PRIVWEAVE: BENCHMARKING ADAPTIVE SECURE INFERENCE FOR MODERN LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

When language-model services run over private prompts and proprietary weights, secure inference, whether MPC-based, HE-based or hybrid, is the de-facto route. However, our systematic review of 128 records finds 21 of 64 cryptographic LLM-inference systems fix their plan, one is environment-aware, 29 report no network setting, and 40 no offline cost. The key building block is the execution plan: typed operators joined by directional, possibly invalid conversions; no silver-bullet plan exists, since the best tracks installed state and link. We present PRIVWEAVE, a benchmark framework testing when adaptation is admissible and beneficial, with three designs: numerically admitted typed contracts, conversion-aware planning, and stateful session accounting. On 181 held-out configurations, fitted costs cut MAPE 13.05× (298.5% to 22.9%). Against the best executed fixed plan, shaped-loopback sessions show 14.6% lower accounted online cost on an improving link, 1.2% higher on a stable constrained one. Across six models and three contexts, numerical emulation meets the perplexity budget in 15 of 18 cells; only 9 show no observed arithmetic-range violation. Even so, no silver-bullet solution covers multi-host confidentiality, malicious security, universal backend coverage, and cross-model gains simultaneously.

1 INTRODUCTION

Language-model services run over private prompts and proprietary weights, and secure inference from secure multi-party computation (MPC), homomorphic encryption (HE) and hybrids is the de-facto route. Guarantees differ: hiding a prompt, protecting weights and limiting disclosure of a training record need distinct contracts (Dwork & Roth, 2014; Chen et al., 2026c). Given a contract, the problem is composition: assembling a model's operators into a plan that meets it at acceptable cost.

The composition problem. A Transformer mixes linear algebra, normalization, comparisons and autoregressive state, each interacting differently with encryption, secret sharing and preprocessing. Fast kernels thus cannot predict plan cost or validity: boundary costs can reverse kernel rankings, and a cheap conversion can be invalid when the ring, signed interpretation or encryption scheme differs. Decoding is stateful: weight masks are reusable, a key–value cache persists across tokens, and paid setup is not recovered by a lane switch. Figure 1 places PRIVWEAVE in the workflow.

What the literature settles and what it leaves open. Our systematic review codes 128 records and finds three productive families: MPC-based, HE-based, hybrid (Gupta et al., 2024; Zhang et al., 2025a; Pang et al., 2024). Yet 21 of the 64 cryptographic systems fix their plan and one adapts to its environment (Yang et al., 2026); only 11 report an offline cost, 29 report no network setting, and none is evaluated over a multi-request session (Section 2).

Our framework: PRIVWEAVE. We present PRIVWEAVE, a benchmark framework for security-compatible adaptation in Transformer inference, with three designs. The key idea is to describe a plan as operator realizations, directional arithmetic-domain conversions, numerical contracts and state transitions, and to admit each element against the fixed security contract before any cost is compared: PRIVWEAVE types every plan, prices each directional boundary, checks feasibility and switches statefully, charging installation before crediting the predicted gain.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

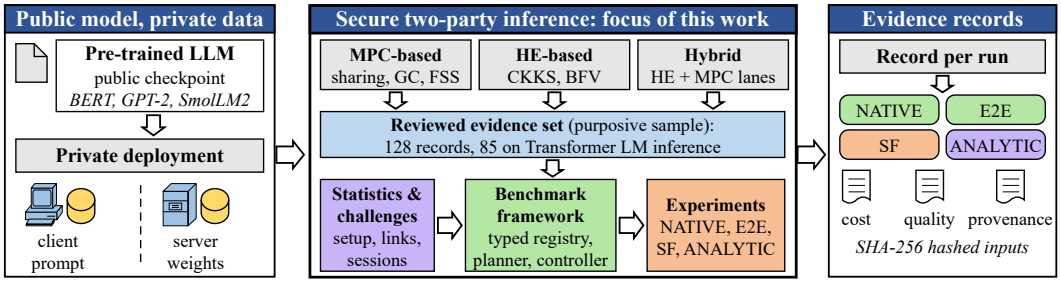


Figure 1: Secure LLM inference and the focus of this work. A public pre-trained LLM runs over a private client prompt and private server weights (left). Secure two-party inference is MPC-based, HE-based or hybrid; PRIVWEAVE reviews 128 records, 85 of which evaluate Transformer language-model inference, and fixes a benchmark framework (centre). Every run yields a record carrying cost, quality, provenance and evidence class (right).

Table 1: Per-family summary of the 64 cryptographic secure-inference systems, from the reviewed corpus (purposive sample, not prevalence). SH: semi-honest only; +dealer: two parties with a dealer; Open: open-source release; ≥1B: largest reported model at least one billion parameters. Nonlinear and adaptivity give the two most frequent codes (co-design: model-level replacement; LUT: lookup table or function secret sharing; polynomial: polynomial or iterative approximation; fixed: one static plan; manual: hand-mixed protocols). Network and offline omit not-applicable and unclear rows.

