AirGapLite - Hybrid Lightweight Data Minimizer: PII Sharing Policies Via Reinforcement Learning

Group 4: Sriram, Richard, Swetha, Aarti, Arin, Aryan

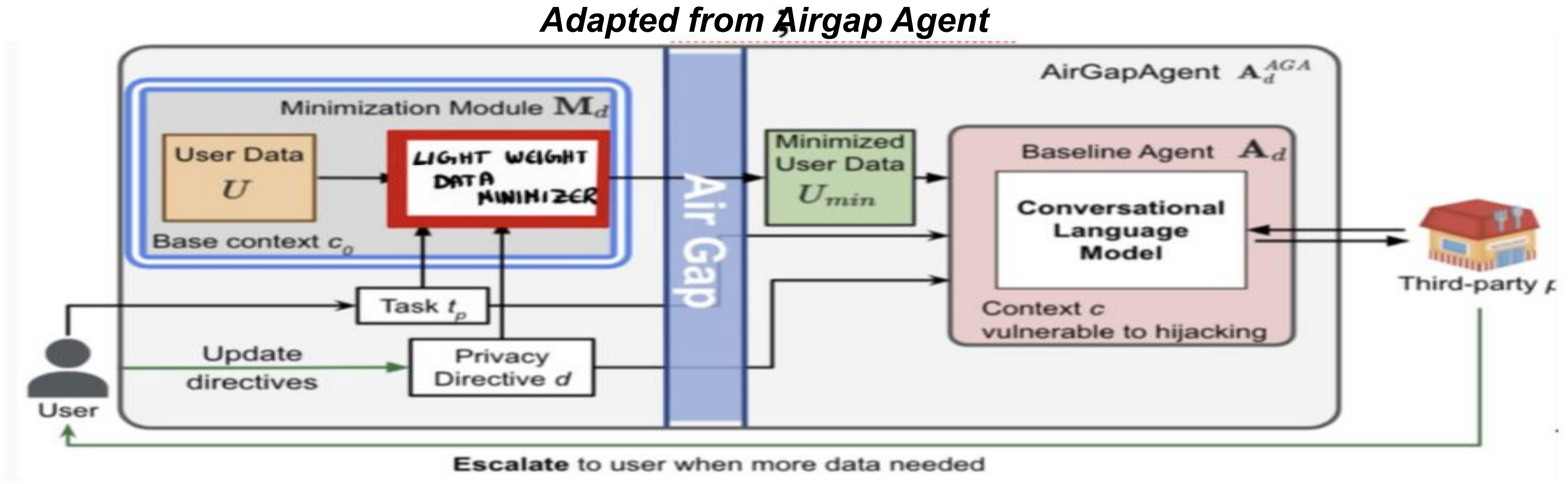
Reference to Original Paper: *E. Bagdasarian et al., "AirGapAgent: Protecting Privacy-Conscious Conversational Agents," CCS'24, 2024.*

