Parivarta: An On-Device Zero-Trust Pseudonymization Proxy for Privacy-Preserving Use of Cloud LLMs with Thai Text

Prasert Luekhong*
National Cyber Security Agency (NCSA)
Bangkok, Thailand
prasert.l@ncsa.or.th
**Corresponding author*

Paradet Sakunwachirodom
National Cyber Security Agency (NCSA)
Bangkok, Thailand
paradet.k@ncsa.or.th

Kamolchanok Taweephon
Rajamangala University of Technology Lanna (RMUTL)
Chiang Mai, Thailand
kamochanok_ta66@live.rmutl.ac.th

***Abstract*—Cloud large language models (LLMs) expose any personally identifiable information (PII) contained in a user's prompt to the service provider. This article presents Parivarta, an on-device zero-trust proxy that detects PII locally, replaces it with reversible and referentially consistent pseudonyms before the prompt leaves the device, and restores the model's reply locally. Parivarta targets Thai text, which is written without explicit word boundaries, by combining deterministic pattern matchers, rule-based heuristics, a dictionary-segmented 21k-name gazetteer, and an on-device neural Thai named-entity recognizer through a precision-aware span-resolution rule. The evaluation adopts a privacy-first methodology that separates leak-preventing coverage from typed and exact-span quality. Under an eval-first, one-variable-at-a-time protocol, coverage on a 2,500-document synthetic benchmark rises from 77.1% to 100% and exact-span quality from 65.7% to 100% at near-flat over-mask. Against five baselines, Parivarta strictly dominates per-document coverage on this benchmark, while a minimal-pair context-sensitivity suite indicates that remaining errors are mainly over-masking rather than leakage. Because the smaller hard-negative and robustness suites also guided rule refinement, results on them are reported as development-set findings; validation on independently annotated Thai text and an end-to-end study of utility, restoration, and on-device overhead are identified as future work.**

Keywords*—PII detection, data pseudonymization, privacy-preserving NLP, Thai NLP, named-entity recognition, on-device machine learning, LLM privacy, zero-trust

I. Introduction

Cloud large language models (LLMs) have become everyday tools for drafting, question answering, translation, and analysis. Their convenience carries a structural privacy cost: every prompt is transmitted to, and may be logged or retained by, a third-party provider. When a prompt contains personally identifiable information (PII), such as a customer's name, a national identification number, a telephone number, or a home address, that information leaves the user's control the moment the request is sent. This conflicts with modern data-protection regimes such as Thailand's Personal Data Protection Act [1] and the EU General Data Protection Regulation [2], and with the zero-trust posture organizations increasingly require [3]. The concern is not hypothetical. Language models memorize personal data from their training corpora and can be induced to emit it [4], [5], and frontier models infer sensitive attributes from text that carries no explicit identifier [6].

A natural mitigation is to remove PII from a prompt before it is sent and restore it in the reply. This only helps if the detection and pseudonymization run on the user's device. Server-side redaction reintroduces the very trust assumption that zero-trust protection is meant to avoid. Several recent client-side sanitizers take this position [7]–[10]. All of them were built and evaluated on English, and those that restore masked content reconstruct the mapping by learning it rather than reading it from an authoritative record, which makes restoration probabilistic. Thai makes the task harder still: it is written without inter-word spaces and uses combining vowels and tone marks, so token boundaries and span offsets are less straightforward than in English-centric systems.

This article presents Parivarta, an on-device zero-trust pseudonymization proxy for Thai and English LLM prompts. A hybrid detection ensemble reconciled by a precision-aware span-fusion rule finds PII, which is replaced by reversible, referentially consistent pseudonyms before the prompt leaves the device; the reply is restored locally. Because a missed entity leaks data while over-detection only reduces utility, evaluation reports leak-preventing coverage separately from typed and exact-span quality.

A. Contributions

- An on-device, offline zero-trust pseudonymization proxy for Thai and English LLM prompts, with reversible and referentially consistent surrogate replacement backed by an encrypted local vault.
- A hybrid PII-detection ensemble with a precision-aware span-fusion rule that allows high-recall statistical recognizers to coexist with high-precision deterministic matchers.
- A privacy-first evaluation methodology for Thai span scoring, including Unicode-scalar coordinates and newline handling, along with reproducible ablations and baseline comparisons.
- A minimal-pair ambiguity suite that measures context sensitivity and shows that residual system errors tend toward over-masking rather than leakage.

II. Related Work

Parivarta draws on text de-identification, Thai NLP, and work on privacy in cloud LLM use; formal guarantees and detector robustness are discussed as complementary concerns, and Section II-E states where this work stands.

A. Text De-identification and Anonymization Systems

Automated de-identification began in the clinical domain, where the i2b2/UTHealth shared tasks fixed the span-labeling formulation still in use and showed hybrid rule-and-learning systems beating either component alone [11]. Neural sequence labelers later became the default [12], and toolkits such as Presidio [13] and spaCy [14] package the design as configurable pipelines of recognizers, validators, and anonymization operators. General-purpose LLMs reach

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

competitive recall but have unstable span boundaries and need heavy quantization to run offline [15]. The Text Anonymization Benchmark argues that category-level metrics overstate protection and reports recall-oriented, privacy-weighted scores instead [16], a view adopted in Section V-B. Two limits restrict transfer to this setting: almost every resource assumes whitespace-delimited Latin text [11]–[14], [16], and anonymization in this literature is terminal, discarding the original values, whereas a proxy must splice the model's reply back into the user's own data.

B. Thai Word Segmentation and Named-Entity Recognition

Thai has no spaces between words, so every span-level task inherits the segmentation problem. Dictionary-based maximal matching ships with PyThaiNLP [17], and neural segmenters such as AttaCut trade a little accuracy for speed [18]; because segmentation errors propagate into entity boundaries, segmentation is treated here as a boundary hypothesis that the fusion stage arbitrates (Section IV-D). Thai NER now builds on WangchanBERTa [19] and PhayaThaiBERT [20], with annotated resources such as LST20 [21] and the Thai Nested NER corpus of 264,798 mentions in 104 classes [22]. These inventories were designed for information extraction rather than data protection: person, location, and organization are present, but the categories carrying most of the real PII risk in Thailand, namely national identification numbers, bank account and tax identifiers, house registration numbers, license plates, and hospital numbers, are absent or folded into a generic numeric label. Neural Thai NER also splits checksummed identifiers and over-predicts given-name homographs as persons, and nested annotation [22] leaves a single correct span ill-defined, which motivates both the hybrid ensemble (Sections IV-C and IV-D) and coverage-based scoring.

C. Privacy Risks and Client-Side Sanitization for Cloud LLM Interactions

Trained models can regurgitate verbatim PII [4] at rates that grow with scale, duplication, and context length [23], and scrubbing training data reduces but does not eliminate leakage [5], so treating text as public is not a sound privacy boundary [24]. Frontier models also infer personal attributes from text that contains no explicit identifier [6] and violate contextual-integrity norms about onward flow [25]. For an interactive proxy the first-order risk is simpler: the prompt itself is transmitted to and retained by a third party, which is what the PDPA [1] and GDPR [2] regulate and what zero-trust architecture [3] refuses to take on trust, and practitioner taxonomies list prompt-borne disclosure alongside indirect prompt injection [26]. Server-side sanitization merely relocates trust, so recent systems sanitize on the client. Casper combines rule-based filtering, an ML recognizer, and a browser-resident LLM [7]; Rescriber's user study found a small local model perceived as about as protective as GPT-4o [8]; PAPILLON pairs a trusted local model with an untrusted remote one [9]; and HaS [27] and ProSan [10] attempt restoration but recover the mapping by learning it rather than recording it, so a surrogate can be restored to the wrong value. All were built and evaluated on English, and none reports Thai identifier formats or referential consistency across mentions.