Family	n	Coded properties[†]												
		SH	Parties			Network			Offline		Nonlinear	Adaptivity	Open	≥1B
		only	2PC	+dealer	3PC	none	LAN	both	yes	no	(top two)	(top two)	source	params
MPC-based	20	17	3	11	6	3	9	8	5	13	co-design 7 LUT 6	co-design 8 fixed 6	15	9
HE-based	22	22	22	0	0	20	0	1	0	12	polynomial 10 co-design 9	co-design 12 fixed 7	13	9
Hybrid	22	21	15	4	1	6	4	11	6	15	polynomial 8 co-design 5	manual 9 fixed 8	9	8

[†] Per-record coding for all 128 records: Table 4, Appendix A.

Results. Fitted costs cut held-out mean absolute percentage error from 298.5% to 22.9% over 181 operator configurations (13.05× lower error, not a speedup). Against each trace's best executed fixed plan, adaptive switching over shaped loopback saves 14.6% accounted online cost on an improving link and loses 1.2% on a stable constrained one. Numerical emulation over six causal LMs and three contexts meets the perplexity budget in 15 of 18 settings, only 9 with no observed range violation. A boundary audit exposes a missing correctness condition in the supplied ring-to-BFV routine.

Contributions. We (i) review 128 records and code their families, parties, reporting and adaptivity; (ii) define a typed admission and conversion contract that makes plans comparable; (iii) build the benchmark and its executable harness; and (iv) measure two new campaigns locally and ship drivers for strata needing a server. Three questions follow: **RQ1**, which comparisons survive matched security, representation, and setup accounting; **RQ2**, how calibration and conversion constraints shape candidate plans; **RQ3**, when a causal policy beats fixed execution over a trace.

2 A SYSTEMATIC REVIEW OF SECURE INFERENCE FOR LANGUAGE MODELS

We review the field under one protocol with five facets: privacy mechanism, workload, task framing, component and evidence type, combined pairwise over security, cryptography, learning and systems venues plus arXiv, cut off on 11 September 2026 (inclusion and exclusion rules: Appendix A). Of 197 candidates, 133 entered full-text verification and 128 are included: 85 Transformer-inference systems, 64 of them cryptographic. Three families partition those 64: 20 MPC-based, 22 HE-based and 22 Hybrid; Table 1 codes their threat model, parties, reporting, nonlinear method, adaptivity, open source and model size. Forty are two-party without a dealer, 15 with a dealer, 7 three-party and 2 single-party (Figure 2(e)); 60 of 64 are semi-honest only (g) and 37 release open source (h).

MPC-based systems pay in rounds. They carry the arithmetic in secret sharing, garbled circuits or function secret sharing (FSS): MPCFormer distils an MPC-friendly network, PUMA approximates nonlinearities in an unchanged one, and SIGMA puts GELU and softmax into FSS keys (Li et al., 2023; Dong et al., 2025b; Gupta et al., 2024). Every nonlinearity costs rounds, and FSS, an MPC technique, not a separate family, shifts awkward gates into offline correlated randomness (Boyle et al., 2019; Gupta et al., 2022; Jawalkar et al., 2024), which someone must price (C3, C4).

HE-based systems pay in multiplicative depth. They evaluate the arithmetic homomorphically at one server, converging on RNS-CKKS with bootstrapping (Zhang et al., 2025a; Moon et al., 2025; Zhang et al., 2026b). Where evaluation is non-interactive the price is depth, not rounds: every non-polynomial operator becomes a polynomial whose degree sets the bootstrapping budget, searched per layer by ATLAS (Xie et al., 2026). This is also why 20 of the 22 report no network setting.

Hybrid systems pay at the boundary. Hybrid systems combine two or more of MPC, HE, a trusted execution environment and permutation: HE linear layers with MPC nonlinearity (Pang et al., 2024; jie Lu et al., 2025; Xu et al., 2025), an enclave with blinded linear algebra (Tramèr & Boneh, 2019), and permuted data with exact plaintext nonlinearity (Yuan et al., 2026; Luo et al., 2025). Each conversion and truncation adds traffic the kernel saving must repay (C1). The field also changed subject (Appendix Figure 5(a)): MPC-based systems arrive in 2023, HE-based and Hybrid lead the cryptographic families in 2025 and 2026, and 85 of 128 records postdate 2023.

Observation-1: execution plans are fixed before the deployment is known. Figure 2(a) codes how each plan is chosen: 21 of 64 are fixed, 21 co-design the model then fix it, 11 mix protocols by hand, 5 adapt to the input, 4 select at compile time, one is environment-aware (Yang et al., 2026) and one dispatches on type (Wu et al., 2024). Adaptivity is an interpretive code over a purposive sample; a benchmark must therefore score a plan against a deployment, not the proposing paper (C8).