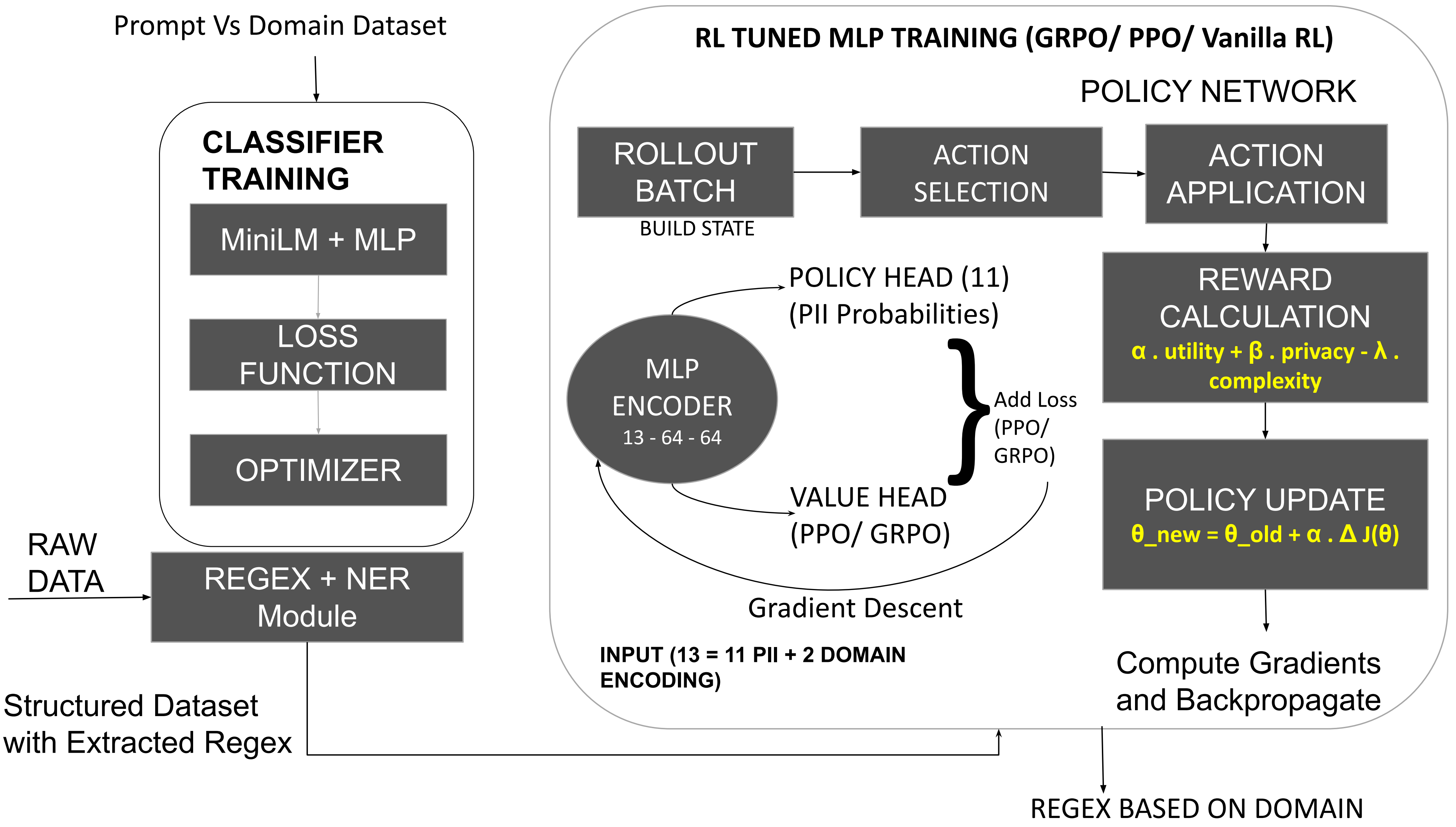
Problem Statement and Motivation

The AirGap minimizer is a privacy-preserving system that determines which PII to share based on context. The original approach(AirGapAgent) uses large language models to make these decisions, but it is generic, slow (full LLM inference per request). We propose AirGapLite- a Reinforcement Learning-based minimizer that learns domain-specific PII sharing policies offline. While training for a new domain takes time, this occurs once during setup; at inference, the trained model makes decisions in microseconds, making it deployment-ready. The RL approach learns highly customized patterns for each domain providing stronger privacy guarantees than generic LLM-based methods while maintaining perfect utility-privacy tradeoffs.

Training and System Architecture

Adapted from Airgap Agent


AirGapAgent A_d^{AGA}
Minimization Module M_d
User Data U
LIGHT WEIGHT DATA MINIMIZER
Base context c_o
Air Gap
Minimized User Data U_{min}
Baseline Agent A_d
Conversational Language Model
Context c vulnerable to hijacking
Third-party
Task t_p
Update directives
Privacy Directive d
User
Escalate to user when more data needed



Prompt Vs Domain Dataset

CLASSIFIER TRAINING

MiniLM + MLP

LOSS FUNCTION

OPTIMIZER

RAW DATA

REGEX + NER Module

Structured Dataset with Extracted Regex

RL TUNED MLP TRAINING (GRPO/ PPO/ Vanilla RL)