D. Formal Guarantees and Detector Robustness

k-anonymity [28] and differential privacy [29] offer formal guarantees, but adapting them to free text is hard: ADePT perturbs an autoencoder's latent space [30], Yue et al. apply metric-DP token replacement [31], and DP-Prompt paraphrases with a temperature-controlled LLM [32]. Such rewriting targets inference over released corpora, is lossy by construction, and cannot restore original values in a reply. Secure transformer inference such as Iron [33] hides plaintext from the provider but requires its cooperation and remains far more expensive than plaintext inference. Detection-based pseudonymization needs neither, at the cost of an empirical rather than formal guarantee. Because the user controls the input channel, a sanitizer must also survive perturbed input: invisible characters, homoglyphs, and bidirectional controls degrade NLP systems while leaving rendered text unchanged [34], and the Unicode Consortium lists such confusables as security hazards [35]. Thai widens this surface, since combining marks admit several code-point orders and zero-width space is a legitimate word-break marker. Client-side sanitizers report no robustness under such perturbation [7]–[10], [27], and anonymization benchmarks [11], [12], [16] are far cleaner than live prompts; Parivarta therefore separates a normalization view from the emission view (Section IV-B) and tests robustness explicitly (Dataset D).

E. Positioning

Four properties organize the design space: where sanitization runs, whether masking is reversible, whether surrogates stay referentially consistent, and whether the system is evaluated on Thai. De-identification toolkits [13], [14] and the anonymization literature [11], [12], [15], [16] are irreversible and English-centric; Thai NLP resources [17]–[22] provide no PII taxonomy or masking layer; client-side sanitizers [7]–[10], [27] place the trust boundary correctly but restore by inference and report no Thai results; and DP rewriting [30]–[32] and secure inference [33] buy guarantees with utility or provider cooperation. We are not aware of a prior system that holds all four properties. Parivarta runs fully on device and offline, restores deterministically from an encrypted local vault, keeps surrogates referentially consistent, and is Thai-first in both detection and evaluation, reporting leak-preventing coverage separately from typed and exact-span quality and adding a minimal-pair suite that separates context sensitivity from lexical memorization.

III. Problem Formulation and Threat Model

Let a prompt be an input string processed by a local detector that returns a set of typed PII spans. A pseudonymizer replaces each detected span with a stable, type-indexed surrogate, producing a masked prompt and an invertible local mapping. The masked prompt is sent to the cloud LLM, and the returned reply is restored locally by reversing the mapping.

Trust boundary. The user's device is the trust boundary. Detection, pseudonymization, mapping storage, and de-pseudonymization all run locally. Only the masked prompt crosses the network, and the provider never observes the original prompt or the pseudonym mapping.

Adversary. The system assumes an honest-but-curious LLM provider that observes and may retain masked prompts but follows the protocol. It also considers prompt-injection content embedded in the prompt, which is handled by a local pre-filter before network transmission. The provider does not see the unmasked prompt, the original PII values, or the local mapping vault.

Non-goals. The system does not defend against inference from quasi-identifiers that are not detected as PII, side

channels such as prompt length or writing style, malicious providers that tamper with replies, or compromise of the local device and operating system. Protection therefore extends only to the PII that the detector finds. Parivarta provides no formal guarantee such as k-anonymity or differential privacy, and its persistent pseudonyms intentionally keep masked mentions of the same entity linkable within and across turns (Section VII-B).

Objective. The design is privacy-first and treats error types asymmetrically. A missed PII entity is a leak and represents the primary cost to minimize. Over-masking is a utility cost rather than a safety failure. Detector changes are therefore acceptable only when they do not increase leak rate on recall-oriented evaluation datasets.

A. Formal Description

This subsection states the pipeline formally; the notation is used throughout Sections IV–VI.

Notation. Let Σ denote the Unicode-scalar alphabet and let a prompt be a string $x \in \Sigma^*$, of length $n = |x|$ measured in Unicode scalars. All positions are Unicode-scalar indices and a span is a half-open interval $[a, b)$ with $0 \le a < b \le n$; write $S(x) = \{[a, b) : 0 \le a < b \le n\}$ for the set of spans over x, and $x[a : b)$ for the corresponding substring. Let $T = \{\text{PERSON}, \text{PHONE}, \text{EMAIL}, \text{ADDRESS}, \text{NATIONAL_ID}, \text{LOC}, \text{ORG}, \text{IP}\}$ be the type inventory.

Pre-filter and normalization. Let P be the finite set of injection patterns and $N : \Sigma^* \to \Sigma^*$ the idempotent normalizer of Section IV-B (confusable folding, zero-width-character removal, placeholder substitution for URLs, mentions and hashtags, and whitespace collapse). The pre-filter is

$$\Phi(x) = \begin{cases} \bot & \text{if } \exists\, p \in P \text{ such that } p \text{ matches} \\ N(x) & \text{otherwise.} \end{cases} \tag{1}$$

where $\bot$ denotes rejection before any network egress. All subsequent stages operate on $x' = \Phi(x)$, and every span produced over x' is mapped back to coordinates over x by the monotone alignment $\alpha : \{0, \ldots, |x'|\} \to \{0, \ldots, n\}$ induced by N.

Candidates and ensemble members. A candidate is a quadruple

$$c = (t, [a, b), \kappa, \rho) \in T \times S(x') \times [0, 1] \times [0, 1] \tag{2}$$

with t its type, $[a, b)$ its span, κ its member-reported confidence, and ρ its span precision — a fixed per-member weight (Table I) expressing how tight that member's boundaries are. The ensemble of Section IV-C is a family of detectors $D_1, \ldots, D_6$ (L1 deterministic matchers, L2 OS NER, L3 Thai heuristics, L4 segmentation-gated gazetteer, L5 neural Thai NER, L6 user watchlist), each mapping a string to a finite candidate set, and their union is

$$C(x') = \bigcup_{l=1}^{6} D_l(x'), \qquad \rho(c) = \rho_l \text{ for every } c \in D_l(x') \tag{3}$$

Precision-aware fusion. Define the strict weak order $\prec$ on candidates by lexicographic comparison

$$c \prec c' \Leftrightarrow (a, -\rho, -(b-a), -\kappa) <_{\text{lex}} (a', -\rho', -(b'-a'), -\kappa') \tag{4}$$

i.e. earlier start first, then higher span precision, then longer span, then higher confidence, with the member index ℓ as a final tie-break so that the enumeration order, and hence the output of Phase 1, is deterministic. Phase 1 of Algorithm 1 is the greedy non-overlap selection over $C(x')$ enumerated in $\prec$-order,

$$K = \{c \in C(x') : [a_c, b_c) \cap [a_k, b_k) = \varnothing \text{ for all } k \in K \text{ with } k \prec c\} \tag{5}$$

which is to be read as a recursion rather than a set-builder, since K occurs on both sides: let $K_0 = \varnothing$ and, for the i-th candidate c_i in $\prec$-order, $K_i = K_{i-1} \cup \{c_i\}$ if c_i overlaps no member of K_{i-1} and $K_i = K_{i-1}$ otherwise; K is the final set. Termination is immediate because $C(x')$ is finite. Phase 2 closes K under whitespace-separated person merging: for every $\prec$-adjacent pair $p, q \in K$ with $t_p = t_q = \text{PERSON}$,

$$\begin{aligned} &\text{if } x'[b_p : a_q) \in (\text{ws})^* \\ &\text{then } K \leftarrow (K \setminus \{p, q\}) \\ &\quad \cup \{(\text{PERSON}, [a_p, b_q), \\ &\quad max(\kappa_p, \kappa_q), min(\rho_p, \rho_q))\} \end{aligned} \tag{6}$$

Write $E = F(C(x')) = \{(t_i, [a_i, b_i))\}_{i=1}^{m}$ for the fused output, indexed in increasing a_i. By construction E is pairwise disjoint, and the masked-index set is

$$M(E) = \bigcup_{i=1}^{m} [a_i, b_i) \subseteq \{0, \ldots, n-1\} \tag{7}$$

Phase 1 retains, at a shared start, the tighter of two nested candidates; whether it still contains the gold entity is an assumption about detector behavior, not a consequence of the rule. Where it holds, raising a span-precision weight can only shrink $|M(E)|$ without reducing coverage (Eq. (12)), an effect that Section VI-A measures.