Observation-2: under-reported environments and offline phases decide the ranking. Twenty-nine of 64 report no network setting and 13 LAN only (Figure 2(b)), and the share reporting any setting falls from 80% in 2023 to 25% in 2026 (Appendix Figure 5(c)); 20 of 22 HE-based rows report none, NEXUS alone both (Zhang et al., 2025a). This is because the ranking flips: MPC wins in LAN, HE can overtake it under low bandwidth and high latency (Ng & Chow, 2023; Chen et al., 2026c), so the likeliest WAN winner omits it. Forty report no offline cost and 11 do (Figure 2(c)), disproportionately dealer-based FSS systems (Gupta et al., 2024; Cheng et al., 2025; Gupta et al., 2025; Ma et al., 2026), whose dealer phase is impossible to hide. Note that "none-reported" means

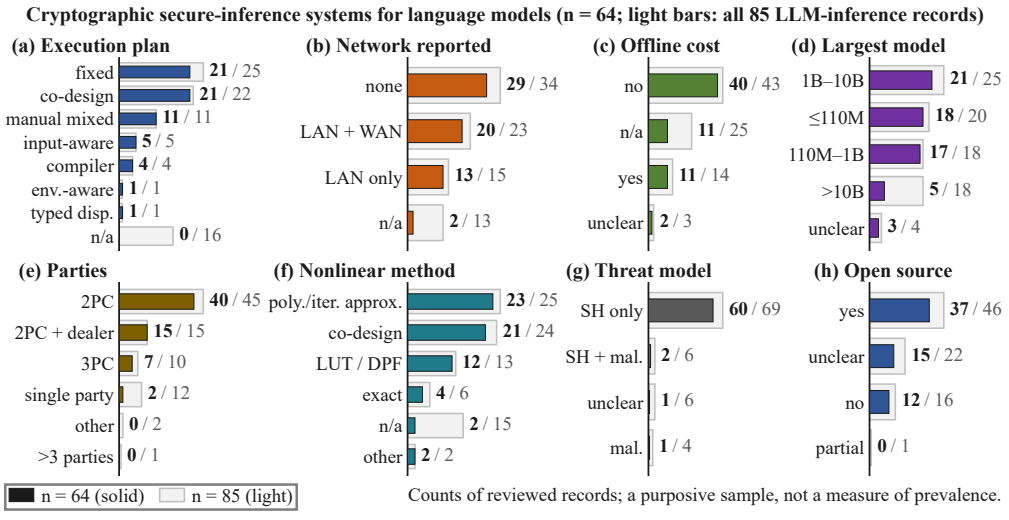


Figure 2: Properties of the 64 cryptographic secure-inference systems for language models (solid bars, bold counts) against all 85 LLM-inference records (light bars, grey counts): (a) execution plan, (b) network setting reported, (c) offline cost reported, (d) largest model evaluated, (e) parties, (f) nonlinear method, (g) threat model (SH: semi-honest, mal.: malicious), (h) open-source release. Rows ordered by the 64-count; descriptive counts over a purposive sample, not prevalence.

the setting is unstated, not that the system ran on one host, and that 11 rows are not-applicable, a non-interactive HE pipeline having no offline phase in the MPC sense. A benchmark must therefore hold one bandwidth and round-trip grid over every family, report rounds beside bytes, split offline from online cost, flag single-use material and price each system without its dealer (C3, C4, C5).

Observation-3: approximations are validated on the task score, not on protocol ranges. Twenty-three of 64 approximate nonlinearity polynomially or iteratively and 21 replace it at model level (Figure 2(f)). Several bound the interval: SHAFT clips softmax inputs, SPRINT caps logits and clipping outliers cuts a SiLU polynomial from degree 83 to 31 (Kei & Chow, 2025; Capano et al., 2026b; Park et al., 2026b), yet none reports how often real inputs leave it. A benchmark must thus log range violations and worst-case absolute error in the run producing the latency number (C2).

Observation-4: evaluation stops at small models and single passes, leaving decoding untested. The measured ceiling is small (Figure 2(d)): 5 of 64 exceed 10B parameters (18 of all 85 records do; the other 13 are TEE, DP or analysis rows) and the only 70B row is permutation-based (Yuan et al., 2026), while hardware-isolated serving runs 70B and 175B routinely (Chrapek et al., 2025; Tan et al., 2025). "Largest model" is what the paper evaluates; some headline figures are accuracy-only or reported without a measurement (Zimerman et al., 2026). Most of the corpus measures one forward pass at a fixed length, so decoding and KV-cache state sit outside the envelope; the exceptions report persistent encrypted caches, cache eviction, split prefill and decode, or per-token cost (Zhang et al., 2026a; Zeng et al., 2025a; Chen et al., 2026a; Zheng et al., 2024; Yuan et al., 2026). A benchmark must therefore separate prefill from per-token decode across prompt and generation lengths, with cache growth and the largest completing configuration (C6, C7).