POLICY NETWORK

ROLLOUT BATCH

BUILD STATE

ACTION SELECTION

ACTION APPLICATION

POLICY HEAD (11) (PII Probabilities)

MLP ENCODER 13 - 64 - 64

Add Loss (PPO/ GRPO)

VALUE HEAD (PPO/ GRPO)

Gradient Descent

REWARD CALCULATION

α . utility + β . privacy - λ . complexity

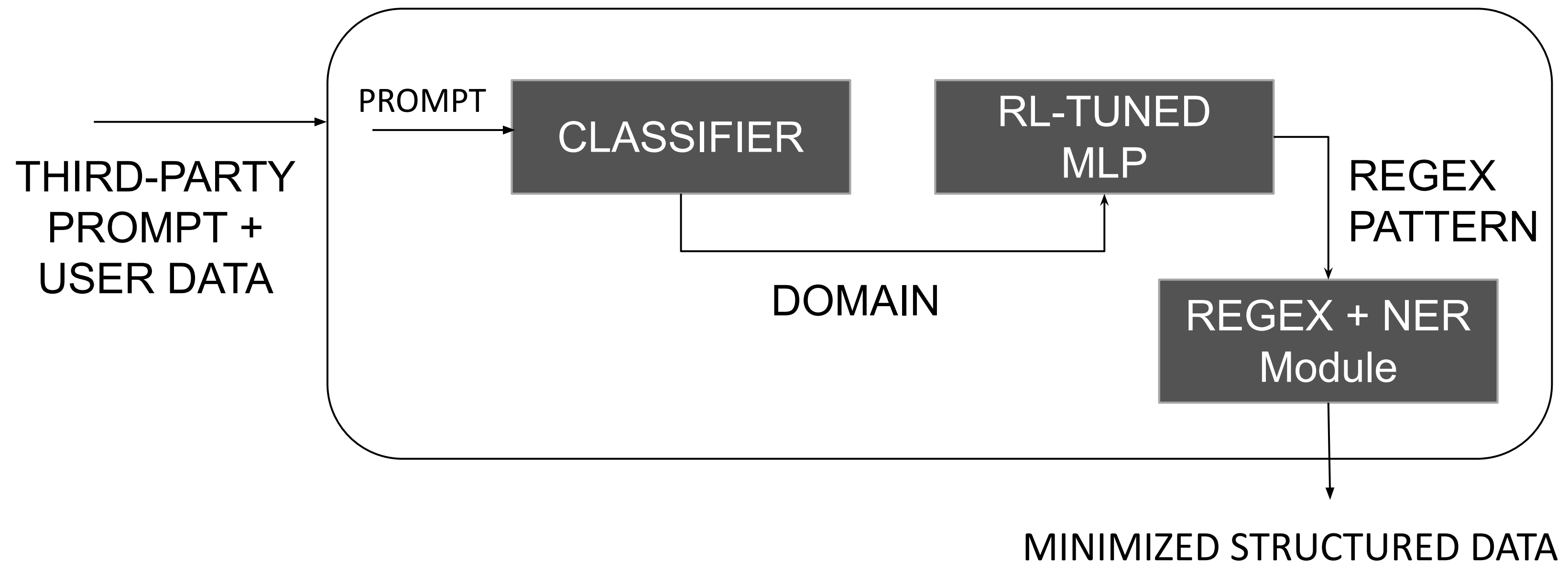
POLICY UPDATE

θ_new = θ_old + α . Δ J(θ)

INPUT (13 = 11 PII + 2 DOMAIN ENCODING)