Reversible pseudonymization. Let normalize = casefold ∘ trim and let Enc_K be AES-GCM-256 under the Keychain-held key K. The vault is a partial map V from digests to records, updated on first occurrence of a value $v = x'[a : b)$ of type t:

$$k(v) = SHA\text{-}256(\text{normalize}(v)) \tag{8}$$

$$\sigma(t, v) = t_j \quad \text{with } j = 1 + |\{\text{entries of } V \text{ of type } t\}| \tag{9}$$

$$V[k(v)] = (\sigma(t, v), t, Enc_K(v)) \tag{10}$$

If $k(v) \in \text{dom } V$ the stored surrogate is returned unchanged, which gives referential consistency

$$\text{normalize}(v) = \text{normalize}(v') \Longrightarrow \sigma(t, v) = \sigma(t, v') \tag{11}$$

within and across turns. The masked prompt is the left-to-right splice $\tilde{x} = \pi(x', E) = x'[0 : a_1) \cdot \sigma_1 \cdot x'[b_1 : a_2) \cdot \sigma_2 \cdots \sigma_m \cdot x'[b_m : n)$, and the reply is restored by $\pi^{-1}(\tilde{y}, V)$, which substitutes surrogates in descending order of $|\sigma|$ so that no

surrogate is a prefix of another. Only $\tilde{x}$ crosses the trust boundary; x, E and V never leave the device.

Evaluation quantities. Let $G = \{(t_g, g)\}$ be the gold annotation of x. For a gold span $g = [s, e)$ the privacy and utility quantities are

$$\begin{aligned} \mathrm{cov}(g) &= 1\left[g \subseteq M(E)\right], \\ \mathrm{typed}(g) &= 1\left[g \subseteq M(E) \wedge \hat{t}(g) = t_g\right], \\ \mathrm{exact}(g) &= 1\left[g \in E\right] \end{aligned} \tag{12}$$

$$\begin{aligned} \mathrm{leak}(g) &= 1 - \mathrm{cov}(g), \\ \text{over-mask}(x) &= \frac{\left|M(E) \setminus \bigcup_{g \in G} g\right|}{\max\left(1, \left|M(E)\right|\right)} \end{aligned} \tag{13}$$

with $\hat{t}(g)$ the type assigned to the covering span, and specificity = 1 − FP-document rate on a negative set. The privacy-first criterion that gates every mechanism in Section VI is asymmetric in these two quantities: a candidate detector D′ replaces D only if

$$\begin{aligned} &\mathbb{E}\left[\mathrm{leak}(D')\right] \le \mathbb{E}\left[\mathrm{leak}(D)\right] \\ &\text{on Datasets A and B} \end{aligned} \tag{14}$$

with over-mask minimized subject to that constraint. On the minimal-pair suite (Section VI-F) each pair (x^+, x^-) shares one candidate span, and pair accuracy is compared with the independence baseline

$$\begin{aligned} \text{pair-acc} &= \mathbb{E}\left[\mathrm{POS}(x^+) \cdot \mathrm{NEG}(x^-)\right], \\ \text{indep} &= \mathbb{E}\left[\mathrm{POS}\right] \cdot \mathbb{E}\left[\mathrm{NEG}\right] \end{aligned} \tag{15}$$

so that pair-acc − indep is the empirical covariance of the two decisions: pair-acc < indep indicates that they are negatively correlated, which is consistent with triggering on the shared surface form rather than on context.

IV. System Design and Methods

A. Pipeline Overview

A prompt passes through five stages, namely an adversarial pre-filter, normalization, a PII-detection ensemble, precision-aware span fusion, and reversible pseudonymization, after which the model's reply is de-pseudonymized locally. Fig. 1 shows the architecture and the placement of the on-device trust boundary. Every stage that touches the original prompt executes on the device, and only the masked prompt is transmitted.

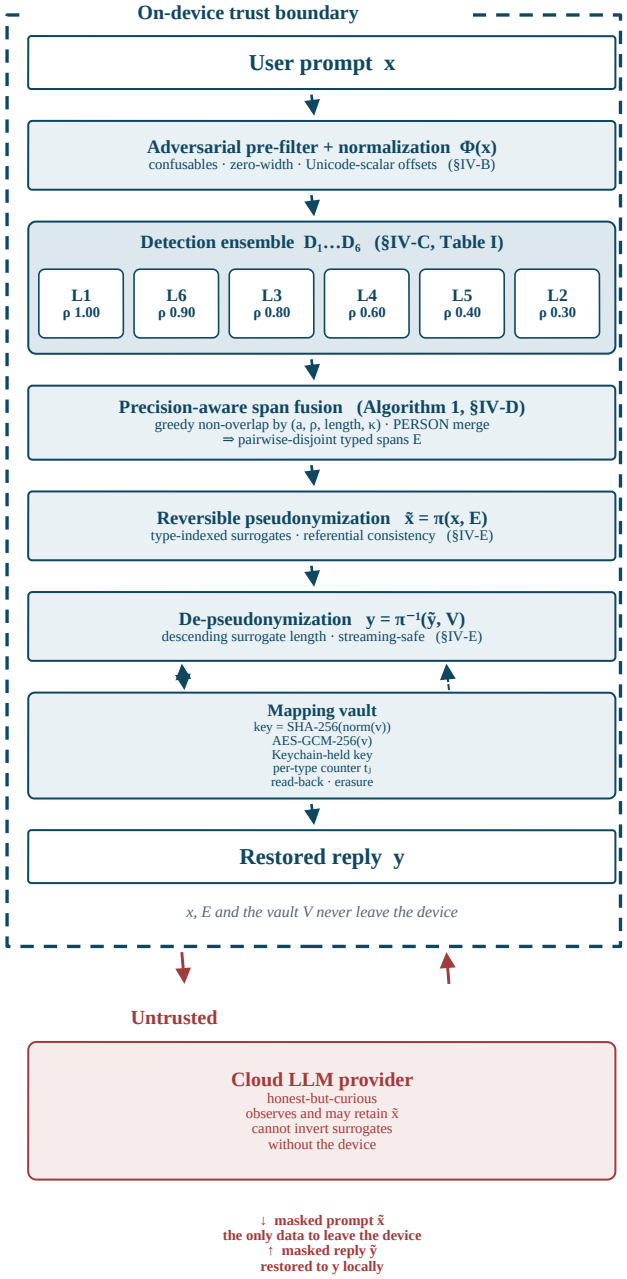


Fig. 1. Parivarta system architecture. Detection, span fusion, pseudonymization, and de-pseudonymization all execute on-device inside the trust boundary; only the masked prompt crosses the network. The six ensemble members (L1 rules, L6 watchlist, L3 heuristics, L4 gazetteer, L5 neural NER, L2 OS NER) are shown ordered by descending span-precision ρ (Table I), the priority used by the fusion rule.