Comparison conditions imposed by each mechanism. Three consequences follow (Appendix Table 5). The mechanisms protect different observations, so no scalar privacy score exists. Fast kernels do not compose, because conversion and truncation are paid at every boundary (Xu et al., 2025; Gupta et al., 2024). Rigidity belongs to pipelines, not to primitives: the mixed-protocol frameworks already select representations at compile time, per operator or per input (Büscher et al., 2018; Wu et al., 2024; Zhang et al., 2025c), so whether a change pays over a stateful session stays open. Note that attacks on exposed shuffled activations do not establish a break of every permutation-based protocol (Thomas et al., 2025; Li et al., 2026).

Implications. No work measures Transformer secure inference across families under matched conditions: the closest prior art finds 47.5% of artefacts executable (Chen et al., 2026c), and the two benchmark-class records stop at CNN scale (Njungle et al., 2025; Taubert et al., 2026). Four requirements follow, becoming the design of Section 3: match the contract before the cost, labelling the heuristic tier that prompt-reconstruction and weight-extraction attacks reach (Thomas et al., 2025; Li et al., 2026); make conversion, truncation and setup line items, not a total; fix one environment for every family; and score a session, not a query, so a mid-session change of plan counts.

3 PRIVWEAVE DESIGN

Overview. PRIVWEAVE separates three objects: a specification of what a plan must declare, the analytic planner that chooses among admitted plans, and a narrower executable harness that measures what runs. The idea is to record capability and provenance before comparing costs, because an adapter description is no evidence that its conversions or security properties are implemented.

Security-conditioned comparison. A request fixes model, workload and contract S: parties and owners; corruption sets; passive or active behavior; protected inputs, weights and outputs; authorized leakage; setup and hardware trust; security parameters; correctness/abort semantics. Admission is a predicate $\mathsf{Admit}(a, S)$ over instantiated operators and conversions, not a scalar "security score", and the contract holds for the session. DP adds output privacy with budget accounting, TEE adapters hardware and attestation assumptions, ZK adapters a relation and verification cost; these coordinates define strata that compose only with compatible contracts. A registry check is a necessary integration condition, not a cryptographic composition proof.

Typed representations and directional conversions. Each tensor carries a type $d = (\mathsf{representation}, \mathsf{owners}, \mathsf{ring}, \mathsf{scale}, \mathsf{layout}, \mathsf{parameters})$. Parameters include authentication status and value bounds for arithmetic shares, level, scale, key identity and error bounds for CKKS, and the plaintext modulus for BFV/BGV. Boolean shares, garbled-wire labels and replicated shares are distinct types. An FSS key shares a *function*, not a tensor: the gate consumes masked public arguments and correlated keys, returning arithmetic shares, so FSS is an MPC technique that moves work into correlated randomness, not a universal value representation reached by free conversions.

Admissible conversions form a directed multigraph whose edges declare preconditions, output type, leakage, numerical error, preprocessing and measured cost. A route through CKKS, BFV or another ring must discharge its encoding, lifting and key requirements; an identical "HE" label is insufficient. Figure 3(b) draws the harness multigraph over client-plain, CKKS, arithmetic shares, Boolean shares and BFV: encryption and H2A (a plaintext mask with at most 20 bits of slack) are executed, two A2B edges and one B2A edge are specified only, and A2H is rejected. Our audit supplies the case: the arithmetic-to-BFV routine fails on valid signed-wrap inputs such as $s_0 = s_1 = 2^{63} - 1$ encoding $x = -2$ (inset), because it reduces each signed $\mathbb{Z}_{2^{64}}$ share independently modulo a prime while their signed sum wraps by $\pm 2^{64}$. Its BFV output also has no CKKS consumer, so we exclude it from statements of general exact conversion (Appendix C). Figure 3(a) and (c) show the type tuple and the FSS gate; Appendix B expands the layers.

Conversion-aware planning. For a serialized schedule of $G = (V, E)$ with separately materialized edge conversions, the modeled cost is

$$\widehat{C}(\pi; \theta) = \sum_{v \in V} \widehat{c}_v(a_v; \theta) + \sum_{(u,v) \in E} \widehat{t}_{uv}(d_u, d_v; \theta), \tag{1}$$

where θ contains public bandwidth, RTT, hardware and tensor shapes. Each conversion direction is counted separately, as are setup, offline generation, online execution and persistent material; an amortized objective must state its reuse horizon. Shared conversions, parallel scheduling and live-tensor memory introduce dependencies beyond Eq. (1). For a chain with Markovian costs and admitted sets $\mathcal{A}_i$, let $F_i(d)$ be the cheapest prefix of type d; the recurrence is

$$F_i(d) = \min_{\substack{a \in \mathcal{A}_i : \mathrm{out}(a) = d \\ d'}} \left[F_{i-1}(d') + t_i(d', \mathrm{in}(a)) + c_i(a) \right]. \tag{2}$$

Infeasible conversions cost infinity; input/output boundaries are included. A constrained additive error proxy requires retaining nondominated (cost, proxy) prefixes per state, whose number need not be polynomial. The planner implements this exact frontier for its *coarse serialized-chain objective* only: not a general DAG, HE-level evolution, global memory, or an end-to-end accuracy certificate. A sum of local proxies does not bound perplexity, so PRIVWEAVE requires independent task-quality evaluation or a valid composed numerical certificate before declaring accuracy feasible.

