted API keys, and Argon2 password hashing.

Audit logging. An immutable compliance trail records all data access, CRUD operations, authentication events, and pipeline runs with full metadata in a dedicated PostgreSQL table with configurable retention.

Rate limiting. Per-key sliding window with four tiers (Free: 60/min, Developer: 300/min, Enterprise: 1,000/min). Two backends: in-memory (single instance) and Redis sorted sets (distributed), with automatic fallback.

Observability. Prometheus metrics endpoint (`/metrics`) with pre-built Grafana dashboards tracking request rate, latency (avg, p95, p99), error rate, and LLM API duration. Structured JSON logging via structlog. Fire-and-forget usage metering for analytics and billing. Per-organization and per-job LLM cost tracking.

Background jobs. Long-running operations run as tracked jobs with progress reporting, cancellation, automatic retry with exponential backoff, and result persistence.

7.4 Deployment Options

Three installation paths are supported:

1. **Git + uv**: `git clone && uv sync -extra all` (development).
2. **pip**: `pip install "uncase[all]"` (integration).
3. **Docker Compose**: Eight services across four profiles (default: API + PostgreSQL + Redis + dashboard; `ml`: + MLflow; `gpu`: + CUDA API; `observability`: + Prometheus + Grafana).

GPU deployment scripts support auto-detection of GPU type (A40, A6000, A100, H100, RTX 5090), tensor parallelism for multi-GPU inference, automated LoRA merge pipelines, vLLM serving, and optional Cloudflare Tunnel for public endpoints.

7.5 Developer Interfaces

CLI. A Typer-based CLI provides commands for seed creation, generation, evaluation, pipeline orchestration, and template export:

```
uncase pipeline run --domain automotive.sales --count 1000
```

Python SDK. Six wrapper classes—`UNCASEClient`, `Pipeline`, `SeedEngine`, `Generator`, `Evaluator`, `Trainer`—provide programmatic access with both synchronous and asynchronous interfaces.

Dashboard. A React 19 web application (Next.js 16, TypeScript, shadcn/ui, Tailwind CSS 4) with 12 pages covering pipeline management, conversation browsing, quality reports, tool configuration, plugin marketplace, job queue, cost tracking, and audit log browsing.

8 Use Cases

Automotive: Dealership AI assistant. A national dealership network fine-tunes a conversational AI trained on patterns from top-performing salespeople. UNCASE ingests 500 real conversations, strips customer PII, generates 10,000 synthetic conversations maintaining the sales methodology, evaluates quality across all nine metrics (including tool-call correctness for inventory and pricing tools), and produces a LoRA adapter on Qwen 3-14B deployed via vLLM.

Healthcare: Medical triage assistant. A health-tech startup trains a patient intake triage system under HIPAA compliance ($\varepsilon \leq 3.0$, 21 PHI categories). The medical domain seed package (50 seeds, 13 scenarios) generates 5,000 synthetic consultations with semantic fidelity ensuring medical terminology accuracy.

Finance: Compliance-safe advisor. A wealth management firm trains conversational AI for portfolio reviews under SOX + GDPR (intersected requirements, 7-year audit trails). The finance domain pack with KYC/AML edge-case scenarios produces training data with full audit logging capturing every pipeline step.

9 Discussion

Limitations. The current implementation has several acknowledged limitations. First, the LoRA training pipeline (Layer 4) is functional but has not yet been validated with formal DP-SGD integration via Opacus—the compliance profiles specify epsilon budgets that represent planned rather than enforced guarantees. Second, quality metric thresholds were established through expert judgment and iterative tuning rather than empirical validation against downstream task performance. Third, the 150 curated seeds cover three domains deeply but three additional domains rely on scenario templates without pre-packaged seeds.

Architectural advantages. The seed-based approach provides a structural advantage over generate-then-anonymize systems: because real PII never enters the generation prompt, there is no risk of incomplete anonymization in generated output. The re-evaluation loop (Layers 2 $\leftrightarrow$ 3) and the feedback flywheel (Layer 4 $\rightarrow$ Layer 0) create self-improving cycles that increase quality over time. The nine-metric evaluation with hard gates prevents the "quality averaging" problem where strong performance on some metrics masks weakness on others.