B. Adversarial Pre-Filter and Normalization

Before any content is processed, the prompt is screened against case-insensitive patterns that characterize common prompt-injection and jailbreak attempts, such as requests to ignore previous instructions, reveal system prompts, or activate developer-mode behavior. A match blocks the prompt before network transmission, even in masked form, and the matched patterns can be surfaced to the user for transparency.

The surviving prompt is then normalized to reduce spurious detections and stabilize offsets. Confusables are folded, zero-width characters are removed, URLs, @-mentions, and #-hashtags are replaced by placeholders, and runs of whitespace are collapsed. All subsequent span arithmetic is performed in Unicode-scalar coordinates, a

requirement for Thai text because a visible grapheme may contain multiple Unicode scalars.

C. Detection Ensemble

Each ensemble member emits typed spans with a confidence score and a span-precision weight that estimates boundary tightness. The confidence score reflects belief that a span is PII, while the span-precision weight drives fusion when overlapping candidates compete.

TABLE I. ENSEMBLE MEMBERS, DETECTION METHOD, AND SPAN-PRECISION WEIGHT P

Member	Targets	Method	Span-prec.
Pattern matchers (L1)	EMAIL, ID, PHONE, IP, ADDRESS	Deterministic regex	1.00
User watchlist (L6)	ORG, keywords	Exact string match	0.90
Thai heuristics (L3)	PERSON, LOC	Honorific rule; province lexicon	0.80
Name gazetteer (L4)	PERSON	Dictionary segmentation + name lexicon	0.60
Neural Thai NER (L5)	PERSON, LOC, ORG	WangchanBERTa via Core ML	0.40
OS NER (L2)	PERSON, LOC, ORG	Apple NaturalLanguage	0.30

Table I lists the members, their targets, and their span-precision weights. Pattern matchers (L1) key on canonical structure with bounded patterns, so national IDs, phone numbers, IP addresses, and addresses are not over-captured in Thai text, which earns them the highest weight. The optional user watchlist (L6), disabled by default, matches literal deployment-specific strings with bounded matches. The statistical members are weighted lower because their boundaries are looser: OS NER (L2) supplies open-vocabulary recall, Thai heuristics (L3) use honorific anchors and province lexicons, the gazetteer (L4) segments text before matching so that name substrings do not fire inside unrelated words, and the neural Thai model (L5), converted for Core ML, improves recall for unseen names and locations but over-extends boundaries.

D. Precision-Aware Span Fusion

Candidate spans are reconciled into a non-overlapping set by a two-phase rule. Phase 1 performs precision-aware de-duplication: when multiple candidates start at the same position, the rule prefers the tighter span with the higher span-precision weight over a longer span whose boundary may be polluted by adjacent tokens. Phase 2 merges adjacent PERSON spans separated only by whitespace, recovering person names that segmentation may split into multiple spans.

For overlaps with different start positions, the earlier span is retained as a conservative choice that avoids introducing new leaks. Where the retained tight span still covers the true entity — the assumption made explicit in Section III-A — collateral masking falls with no loss of coverage. This fusion strategy separates the problem of finding entities from the problem of selecting precise boundaries.

Algorithm 1. Span fusion.

1) Sort candidate spans by start position in ascending order, then by span precision in descending order, then by length, then by confidence, and finally by ensemble-member index, so that the ordering is deterministic.
2) Initialize an empty accepted-span set.
3) For each candidate span, add it only if it does not overlap any accepted span.
4) Sort the accepted spans by start position.
5) Merge adjacent PERSON spans when the text between them consists only of whitespace.
6) Return the final non-overlapping fused span set.

E. Reversible Pseudonymization and the Mapping Vault

The masked prompt is produced by a single left-to-right splice over the fused spans: non-PII text is copied verbatim, and each PII span is replaced by a pseudonym drawn from a persistent mapping vault. For a value of a given type, the vault key is the SHA-256 digest of the normalized value, where normalization consists of trimming and case folding, so that surface-equivalent mentions share a single entry.

On the first occurrence of a value, the vault allocates a surrogate of the form type-plus-index, where the index is a per-type monotonic counter (for example, PERSON_001), and stores the pseudonym, the type, and an AES-GCM-256 ciphertext of the original value. Later occurrences return the existing surrogate. This yields referential consistency: the same entity maps to the same pseudonym within and across turns, preserving the coreference structure that the LLM requires while revealing nothing about the original value. The encryption key is a 256-bit key held in the platform Keychain.

The vault supports read-back for a user audit view and full erasure, so retained data remains under user control. The reply is restored by substituting pseudonyms in descending length order, so that a shorter surrogate is not treated as a prefix of a longer one. Under streaming responses, the entire running buffer is de-pseudonymized on every delta, so a surrogate split across chunk boundaries still resolves once it is complete. Only the masked prompt ever leaves the device.

Altered surrogates. Restoration is exact substring substitution, so affixes that leave the token intact, such as an English possessive (PERSON_001's) or a Thai particle written directly after it, are handled. A surrogate that the LLM rewrites (PERSON-001, Person 1), translates, or fuses with a neighboring word is not matched and remains as a visible placeholder. This failure is fail-closed: it harms readability but cannot expose PII, because restoration happens only on the device. Variant-tolerant matching and an instruction to copy placeholders verbatim are planned (Section VII-D).

F. Implementation

Parivarta is a single native application; the Core ML model, tokenizer, segmentation dictionary, and name lexicon are bundled and run without network access. All span arithmetic uses Unicode-scalar offsets, the gazetteer ablation switch is task-local so that parallel evaluation runs do not race, and every component is covered by an executable test harness, so the reported metrics reproduce exactly.

V. EXPERIMENTAL SETUP

A. Datasets

Five complementary datasets are used, each isolating a distinct capability; a sixth, Dataset E, is planned and is described below for completeness. Every span, gold and predicted alike, is compared in a common Unicode-scalar coordinate system as described in Section IV-B.

- Dataset A — development gold set. Fifty Thai sentences containing 77 mentions with substring annotations, used for the gazetteer ablation and as a regression guard.
- Dataset B — independently generated benchmark. Two thousand five hundred synthetic documents covering five PII types with 500 documents each (PERSON, PHONE, EMAIL, ADDRESS, NATIONAL_ID), with one target entity per document and character-offset spans.
- Dataset C — hard-negative suite. Thirty-five Thai sentences that contain no PII but include adversarial look-alikes, plus three held-out format-collision items; this dataset measures false positives.
- Dataset D — robustness probes. Non-templated hard positives comprising 22 mentions, together with a ten-name out-of-vocabulary person probe.
- Dataset AMB — minimal-pair ambiguity suite. 1,250 pairs (2,500 documents) over five PII types, 250 pairs each. Each pair shares a candidate surface form and differs only in context: in the positive the candidate is the entity, and in the hard negative it is a non-PII look-alike, several of which are byte-for-byte valid for a format detector.
- Dataset E — independent real-carrier set. Synthetic PII injected into real Thai carrier text with multiple annotators and inter-annotator agreement reported as Cohen's or Fleiss' kappa. No real personal data is handled. This dataset is planned and not yet available.

All PII in Datasets A–D and AMB is synthetic; no real personal data is used.

Role of each dataset in development. No dataset in this study is a held-out test set. Datasets A and B served as the leak-rate gate of the protocol (Section V-D), and the format fixes E1–E3 were motivated by Dataset B failures. The E5 precision fixes were derived from the culprit analysis of the E4 false positives on Dataset C (Section VI-C), and the E8a ADDRESS generalization was designed in response to the non-template address failures in Dataset D (Section VI-D). Results on Datasets B, C, and D are therefore development-set findings and are likely to be optimistic for unseen text. Dataset AMB was constructed and scored after the detector configuration reported here was fixed, and no component was changed in response to its results. An independently annotated held-out set is planned as Dataset E (Section VII-D).