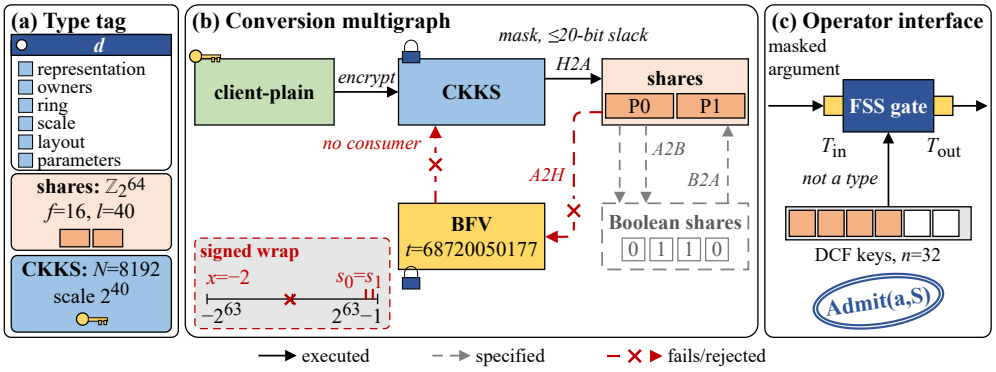


Figure 3: Typed domains and conversions. (a) Type tag d: six slots, two instantiated tags (shares over $\mathbb{Z}_{2^{64}}$, $f = 16$, $l = 40$; CKKS, $N = 8192$, scale 2^{40}, client key). (b) Conversion multigraph (padlocks: ciphertexts; shares: arithmetic, halves $P_0 \mid P_1$). Solid edges are executed (H2A: plaintext mask, slack at most 20 bits), grey dashed edges specified only, red dashed edges fail or are rejected: A2H wraps on $s_0 = s_1 = 2^{63} - 1$ (inset) and its BFV output has no consumer (Appendix C). (c) Operator interface: an FSS gate whose DCF keys (reel) are material, not a type; $\mathsf{Admit}(a, S)$ gates admission.

Stateful switching. At token boundary t the controller sees only the public telemetry prefix, tracks installed material and past dwell lengths, and evaluates admitted challengers q to plan p. The complete switching charge is $K_t = K_{\text{setup}} + K_{\text{install}} + K_{\text{migrate}} + K_{\text{replenish}} + K_{\text{control}}$. Spent preprocessing is reported as waste, not charged twice; sensing is common overhead, not switch cost. For predicted remaining benefit $\widehat{G}_t(p, q)$, uncertainty allowance U_t and hysteresis $h \geq 0$, the rule is

$$\widehat{G}_t(p, q) > (\widehat{K}_t(p, q) + U_t)(1 + h), \tag{3}$$

subject to a minimum residency. The challenger maximizes predicted net benefit, not the next token's speed, and cannot see future trace values or secret activations. Publishing telemetry is an explicit leakage decision; authentication alone does not make a data-dependent timing signal independent of secrets. A constant-regime break-even proposition (Appendix C) fixes the first profitable horizon at $\lfloor K/(c_p - c_q) \rfloor + 1$ epochs, but guarantees nothing under incorrect cost or dwell predictions; the uncertainty allowance is heuristic, not a calibrated confidence bound.

Current implementation. The native harness considers six plans: three embedding lanes $\times$ two layer profiles. In the HE embedding lane, CKKS embedding accepts a client-owned token; subsequent layers and the KV cache stay arithmetic-shared with Beaver/FSS-assisted gates. Admission rejects HE linear layers on arbitrary shared intermediates. The controller charges missing weight-mask installation in its decision and measures sensing and planning separately. KV migration is zero by construction; discarded preprocessing is recorded, but its regeneration is outside online time. Eq. (3)'s full accounting is therefore a specification that the executed decision instantiates in its installation-cost case. The harness runs `SH-2PC` (semi-honest two-party) as two OS processes on shaped loopback TCP; the dealer is emulated in-process from a common seed. These runs support protocol-cost and controller measurements, not confidentiality against a party inspecting its entire process. Separate-party dealer provisioning, arbitrary state migration, malicious security and full cross-paradigm adapters remain unvalidated extensions.

4 BENCHMARK PROTOCOL

Section 3 fixes what a plan is; five rules fix what a record must declare before comparison.

Only matched records are comparable. PRIVWEAVE indexes every record by checkpoint and tokenizer revision, input/output lengths, batch, prefill/decode phase, numerical format, security contract, protocol parameters, hardware, topology, bandwidth and RTT. Records compare only on equal declared contract and workload, and within a stratum latency, communication, memory and task quality remain distinct Pareto coordinates.