Compute Gradients and Backpropagate

REGEX BASED ON DOMAIN



THIRD-PARTY PROMPT + USER DATA

PROMPT

CLASSIFIER

DOMAIN

RL-TUNED MLP

REGEX PATTERN

REGEX + NER Module

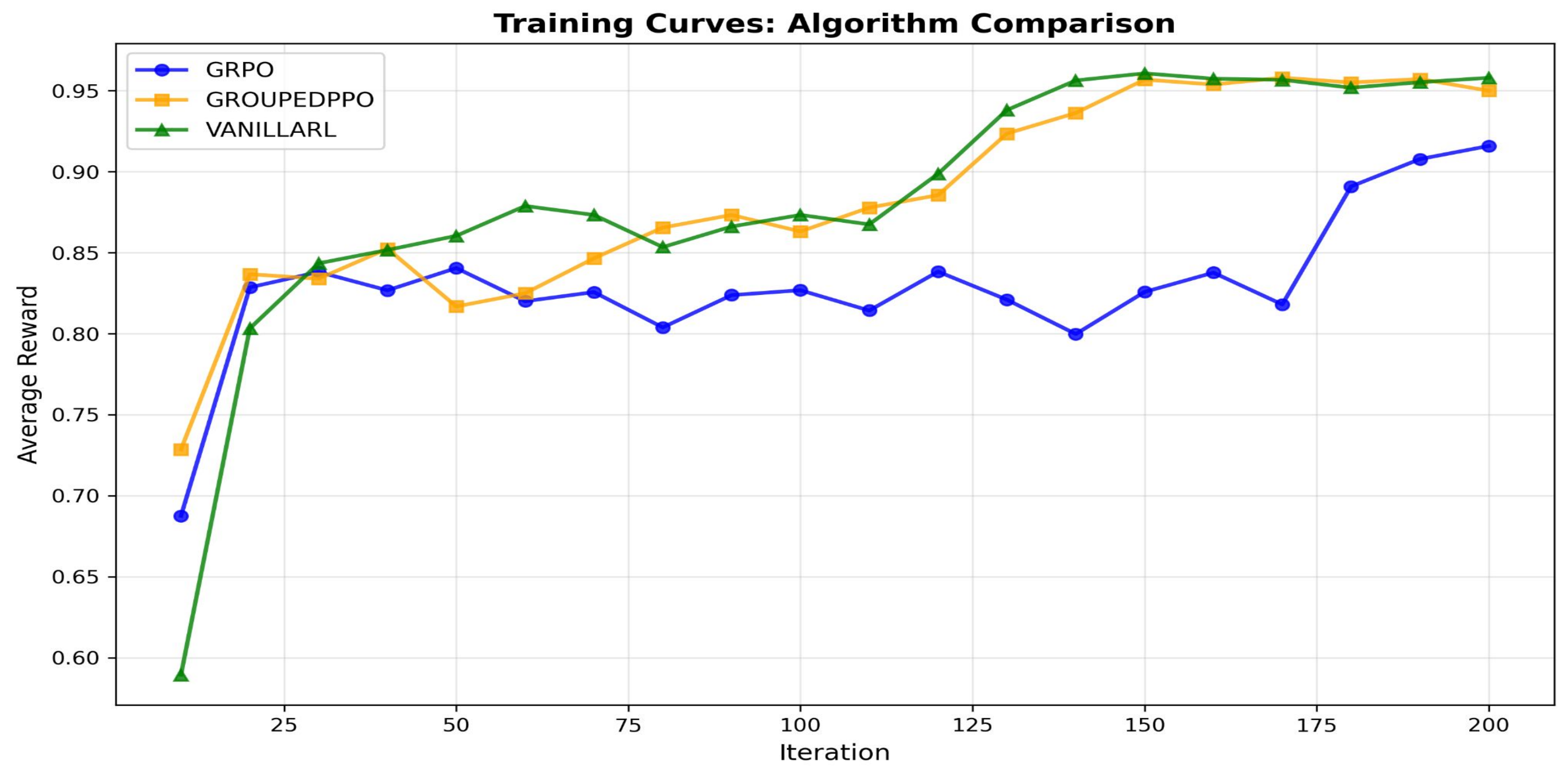
MINIMIZED STRUCTURED DATA