B. Metrics

All spans are compared in Unicode-scalar coordinates. For a gold span and the set of masked indices, the evaluation reports coverage, the privacy metric, defined as full containment of the gold span in the masked-index set, with leak defined as one minus coverage; typed coverage, which additionally requires the predicted type to match; and exact-span match. On negative data, specificity is reported as one minus the false-positive document rate. Over-mask, the utility cost, is the fraction of masked scalar positions that fall outside gold spans.

On Dataset AMB, each judgment is restricted to the shared candidate span, yielding POS-recall, the true-positive rate at which the candidate is masked; NEG-specificity, one minus the false-positive rate, that is, the rate at which the candidate is left untouched; and pair-accuracy, the rate at which both members of a pair are handled correctly. Pair-accuracy is reported against the independence baseline given by the product of POS-recall and NEG-specificity. All headline numbers carry bootstrap 95% confidence intervals over 10,000 resamples of documents or pairs, and coverage differences between systems are assessed with the paired McNemar test [36] over the shared documents.

C. Baselines

Parivarta is compared against five systems on identical inputs and metrics, using a common span-JSONL format and a single standard-library scorer: Microsoft Presidio [13], with its default analyzer plus custom Thai national-identification and phone recognizers; the multilingual spaCy model xx_ent_wiki_sm [14]; PyThaiNLP's CRF-based thainer NER [17]; a generic regular-expression lower bound; and a zero-shot local LLM, qwen2.5:7b served through Ollama with no cloud access. Dataset E will be added to this comparison once it becomes available.

Baseline configuration. Each baseline was run in its released form with default settings and without fine-tuning: presidio-analyzer 2.2.360, spaCy 3.7.5 with xx_ent_wiki_sm 3.7.0, and PyThaiNLP 5.3.4 with its CRF-based thainer engine (ThaiNameTagger), which is distinct from the WangchanBERTa model thainer-corpus-v2 used as member L5 of Parivarta. Presidio used its default English analyzer (language en, default score threshold), extended with two Thai pattern recognizers, a 13-digit national identification number with optional separators (score 0.6) and a Thai phone number (score 0.4), neither with checksum validation. The regular-expression baseline matches only e-mail addresses, bare 13-digit numbers, Thai phone numbers, and IPv4 addresses. The LLM baseline, qwen2.5:7b served by a local Ollama instance through its OpenAI-compatible endpoint, received a single zero-shot instruction to return a JSON array of {type, text} objects over the eight types of T, copying each value verbatim; no temperature, seed, or token limit was set, so the server defaults applied. Every occurrence of each returned value was located by exact substring search to recover its offsets, and a malformed reply yields no spans. All outputs were converted to the common span-JSONL format, and the scorer folds labels onto T by a fixed canonical map (e.g., PER → PERSON, GPE → LOC, IP_ADDRESS → IP); labels without a counterpart in T are retained, so they count toward coverage and false positives but never toward typed or exact-span matches. The runner scripts, prompt, and canonical map are publicly available (see Data Availability).

D. Protocol

The protocol is eval-first and changes one variable at a time. A baseline configuration (E0) is established, and each mechanism (E1 through E8) is then added singly and accepted only if it does not increase the leak rate on Datasets A and B; E6 and E7 are measurement-only probes. The pipeline is deterministic given fixed model weights and is therefore invariant to the host: the detector figures reproduce across Apple-silicon toolchains, and the Python-side baselines and scorer run under a pinned virtual environment. An end-to-end utility study and on-device latency and memory measurements are outside the scope of this paper and are described as future work in Section VII-D.

VI. Results

A. Ablation: Building the Detector (Dataset B)

Table II traces coverage as each mechanism is added. Coverage is the privacy metric, for which higher is better, while over-mask is the utility cost, for which lower is better.

TABLE II. Cumulative ablation on Dataset B (2,500 documents), micro-averaged

Stage	Change	Coverage	Exact-span	Over-mask
E0	Baseline	77.1%	65.7%	4.5%
+E1	National-ID formats	89.1%	77.7%	4.5%
+E3	Phone 2-4-4 form	91.1%	79.7%	4.5%
+E2	Address detector	100.0%	88.6%	4.5%
+E4	Precision-aware fusion	100.0%	100.0%	4.4%
+E5	Precision fixes	100.0%	100.0%	1.5%

Stages are listed in order of application, which is not their numeric order.

The bottlenecks were surface-format variants rather than modeling capacity. National-ID recall was capped by a regular expression that matched only bare thirteen-digit strings, and E1 lifts NATIONAL_ID coverage from 40.0% to 100.0%. Phone recall was limited by the unhandled 2-4-4 dashed form, and E3 lifts PHONE coverage from 90.0% to 100.0%. ADDRESS was limited by the absence of a dedicated detector, and E2 lifts it from 55.4% to 100.0%. The precision-aware fusion introduced in E4 then corrects boundary over-capture, raising exact-span match from 88.6% to 100.0%, and the E5 precision fixes cut over-mask from 4.4% to 1.5%, as discussed in Section VI-C. No step reduced coverage.

B. Per-Type Detection (Dataset B)

For the final system, every PII type present in Dataset B reaches 100% on all three per-type measures, namely leak-preventing coverage, typed coverage, and exact-span match. The five types are PERSON, PHONE, EMAIL, ADDRESS, and NATIONAL_ID. The benchmark is therefore saturated at the type level, and the open questions concern precision on hard negatives (Section VI-C) and context sensitivity (Section VI-F), not per-type recall.

C. Precision on the Negative Suite (Dataset C)

The recall-optimized E4 system reaches a specificity of 74.3%. The E5 precision fixes raise specificity to 94.3% and reduce over-mask on negatives from 14.5% to 1.0%, at no recall cost: Dataset B remains at 100% for all five types and recall on Dataset A is unchanged. Table III reports the per-category false-positive rates.

The culprit-type breakdown attributes the E4 false positives to the greedy location-prefix regular expression (five of nine), the honorific rule firing on คุณ inside ขอบคุณ and ของคุณ (two), an unvalidated IP octet (one), and a single OS-NER organization span. The E5 fixes target exactly these cases, and the two residual false positives are both OS-NER artifacts outside the rule-based components. Because these fixes were designed from the Dataset C errors themselves, the post-fix specificity is a development-set figure, and with 35 negatives its exact 95% interval spans 80.8% to 99.3% (Table VI).

TABLE III. Precision and robustness fixes on the hard suites

Set	Category	N	Before	After
C	NAME_COLLIDE	9	0.0%	0.0%
C	NUMERIC	5	20.0%	0.0%
C	EMAIL_URL	5	20.0%	20.0%
C	DATE_REF	5	20.0%	0.0%
C	BENIGN	5	20.0%	0.0%
C	PLACE_COLLIDE	6	83.3%	16.7%
C	Overall (FP / spec.)	35	25.7% / 74.3%	5.7% / 94.3%
D	MULTI	9	100%	100%
D	NESTED	5	100%	100%
D	ROMANIZED	3	100%	100%
D	ADDR_VAR	5	20%	80%
D	Overall (coverage)	22	81.8%	95.5%

Rows marked C are Dataset C hard negatives, giving the per-category false-positive document rate before (E4) and after (E5) the precision fixes, with overall FP rate / specificity; lower is better. Rows marked D are Dataset D hard positives — MULTI (multiple entities), NESTED (person-in-org), ROMANIZED / foreign, ADDR_VAR (non-template address) — giving coverage before and after the E8a ADDRESS generalization; higher is better.