Amortize only what is reused. For N served requests, the benchmark records

$$C_{\text{request}}(N) = \frac{C_{\text{compile}} + C_{\text{keygen}} + C_{\text{reusable}}}{N} + C_{\text{one-time}} + C_{\text{online}}. \tag{4}$$

Only reusable setup divides by N: a mask or triple consumed once stays a per-use cost. Time to first token covers the declared request boundary, prefill, and first-token selection; time per output token covers subsequent tokens. Offline time, bytes, rounds, peak memory and throughput are reported separately; shaped links describe the shaper, not a WAN.

Numerical feasibility precedes ranking. Every approximation specifies its domain, clipping, scale, truncation rule and tolerated overflow or wrap behavior. The benchmark reports tensor error and task metrics on the same checkpoint, tokenizer and evaluation slice, and a predeclared joint accuracy/perplexity criterion filters configurations before Pareto ranking. A small local error alone is insufficient: an approximate nonlinearity can accumulate across layers and decoding steps.

Adaptation is scored against declared controls. The required controls are admissible uniform backends, a manual hybrid, a greedy selector, a conversion-aware fixed plan, a causal adaptive policy and the hindsight trace-optimal fixed plan. An unavailable uniform route is recorded as such, without invented time; a clairvoyant, zero-switch-cost oracle is only a lower bound. Normalized oracle regret is $(C_{\text{plan}} - C_{\text{oracle}})/C_{\text{oracle}}$ for a static instance and, for changing sessions, the signed total-cost difference from the best executed fixed plan, counting control, installation, migration and material costs once. Current session records support only the narrower accounted-online-time comparison.

Each result declares its evidence and uncertainty. It records an origin (author artifact, local reimplementation, or reported-only) and an execution scope (analytic, functional, microbenchmark, local protocol, or deployed secure model). An emulator score does not establish secure task accuracy. Confidence intervals never treat correlated operators or tokens as independent trials; this reanalysis uses descriptive statistics. Appendix D specifies the prospective multi-seed, cross-model, cross-hardware evaluation.

5 EXPERIMENTAL SETUP

Host and harness. Native runs execute as two processes over shaped loopback TCP on two Ubuntu 22.04 LTS hosts per campaign (frozen campaigns: Core Ultra 9 285K, 24 logical CPUs; blocks added here: Core i9-14900K, 32 logical CPUs). One host is equipped with an RTX 4090, and the other is equipped with an RTX 6000 Ada Generation GPU, under in-process dealer emulation, a common seed and profile `SH/M-2PC`. The protocol path is CPU-only. Inherited are 876 operator timings, 135 emulation runs and 35 sessions over 280 epochs. This revision adds the three completed blocks of the campaign status record (Appendix E): 54 causal-language-model and 18 classifier evaluations over nine checkpoints, three seeds each, and a session replicate block that finished all 20 four-layer cells under both dealer seeds (40 runs); headline cells keep the frozen single run. Replicate runs share their host with light editorial workloads, so we compare ordering and sign across blocks, never absolute seconds.

6 INITIAL BENCHMARK FINDINGS

Under Section 4's protocol we reanalyze three frozen campaigns with SHA-256 provenance and add the blocks of Section 5. NATIVE/E2E measure protocol cost under one harness, not process-isolated deployment confidentiality, and SF replaces arithmetic locally without cryptography; these are initial findings, not a matched ranking of independent frameworks.

Finding 1: calibration reduces prediction error, but residual error remains. The operator campaign contains 876 timings, three repeats of 292 configurations, split 111/181 by holding out two matrix shapes, WAN-mid and row count 32, but no unseen hardware; errors use per-configuration means. Held-out MAPE falls from 298.52% (handwritten) to 22.87% (fitted), a $13.05\times$ reduction (Appendix Figure 11). The median falls from 79.62% to 8.26%, but the fitted 95th percentile remains 77.50% and the robust predictor has 30.94% mean error. All 252 HE timing rows consume client-owned plaintext and record zero A2H time, so they cannot admit HE on shared activations: cost accuracy does not imply domain compatibility.

Table 2: Experiment axes and evidence primitives. Status: ✓ measured, △ constrained measured, ○ equivalent emulated. Where names the table or figure reporting each measured axis, and Appendix E the driver of each one.

Axis	Status	Evidence[†]	Where
Workload	✓ measured	contexts 64–1024, emulated; native GPT-2 sessions	Table 3, Table 7
Scale	△ constrained measured	48.9 GB per process for gpt2-xl, from GPT-2's measured 14.2 GB; awaits checkpoint download	Table 6
Execution	✓ measured	shaped loopback TCP, two processes, dealer emulated in-process	Table 3
Network	✓ measured	LAN and WAN-constrained; 5 four-layer traces, 1 at full depth	Table 3, Figure 10
Isolation	○ equivalent emulated	physical two-host link tested; shaper only	Appendix D∼E
Hardware	○ equivalent emulated	same-host fit only: 22.9% held-out MAPE, 7.1–35.8% per linear lane	Figure 11
Accounting	✓ measured	accounted online time: execution, sensing, planning, switch	Table 3, Figure 10
Quality	✓ measured	16 fractional bits, GELU degree 12; 72 emulated evaluations, three rounding seeds per cell	Figure 4, Table 7
Statistics	✓ measured	two seeds per replicate cell (40 runs over 20 cells); frozen headline cells single-run	Table 3, Table 6
Malicious security	✓ measured	two active-security capabilities; Cond-Mal-1 admitted by the gate	Appendix E
External frameworks	○ equivalent emulated	local proxy; each framework has its own toolchain	Appendix E

[†]Full evaluation matrix with every driver script: Table 6, Appendix D; drivers of the axes: Appendix E.