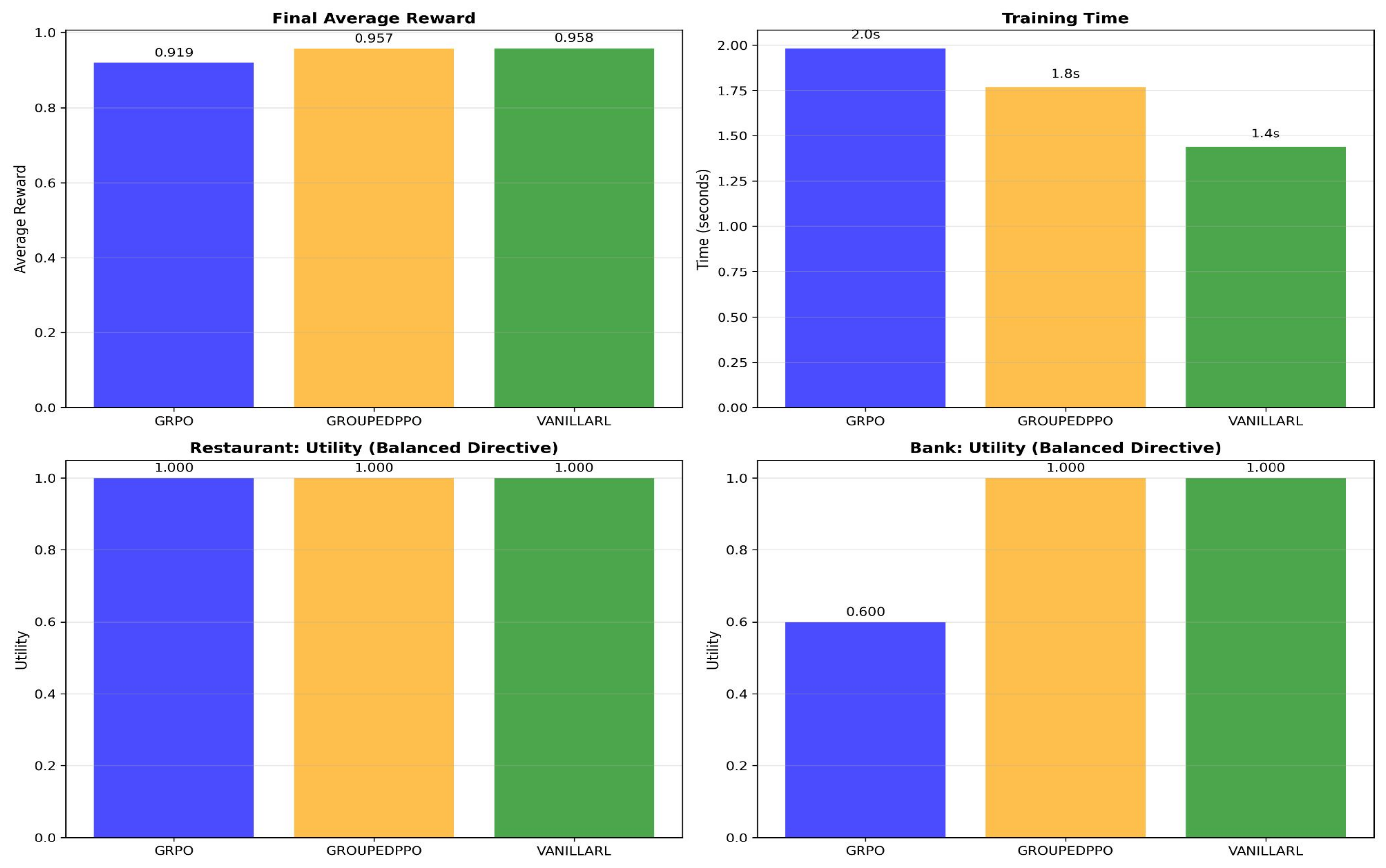
Vanilla RL: REINFORCE policy gradient with no value function; **GRPO:** PPO-style updates with KL regularization only **GroupedPPO:** Standard PPO with clipping plus KL penalty and entropy bonus;