D. Robustness (Dataset D)

Generalizing the ADDRESS pattern beyond the template form (E8a) raises Dataset D coverage from 81.8% to 95.5% (exact 95% interval 77.2%–99.9%, Table VI), with the gain concentrated entirely in the non-template address category (20% to 80%) and the multi-entity, nested, and romanized categories already at 100% throughout (Table III, rows D). The out-of-vocabulary person probe is covered in all ten cases by the neural NER alone, indicating that the system is not over-reliant on the 21,218-entry name lexicon, a point taken up in Section VII. As with Dataset C, the E8a pattern was designed in response to the Dataset D failures, so this gain is a development-set result rather than an estimate of performance on unseen addresses.

E. Comparison with Baselines

Table IV compares Parivarta with the five baselines on identical inputs and metrics, using the unified scorer, Unicode-scalar offsets, and bootstrap 95% confidence intervals, and including the zero-shot local LLM. Baseline settings are given in Section V-C.

TABLE IV. Baseline comparison: Dataset B coverage and exact-span match (2,500 documents), Dataset C specificity (35 negatives), and Dataset D coverage (22 gold spans)

System	B cov.	B exact	C spec.	D cov.
Generic regex	28.0% [26.2, 29.8]	28.0%	97.1%	18.2%
spaCy (multilingual)	9.8% [8.6, 11.0]	0.1%	42.9%	27.3%
PyThaiNLP (thainer)	27.8% [26.0, 29.5]	17.0%	80.0%	50.0%
Presidio (+Thai regex)	61.2% [59.2, 63.0]	40.2%	77.1%	36.4%
LLM zero-shot (qwen2.5:7b)	60.2% [58.2, 62.1]	58.6%	100.0%	63.6%
Parivarta (ours)	100.0% [100.0, 100.0]	100.0%	94.3%	95.5%

Higher is better throughout. Brackets give bootstrap 95% confidence intervals for Dataset B coverage (10,000 resamples over documents).

Every baseline collapses on multi-type Thai PII. The single neural recognizers, spaCy and PyThaiNLP, miss formatted identifiers entirely and, on Thai, over-fire on common words, with spaCy reaching a specificity of only 42.9%. Presidio, the strongest of the non-LLM baselines, still attains only 61.2% coverage because its recognizers are English-centric even when Thai patterns are added. The generic regular-expression baseline achieves the highest

specificity among the non-LLM baselines (97.1%) precisely because it detects very little, a precision-recall trade that Parivarta avoids by pairing 100% coverage with 94.3% specificity. On 35 negatives, however, the specificity differences among Parivarta, the regular-expression baseline, Presidio, PyThaiNLP, and the LLM are not statistically resolved: their exact 95% intervals overlap (Table VI).

The zero-shot LLM covers only 60.2% of documents, trailing Parivarta by 39.8 points and statistically indistinguishable from Presidio (61.2%), whose bootstrap interval overlaps its own (Table IV). It collapses on the free-text types, reaching just 18.4% on PERSON and 23.6% on ADDRESS, while handling the rigidly formatted EMAIL (90.6%), PHONE (87.6%), and NATIONAL_ID (80.8%) far better. Tellingly, it never produces a false positive on Dataset C, achieving 100% specificity, which indicates that it under-extracts. A paired exact McNemar test on Dataset B coverage confirms that the gap is significant for every baseline: because Parivarta covers every document, no baseline covers a document that Parivarta misses, so Parivarta's coverage is a strict superset of each baseline's, covering an additional 995 documents relative to the LLM, 971 relative to Presidio, and up to 2,256 relative to spaCy. The discordant counts are therefore b = 995, 971, and 2,256 with c = 0, giving continuity-corrected statistics χ^2 = 993.0, 969.0, and 2,254.0 and exact two-sided $p = 2^{1-b} < 10^{-290}$; the regular-expression and PyThaiNLP baselines, whose coverage is lower still, yield larger b and smaller p.

F. Context Sensitivity: The Minimal-Pair Suite (Dataset AMB)

Datasets B through D cannot test the hardest case, in which the same surface string is PII in one context and a look-alike in another. Dataset AMB (Section V-A) isolates exactly this case; each member is scored on the shared candidate span, and pair-accuracy is compared with the independence baseline.

TABLE V. DATASET AMB (1,250 PAIRS): POS-RECALL, NEG-SPECIFICITY, AND PAIR-ACCURACY, WITH THE INDEPENDENCE BASELINE AND THE BALANCED MEMBER ACCURACY, FOR WHICH CHANCE IS 50%

System	POS-rec.	NEG-spec.	Pair-acc.	Indep.	Bal. acc.
LLM (qwen2.5:7b)	42.4% [39.6, 45.1]	75.3% [72.9, 77.7]	26.1% [23.7, 28.6]	31.9%	58.9%
Parivarta (ours)	74.1% [71.6, 76.5]	21.8% [19.4, 24.2]	8.6% [7.0, 10.2]	16.1%	47.9%
spaCy	14.6% [12.7, 16.6]	75.2% [72.8, 77.6]	7.9% [6.5, 9.4]	11.0%	44.9%
Presidio	64.1% [61.4, 66.7]	30.6% [28.1, 33.3]	2.9% [2.0, 3.8]	19.6%	47.3%
PyThaiNLP	4.5% [3.4, 5.7]	61.5% [58.8, 64.2]	0.1% [0.0, 0.2]	2.8%	33.0%
Generic regex	34.7% [32.2, 37.4]	65.3% [62.6, 67.8]	0.0% [0.0, 0.0]	22.7%	50.0%

Rows are ordered by pair-accuracy. Brackets give bootstrap 95% confidence intervals (10,000 resamples; over members for POS-recall and NEG-specificity, over pairs for pair-accuracy).

Three findings emerge. First, Dataset AMB collapses the whole field: the best pair-accuracy is 26.1%, against Dataset B's coverage regime where leading systems exceed 60% and Parivarta reaches 100%, making AMB the one benchmark reported here that no method saturates. Second, surface triggering is universal — every system's pair-accuracy lies below its own independence baseline, so POS- and NEG-correctness are negatively correlated, which is consistent with each system treating the shared surface form identically in both contexts. Only the LLM exceeds chance on balanced accuracy (58.9%), and the generic regular expression sits exactly on the 50% trade-off line with 0% pair-accuracy, because a deterministic surface rule fires on both members of a pair or on neither.

Third, the ranking inverts. The coverage leader, Parivarta, is not the context leader, the LLM, while the coverage laggard, spaCy, rises to third. Coverage and context sensitivity are therefore distinct capability axes. For Parivarta the format layers yield the highest POS-recall in the comparison (74.1%) but the lowest NEG-specificity (21.8%), so its residual error is over-masking of look-alikes and not leakage. That is the privacy-safe direction, and it localizes the next worthwhile investment, namely a context-aware disambiguation stage constrained so that it can only withhold a mask and never remove coverage (Section VII).

Practical consequence of over-masking. A NEG-specificity of 21.8% means that roughly four in five look-alike negatives in Dataset AMB would be masked. Because the mapping is exact, an over-masked span is restored in the reply, so the user's text is not corrupted; the cost is that the model reasons over a placeholder, which degrades answers when the token carries task-relevant meaning, such as a product code or a name-like common word. How often this happens in real prompts is not measured here, since Dataset AMB deliberately over-represents look-alikes; quantifying it and demonstrating a leak-safe context-aware refinement are future work (Section VII-D).

VII. DISCUSSION

A. Principal Findings

Six findings emerge. First, a hybrid, largely deterministic pipeline attains ceiling coverage on structured Thai PII in Dataset B with essentially flat over-mask. Second, the dominant failure mode for formatted PII was surface-format brittleness rather than modeling capacity. Third, boundary quality is separable from detection and is best fixed by policy: precision-aware fusion raised exact-span match from 88.6% to 100% without retraining. Fourth, the segmentation-gated lexicon improved recall and precision together on the development set. Fifth, in a script without word boundaries the precision risk lies in prefix heuristics, and removing a greedy location-prefix rule raised specificity on Dataset C from 74.3% to 94.3%. Sixth, coverage and context sensitivity are distinct axes, and Parivarta's residual error on Dataset AMB is over-masking rather than leakage.