Finding 2: a task score alone does not certify numerical admissibility. The source-declared SF grid spans three models, five fractional-bit settings, three degrees and three rounding seeds: 135 runs. BERT uses 256 SST-2 examples; GPT-2 and SmolLM2-135M use 2,046 scored WikiText-2 tokens. Budgets are at most 0.5 percentage-point accuracy loss and +2 perplexity; seeds vary rounding on fixed data, not test sampling. Figure 4(a) to (c) shows ten of 15 BERT and eight of 15 GPT-2 settings meeting their score budget, versus zero SmolLM2. All ten BERT settings are clean (budget met without an observed reciprocal-square-root range or comparison-domain violation), but four score-passing GPT-2 settings record range violations, leaving four clean. All 15 SmolLM2 settings violate a range, eight of them nonfinite, and no setting meets all three budgets.

At the engine operating point of 16 fractional bits and degree 12, BERT changes from 92.188% to 91.797% (-0.391 percentage points) and GPT-2 from 23.2367 to 23.2781 perplexity, i.e., $+0.04143 \pm 0.00075$ across seeds (mean $\pm$ sample standard deviation); SmolLM2 still fails. An extended range/scaling contract passes it by emulation, validating neither the original contract nor native coverage (Appendix F); an absent violation still does not prove validity for every input.

The gap recurs in three of six models and at every context length. The accuracy-extension block (Table 7) holds that operating point over three seeds and six causal LMs at contexts 64, 256 and 1024 (Figure 4(d)). Of the 18 cells, 15 meet the +2 perplexity budget, but only nine are clean. SmolLM2-135M is nonfinite everywhere and violates both the reciprocal-square-root range and the comparison domain; TinyLlama and pythia-160m always pass the score but violate that range and that domain respectively. GPT-2, Qwen2.5-0.5B and Llama-3.2-1B pass cleanly; the largest increase is 1.0517 perplexity (Llama-3.2-1B, context 64). In panel (e), three SST-2 classifiers at contexts 64 and 128 pass the 0.5-point budget in all six cells without a violation (BERT loses 0.391 points at both contexts, DistilBERT and RoBERTa none).

Finding 3: installation can reverse the fastest execution choice. The 35 frozen sessions cover five traces on a four-layer GPT-2 prefix and one on all 12 layers, under fixed, dynamic and ablated policies, with a public 16-token prompt and eight output tokens. P1 keeps CKKS embedding on client-owned IDs over weight-mask MPC layers; P2 is weight-mask MPC for both. KV state stays arithmetic-shared at zero migration cost; installation is charged at the switch epoch (Appendix F).

Installation, not execution speed, separates P2 from adaptive on four-layer WANc-to-LAN (Table 3, Appendix Figure 10): P2 executes faster (245.29 versus 320.55 seconds) but pays more installation (434.63 versus 193.26). Adaptive totals 516.30 seconds, 14.58% below best fixed P1, and 5.58% at

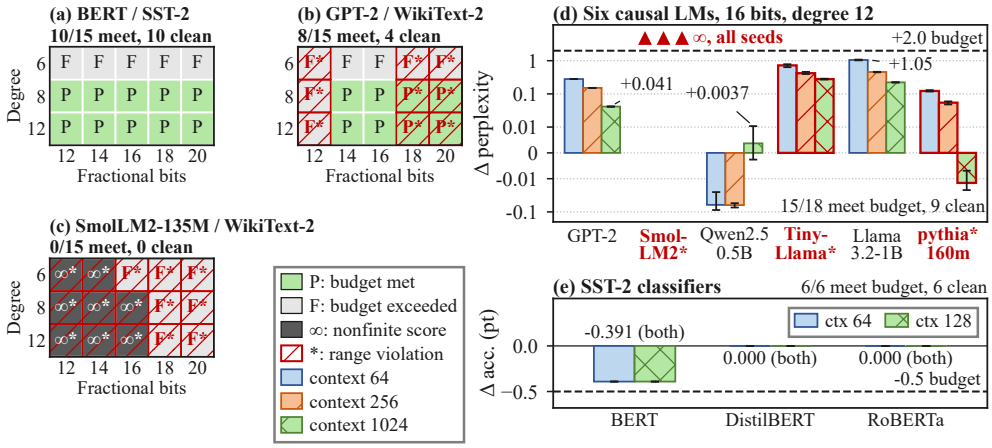


Figure 4: SF emulations, not encrypted inference. (a) to (c): budget outcome per fractional-bit and degree setting (three rounding seeds): P meets the budget (0.5 accuracy points, +2 perplexity), F exceeds it, ∞ is nonfinite; red hatch and asterisk mark an observed range violation; "clean" counts P cells without one. (d) Six causal LMs at 16 bits, degree 12: mean perplexity change (emulated minus plaintext, symlog axis) per context; whiskers span the three seeds (descriptive); red edges and names mark a violation; SmolLM2-135M is nonfinite. (e) Three SST-2 classifiers, contexts 64 and 128: mean accuracy-point change, 0.5-point budget.