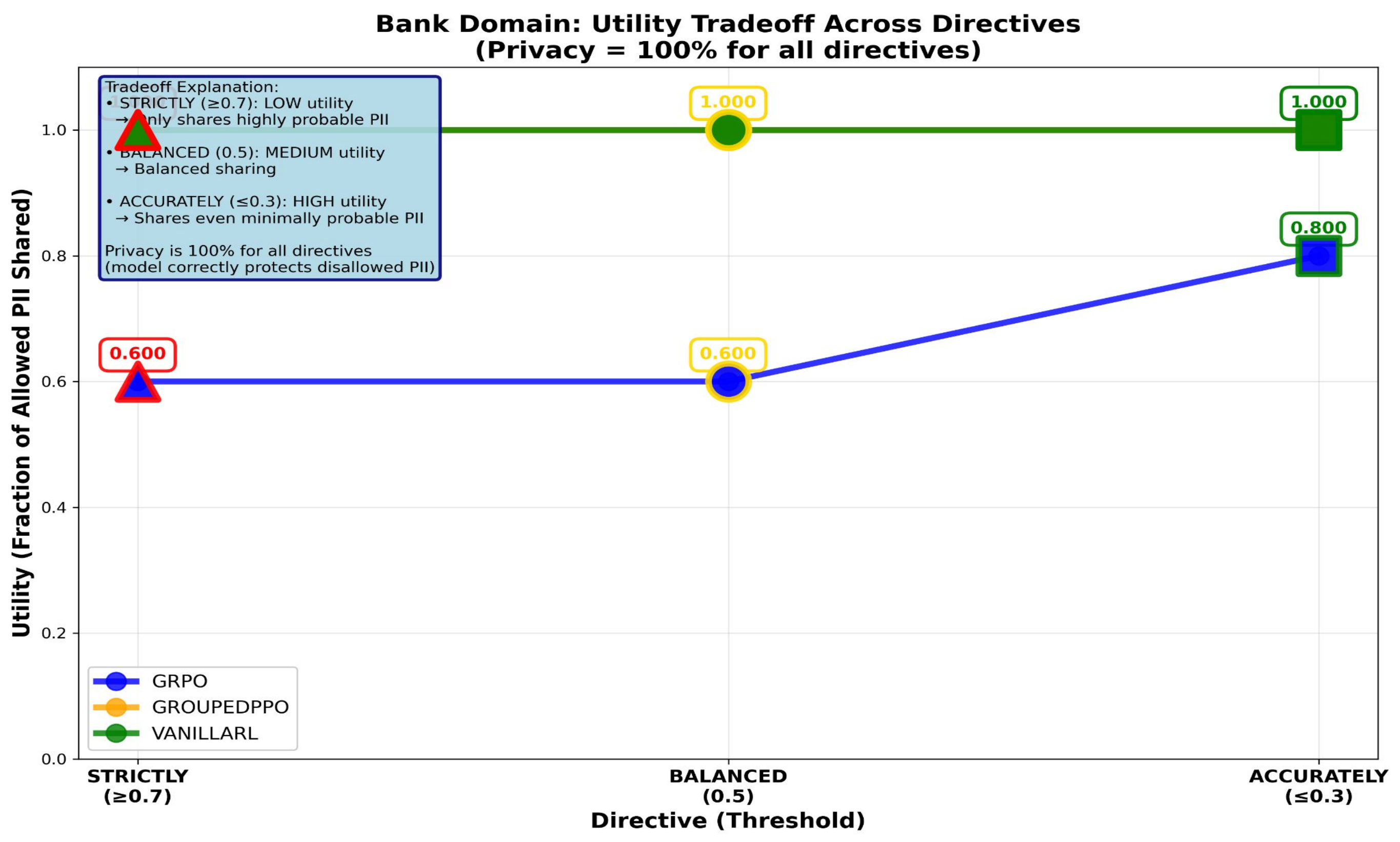
Dataset A – Imbalanced Data (Resembling Real Life): Restaurant: utility = 1.0, privacy ≈ 1.0 for all algorithms (easy pattern), **Bank**: GRPO utility < 1, while GROUPED PPO / VanillaRL reach ≈ 1.0 - misses rarer allowed PII (SSN, CREDIT_CARD, DOB).