Future work. Planned developments include: formal DP-SGD integration with certified epsilon accounting (Q2 2026), empirical benchmark publication against public conversation datasets (Q2 2026), a model marketplace for sharing trained LoRA adapters without sharing data (Q2 2026), additional domain seed packages for real estate, insurance, and e-commerce (Q3 2026), SOC 2 Type I certification (Q3 2026), multi-modal training data support (Q4 2026), and a Kubernetes operator for auto-scaled deployment (Q4 2026).

10 Conclusion

We have presented UNCASE, an open-source framework for generating privacy-safe synthetic conversational training data for LLM fine-tuning in regulated industries. The five-layer SCSF architecture ensures that real personal data never reaches generation or training stages, providing privacy guarantees by design. A nine-metric quality evaluation system with hard rejection thresholds prevents low-quality data from contaminating training. Five compliance profiles, adversarial input protection, audit logging, and enterprise infrastructure make the system suitable for deployment in healthcare, finance, legal, and manufacturing environments.

With 203 Python source files, 106 API endpoints, 1,160 automated tests, 150 curated domain seeds, and five regulatory compliance profiles, UNCASE represents production-ready infrastructure for the next generation of privacy-safe, domain-specific conversational AI.

Availability. UNCASE is open source under the MIT license.
Source code: `https://github.com/uncase-ai/uncase`
Documentation: `https://uncase.md`

References

[1] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 308–318, 2016.

[2] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, vol. 33, pp. 1877–1901, 2020.

[3] L. Chen, S. Li, J. Yan, H. Wang, K. Gunaratna, V. Yadav, Z. Tang, V. Srinivasan, T. Zhou, H. Huang, and H. Ji. AlpaGasus: Training a better Alpaca with fewer data. In *International Conference on Learning Representations*, 2024.

[4] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer. QLoRA: Efficient finetuning of quantized language models. In *Advances in Neural Information Processing Systems*, vol. 36, 2023.

[5] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography Conference*, pp. 265–284, 2006.

[6] European Parliament. Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence (AI Act). *Official Journal of the European Union*, 2024.

[7] European Parliament. General Data Protection Regulation (GDPR), Regulation (EU) 2016/679. *Official Journal of the European Union*, 2016.

[8] M. Honnibal, I. Montani, S. Van Landeghem, and A. Boyd. spaCy: Industrial-strength natural language processing in Python. 2020. https://spacy.io.

[9] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[10] M. Josifoski, M. Sakota, M. Peyrard, and R. West. Flows: Building blocks of reasoning and collaborating AI. *arXiv preprint arXiv:2308.01285*, 2023.

[11] C.-Y. Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pp. 74–81, 2004.

[12] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, pp. 1273–1282, 2017.

[13] Microsoft. Presidio: Data protection and de-identification SDK. 2023. https://github.com/microsoft/presidio.

[14] B. Nowok, G. M. Raab, and C. Dibben. synthpop: Bespoke creation of synthetic data in R. *Journal of Statistical Software*, 74(11):1–26, 2016.

[15] N. Park, M. Mohammadi, K. Gorde, S. Jajodia, H. Park, and Y. Kim. Data synthesis based on generative adversarial networks. *Proceedings of the VLDB Endowment*, 11(10):1071–1083, 2018.

[16] S. Ramírez. FastAPI: A modern, fast web framework for building APIs with Python. 2018. https://fastapi.tiangolo.com.

[17] M. C. Templin. *Certain Language Skills in Children*. University of Minnesota Press, 1957.

[18] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[19] U.S. Congress. Health Insurance Portability and Accountability Act of 1996 (HIPAA). *Public Law 104-191*, 1996.

[20] U.S. Congress. Sarbanes-Oxley Act of 2002. *Public Law 107-204*, 2002.

[21] Gemma Team, T. Mesnard, C. Hardin, R. Dadashi, S. Bhupatiraju, S. Pathak, L. Sifre, M. Rivière, M. S. Kale, J. Love, et al. Gemma: Open models based on Gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

[22] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. P. Xing, H. Zhang, J. E. Gonzalez, and I. Stoica. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[23] C. Zhou, P. Liu, P. Xu, S. Iyer, J. Sun, Y. Mao, X. Ma, A. Efrat, P. Yu, L. Yu, S. Zhang, G. Ghosh, M. Lewis, L. Zettlemoyer, and O. Levy. LIMA: Less is more for alignment. In *Advances in Neural Information Processing Systems*, vol. 36, 2024.