B. Security and Re-identification Analysis

Surrogates are type-tagged counter values bound to their originals only inside the encrypted local vault (Section IV-E), so an honest-but-curious provider cannot invert them, yet still receives the coreference structure it needs. Two residual risks remain. Quasi-identifiers that the detector does not mask still reach the provider, a fundamental limit of span-level masking. Stable pseudonyms reused across turns let the provider link mentions of the same masked entity; where this is unacceptable, per-session key and counter rotation removes it at the cost of cross-session coreference. Reply tampering, device compromise, and non-content side channels remain out of scope (Section III).

Scope of the guarantee. Protection is empirical and conditional on detection: a missed PII value is transmitted in clear, and Section VI bounds that risk only on the benchmarks reported here. Parivarta should therefore be deployed as a risk-reduction control that complements, rather than replaces, organizational measures such as data

minimization and data-processing agreements with the provider under the PDPA [1].

C. Threats to Validity

The primary recall benchmark, Dataset B, is synthetic and positive-only with one entity per document, so it cannot measure precision, and its surface diversity is bounded by the generator's templates. Its 100% ceiling should therefore be read as saturation of the benchmark rather than as perfection in the wild. Precision rests on the hand-authored negative suite of 35 items (Dataset C), and robustness rests on Datasets D and AMB, which establish the locus of errors rather than population-level rates. The comparative claims are therefore claims about these benchmarks: they show that the design closes specific gaps of existing tools on Thai formatted PII, not that Parivarta outperforms them on naturally occurring Thai text. Table VI makes the small-sample uncertainty on Datasets C and D explicit.

TABLE VI. EXACT (CLOPPER–PEARSON) 95% CONFIDENCE INTERVALS ON THE SMALL SUITES, DATASETS C AND D

System	C k/n	C spec. %	95% CI	D k/n	D cov. %	95% CI
Generic regex	34/35	97.1	85.1–99.9	4/22	18.2	5.2–40.3
spaCy	15/35	42.9	26.3–60.6	6/22	27.3	10.7–50.2
PyThaiNLP	28/35	80.0	63.1–91.6	11/22	50.0	28.2–71.8
Presidio	27/35	77.1	59.9–89.6	8/22	36.4	17.2–59.3
LLM (qwen2.5:7b)	35/35	100.0	90.0–100.0	14/22	63.6	40.7–82.8
Parivarta (ours)	33/35	94.3	80.8–99.3	21/22	95.5	77.2–99.9

Computed from the counts underlying Table IV. On Dataset B, Parivarta's 2,500/2,500 coverage has an exact 95% interval of 99.85%–100%, and for any proportion on Dataset AMB (1,250 items per side) the half-width of a 95% interval is at most about 2.8 percentage points.

Entity coverage. T and every evaluation set exclude bank account numbers, house registration numbers, license plates, and hospital numbers (Section II-B). Because the NATIONAL_ID pattern matches any 13-digit number in the 1-4-5-2-1 grouping without checksum validation, 13-digit Thai taxpayer identification numbers are matched by construction, but they were not evaluated separately. No claim is made for these classes.

The span-precision weights that drive fusion are expert-assigned rather than learned; their ordering is justified by diagnostics, but their exact values are untuned. The pipeline is deterministic given fixed weights, so point estimates are reported for the detector and bootstrap 95% confidence intervals for all comparative metrics, complemented by exact binomial intervals for the small suites (Table VI). ADDRESS remains the least generalizable component, as Dataset D shows, and the LLM baseline reflects a single open model, qwen2.5:7b; a different model would likely shift its absolute numbers, though not the qualitative finding on Dataset AMB.

D. Limitations and Future Work

The positive benchmarks are saturated and the negative suite nearly so, so meaningful further evaluation requires harder, larger, and independently annotated data with multiple and nested entities, noisy addresses, and code-mixed names; Dataset AMB is a first, discriminating step. The planned work is as follows.

Independent held-out evaluation. Dataset E will inject synthetic PII into real Thai carrier text with several entities per document, a negative set of several hundred items, and annotation by multiple annotators with reported agreement. The detector will be frozen before Dataset E is built, so that it serves as a genuine held-out test, and every headline proportion will be reported with its confidence interval.

End-to-end evaluation. The full round trip of masking, cloud inference, and restoration will be evaluated on representative tasks such as question answering, summarization, and drafting. The study will compare answer quality for original and pseudonymized prompts, measure the restoration rate including surrogates that the model alters (Section IV-E), and report per-stage and end-to-end latency, model load time, and peak memory on the target devices.

Context-aware refinement. A disambiguation stage that can only withhold masks will be accepted only if it raises NEG-specificity on Dataset AMB without increasing the leak rate on any positive set, following the gate of (14). Learned span-precision weights with calibrated confidence, a learned ADDRESS model for interleaved-token addresses, and filtering of the residual OS-NER noise are further modeling directions.

Entity types and restoration. The type inventory will be extended to bank account numbers, house registration numbers, license plates, and hospital numbers, each with a validated pattern and gold data, restoration will gain variant-tolerant surrogate matching, and the approach will be extended to further languages.

Privacy hardening and reproducibility. Per-session key and counter rotation will be offered for deployments that cannot accept cross-turn linkage, and the detection of common quasi-identifier combinations will be investigated. Dataset E and the end-to-end harness will be made available on the same terms as the materials described in the Data Availability statement.

VIII. CONCLUSION

This article presented Parivarta, an on-device, reversible, Thai-first zero-trust pseudonymization proxy whose hybrid detection ensemble is reconciled by a single precision-aware span-fusion rule and evaluated under a privacy-first protocol that treats a leak as strictly worse than an over-mask.

Through an eval-first ablation, coverage on a 2,500-document benchmark rose from 77.1% to 100% and exact-span match from 65.7% to 100% at near-flat over-mask, with 94.3% specificity on hard negatives and 95.5% coverage on hard positives, figures that are development-set results on small suites. Against five baselines, including a zero-shot local LLM, Parivarta strictly dominates per-document coverage on this benchmark (exact McNemar $p < 10^{-290}$). A new minimal-pair suite then showed that context sensitivity, meaning the ability to distinguish PII from a same-surface look-alike, remains unsolved by all current methods, and that Parivarta's residual error in that setting is over-masking and not leakage, which is the safe direction for a proxy. The main open problems are a context-aware disambiguation stage that can only withhold masks and a generalizable ADDRESS model. Validation on independently annotated real-carrier Thai text, an end-to-end study of utility, restoration, and on-device overhead, and support for further Thai identifier types are the immediate next steps, and the datasets, scorers, and harnesses are publicly available so that they can be pursued reproducibly.

DATA AVAILABILITY

The synthetic evaluation datasets (B, C, D, and the AMB minimal-pair suite), the per-system prediction exports, the standard-library scorer, the baseline runners, and the source

code are publicly available at https://github.com/tluokong/parivarta under the Apache License 2.0 and archived at https://doi.org/10.5281/zenodo.24051056. With these materials, the results in Tables IV–VI can be recomputed from the included prediction exports without network access, except for the LLM baseline on Datasets B and AMB, whose exports must be regenerated with a locally installed model; regenerating Parivarta's own exports requires building the app, whose Thai NER model is converted once by an included script. No real personal data is used or distributed: all PII in the datasets is synthetic.