Dataset B – 1.5k balanced data: Increased coverage and diversity for SSN, CREDIT_CARD, DOB (≈50–60% presence). Restaurant unchanged; Bank patterns - easy to infer. **utility ≈ 1.0 and privacy ≈ 1.0**

Training Curve and Performance Summary



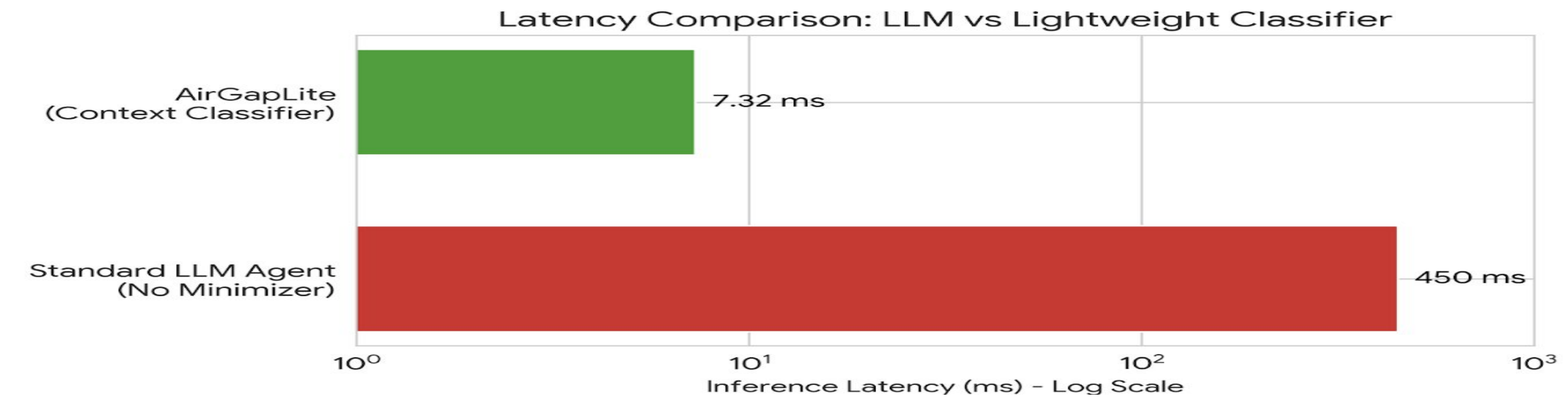




Tradeoff Explanation:
- STRICTLY (≥0.7): LOW utility → Only shares highly probable PII
- BALANCED (0.5): MEDIUM utility → Balanced sharing
- ACCURATELY (≤0.3): HIGH utility → Shares even minimally probable PII

Privacy is 100% for all directives (model correctly protects disallowed PII)

Domain Classifier(MiniLM + MLP) Performance

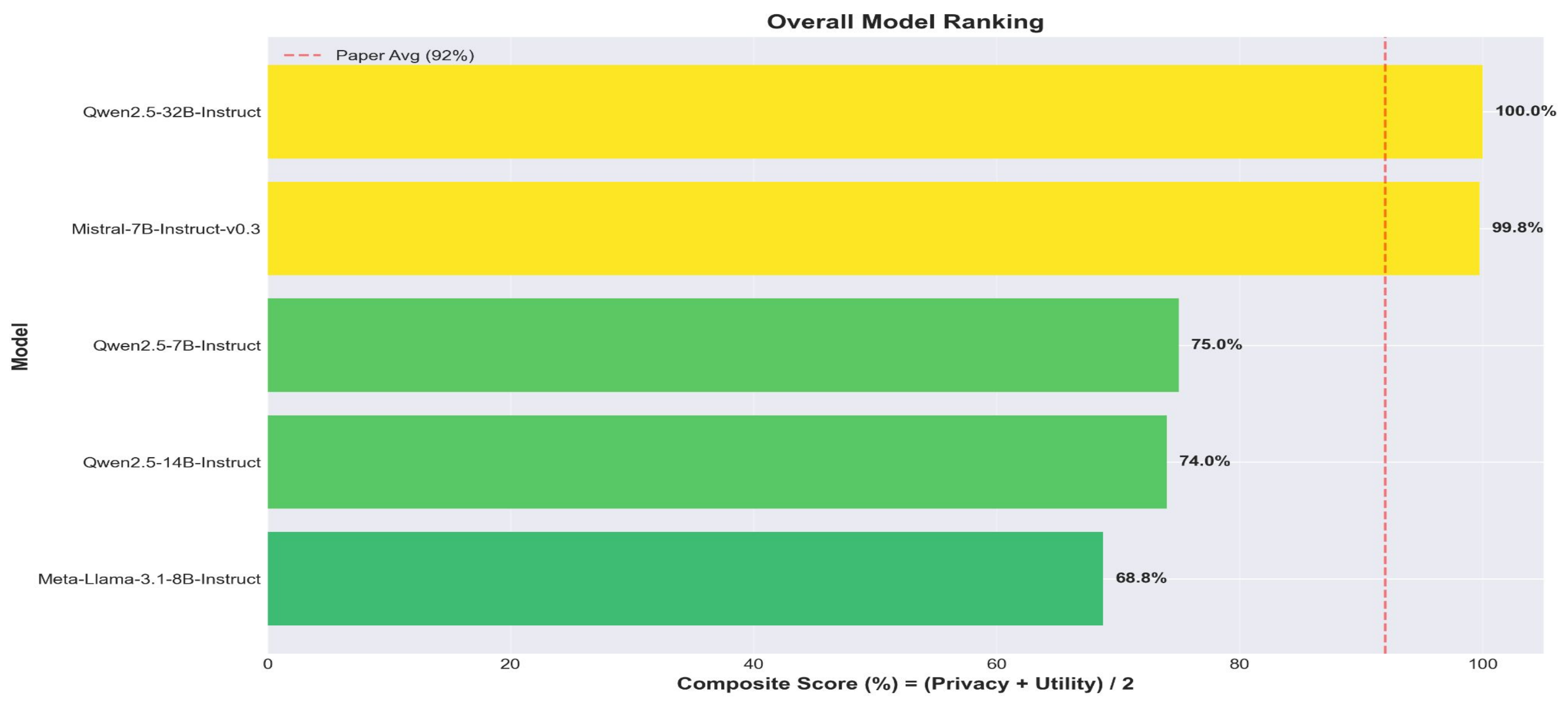


PII Extraction Summary (Regex + Spacy NER)

Dataset	Rows	Precision	Recall	F1 Score	Exact Match
Dataset A (original)	15,805	100%	100%	**1.000**	**100%**
Dataset A (balanced)	1,500	99.3%	100%	**0.996**	**95.8%**
Dataset B (bank-balanced)	1,500	99.3%	100%	**0.996**	**95.7%**

- Our spaCy + regex pipeline achieves F1 >= 0.996 and 100% recall across all datasets, detecting all 11 PII types with no false negatives.
- False positives come only from over-extraction (for example, “vertex” from @vertex.ai), a deliberate high-recall choice since many companies appear like emails structurally within the dataset.

Baseline Inference and Comparison



Aspect	Baseline LLM	RL Pipeline	Improvement
Utility (Restaurant)	100%	100%	—
Privacy (Restaurant)	100%	100%	—
Avg Time (Restaurant)	9.7s	0.7s	**~14x faster**
Utility (Bank)	24%	72%	**+48%**
Privacy (Bank)	90%	100%	**+10%**
Avg Time (Bank)	8.5s	0.75s	**~11x faster**

- LLMs are unstable and hard to customize without fine-tuning or RLHF.
- With good training data, RL gets near-perfect privacy (LLMs cannot guarantee this).
- High one-time training cost, but RL inference is extremely fast — 11–14× speedup in Avg Time for both domains.