ACKNOWLEDGMENT

This work was supported by the National Cyber Security Agency (NCSA), Thailand. The authors thank Rajamangala University of Technology Lanna (RMUTL) for research facilities and support, and the anonymous reviewers of KICSS 2026 for constructive comments that improved the clarity and rigor of this paper.

REFERENCES

[1] Kingdom of Thailand, "Personal Data Protection Act B.E. 2562 (2019)," 2019.

[2] European Parliament and Council, "Regulation (EU) 2016/679 (General Data Protection Regulation)," 2016.

[3] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, "Zero trust architecture," National Institute of Standards and Technology, *NIST Special Publication* 800-207, 2020.

[4] 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 Symp.*, 2021, pp. 2633–2650.

[5] N. Lukas, A. Salem, R. Sim, S. Tople, L. Wutschitz, and S. Zanella-Béguelin, "Analyzing leakage of personally identifiable information in language models," in *Proc. IEEE Symp. Security and Privacy (S&P)*, 2023, pp. 346–363.

[6] R. Staab, M. Vero, M. Balunović, and M. Vechev, "Beyond memorization: Violating privacy via inference with large language models," in *Proc. 12th Int. Conf. Learning Representations (ICLR)*, 2024.

[7] C. J. Chong, C. Hou, Z. Yao, and S. M. S. Talebi, "Casper: Prompt sanitization for protecting user privacy in web-based large language models," in *Proc. 2025 IEEE 12th Int. Conf. Cyber Security and Cloud Computing (CSCloud)*, 2025, pp. 122–133.

[8] J. Zhou, E. Xu, Y. Wu, and T. Li, "Rescriber: Smaller-LLM-powered user-led data minimization for LLM-based chatbots," in *Proc. 2025 CHI Conf. Human Factors in Computing Systems (CHI)*, 2025.

[9] L. Siyan, V. C. Raghuram, O. Khattab, J. Hirschberg, and Z. Yu, "PAPILLON: Privacy preservation from Internet-based and local language model ensembles," in *Proc. 2025 Conf. Nations of the Americas Chapter of the ACL (NAACL-HLT)*, 2025, pp. 3371–3390.

[10] Z. Shen, Z. Xi, Y. He, W. Tong, J. Hua, and S. Zhong, "The fire thief is also the keeper: Balancing usability and privacy in prompts," 2024, arXiv:2406.14318.

[11] A. Stubbs, C. Kotfila, and Ö. Uzuner, "Automated systems for the de-identification of longitudinal clinical narratives: Overview of 2014 i2b2/UTHealth shared task Track 1," *J. Biomed. Inform.*, vol. 58, pp. S11–S19, 2015.

[12] F. Dernoncourt, J. Y. Lee, O. Uzuner, and P. Szolovits, "De-identification of patient notes with recurrent neural networks," *J. Am. Med. Inform. Assoc.*, vol. 24, no. 3, pp. 596–606, 2017.

[13] Microsoft, "Presidio: Context-aware, pluggable and customizable PII de-identification," 2020. [Online]. Available: https://github.com/microsoft/presidio

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

[15] C. Patsakis and N. Lykousas, "Man vs the machine in the struggle for effective text anonymisation in the age of large language models," *Sci. Rep.*, vol. 13, Art. no. 16026, 2023.

[16] I. Pilán, P. Lison, L. Øvrelid, A. Papadopoulou, D. Sánchez, and M. Batet, "The text anonymization benchmark (TAB): A dedicated corpus and evaluation framework for text anonymization," *Comput. Linguist.*, vol. 48, no. 4, pp. 1053–1101, 2022.

[17] W. Phatthiyaphaibun *et al.*, "PyThaiNLP: Thai natural language processing in Python," 2023. [Online]. Available: https://github.com/PyThaiNLP/pythainlp

[18] P. Chormai, P. Prasertsom, and A. Rutherford, "AttaCut: A fast and accurate neural Thai word segmenter," 2019, arXiv:1911.07056.

[19] L. Lowphansirikul, C. Polpanumas, N. Jantrakulchai, and S. Nutanong, "WangchanBERTa: Pretraining transformer-based Thai language models," 2021, arXiv:2101.09635.

[20] P. Sriwirote, A. T. Rutherford, J. Thapiang, and V. Timtong, "PhayaThaiBERT: Enhancing a pretrained Thai language model with unassimilated loanwords," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, vol. 24, no. 11, pp. 1–17, 2025.

[21] P. Boonkwan, V. Luantangsrisuk, S. Phaholphinyo, K. Kriengket, D. Leenoi, C. Phrombut, M. Boriboon, K. Kosawat, and T. Supnithi, "The annotation guideline of LST20 corpus," 2020, arXiv:2008.05055.

[22] W. Buaphet, C. Udomcharoenchaikit, P. Limkonchotiwat, A. Rutherford, and S. Nutanong, "Thai nested named entity recognition corpus," in *Findings of the Association for Computational Linguistics: ACL 2022*, 2022, pp. 1473–1486.

[23] N. Carlini, D. Ippolito, M. Jagielski, K. Lee, F. Tramèr, and C. Zhang, "Quantifying memorization across neural language models," in *Proc. 11th Int. Conf. Learning Representations (ICLR)*, 2023.

[24] H. Brown, K. Lee, F. Mireshghallah, R. Shokri, and F. Tramèr, "What does it mean for a language model to preserve privacy?" in *Proc. ACM Conf. Fairness, Accountability, and Transparency (FAccT)*, 2022, pp. 2280–2292.

[25] N. Mireshghallah, H. Kim, X. Zhou, Y. Tsvetkov, M. Sap, R. Shokri, and Y. Choi, "Can LLMs keep a secret? Testing privacy implications of language models via contextual integrity theory," in *Proc. 12th Int. Conf. Learning Representations (ICLR)*, 2024.

[26] K. Greshake, S. Abdelnabi, S. Mishra, C. Endres, T. Holz, and M. Fritz, "Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection," in *Proc. 16th ACM Workshop Artificial Intelligence and Security (AISec)*, 2023.

[27] Y. Chen, T. Li, H. Liu, and Y. Yu, "Hide and seek (HaS): A lightweight framework for prompt privacy protection," 2023, arXiv:2309.03057.

[28] L. Sweeney, "k-Anonymity: A model for protecting privacy," *Int. J. Uncertainty, Fuzziness Knowl.-Based Syst.*, vol. 10, no. 5, pp. 557–570, 2002.

[29] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Proc. 3rd Theory of Cryptography Conf. (TCC)*, 2006, pp. 265–284.

[30] S. Krishna, R. Gupta, and C. Dupuy, "ADePT: Auto-encoder based differentially private text transformation," in *Proc. 16th Conf. European Chapter of the ACL (EACL)*, 2021, pp. 2435–2439.

[31] X. Yue, M. Du, T. Wang, Y. Li, H. Sun, and S. S. M. Chow, "Differential privacy for text analytics via natural text sanitization," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 2021, pp. 3853–3866.

[32] S. Utpala, S. Hooker, and P.-Y. Chen, "Locally differentially private document generation using zero shot prompting," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023, pp. 8442–8457.

[33] M. Hao, H. Li, H. Chen, P. Xing, G. Xu, and T. Zhang, "Iron: Private inference on transformers," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 35, 2022.

[34] N. Boucher, I. Shumailov, R. Anderson, and N. Papernot, "Bad characters: Imperceptible NLP attacks," in *Proc. IEEE Symp. Security and Privacy (S&P)*, 2022, pp. 1987–2004.

[35] Unicode Consortium, "Unicode Technical Report #36: Unicode security considerations," 2014. [Online]. Available: https://www.unicode.org/reports/tr36/

[36] Q. McNemar, "Note on the sampling error of the difference between correlated proportions or percentages," *Psychometrika*, vol. 12, no. 2, pp. 153–157, 1947.