Table 3: E2E accounted online totals, excluding dealer provisioning, common LM-head setup and a small timing residual (Appendix F). Exec. and Install: two of Adaptive's four timers; sensing plus planning (0.11 to 4.86 s) is the rest. Δ: Adaptive reduction against the better of P1 and P2 (negative: regression); bold: lowest of the four frozen totals. One run per frozen cell; replicate cells: mean $\pm$ sample s.d. over two seeds, descriptive, no significance test. WANc: WAN-constrained.

		Frozen campaign (s)†					Adaptive timers (s)		Replicate block (2 seeds, s)		
Trace	**L**	**P1**	**P2**	**Naive**	**Adaptive**	**Δ (%)**	**Exec.**	**Install**	**Best fixed**	**Adaptive**	**Δ (%)**
Stable LAN	4	214.38	69.78	68.64	**67.20**	3.69	52.85	14.24	65.50 ± 6.45 (P2)	65.91 ± 6.18	-0.64
Stable WANc	4	**752.91**	843.14	842.80	762.11	-1.22	572.08	185.16	784.73 ± 2.44 (P1)	769.68 ± 5.52	1.92
LAN $\rightarrow$ WANc	4	386.66	225.51	224.83	**224.64**	0.39	207.67	14.45	226.12 ± 1.93 (P2)	227.70 ± 0.20	-0.70
WANc $\rightarrow$ LAN	4	604.43	682.39	687.90	**516.30**	14.58	320.55	193.26	618.50 ± 10.19 (P1)	520.67 ± 5.39	15.82
Fluctuating	4	372.17	225.28	225.19	**223.10**	0.96	206.01	14.62	225.97 ± 1.05 (P2)	225.75 ± 0.37	0.10
WANc $\rightarrow$ LAN	12	1361.40	1457.83	–	**1285.39**	5.58	718.90	563.99	–	–	–

† The remaining frozen policies (P3, delayed replan, three ablations) and all replicate runs are analysed in Figure 10 and Appendix F.

full depth; it is 1.22% slower on stable WANc, so adaptation is not uniformly better. Naive replan ignores installation when selecting but pays it: adaptive is 0.09 to 2.10% below it on the other three four-layer traces, 9.57% below on stable WANc and 24.95% below on WANc-to-LAN; delayed replan is slower on every four-layer cell (87.26 to 1458.24 seconds). Each frozen cell is one run, and matching float/emulated tokens is a narrow check, not dataset-level accuracy.

Finding 4: margins of a few percent do not survive a change of host and seed. The session replicate block repeats the four-layer matrix under two dealer seeds, 20260901 and 20260902, on the 32-logical-CPU host: 20 cells, 40 runs, all complete (Table 3, right). Host (32 against 24 logical CPUs) and seed (against 20260821) change together, so magnitudes are not comparable with the frozen block. On two-seed means the WANc-to-LAN advantage survives: adaptive takes 520.67 ± 5.39 seconds against 618.50 ± 10.19 for the best executed fixed plan P1, a 15.82% reduction against 14.58% frozen. Three margins reverse sign: stable LAN turns 0.64% slower than P2 (65.91 versus 65.50 seconds) and LAN-to-WANc 0.70% slower than P2 (227.70 versus 226.12), both smaller than the seed-to-seed spread of the fixed plan in their cell, while stable WANc turns 1.92% faster than P1 (769.68 versus 784.73). Fluctuating shrinks to 0.10% faster, against 0.96% frozen. Replicated traces, isolated multi-host runs and matched implementations therefore remain necessary.

7 Discussion and Limitations

Answers to RQ1–RQ3. RQ1: comparisons survive only within a matched contract and representation stratum; cost accuracy alone admits nothing (Finding 1), and the corpus agrees (Section 2). RQ2: fitted costs shrink prediction error; range checks prune score-passing settings and none passes all three models' budgets (Finding 2). RQ3: a causal policy beats the best executed fixed plan only when gains recover installation; it can lose (Finding 3).

Limitations. Within the observations and benchmarks discussed in this paper, implementations and deployments under resource-constrained environments cannot fully obtain and reproduce performance matched to the real-world deployments of respective schemes. Four conditions bound interpretation: (1) dealer trust model; (2) restricted HE input domain; (3) one frozen run per session cell, two replicate seeds (Finding 4); (4) restricted task slices.

Conclusion. PrivWeave organizes adaptive secure inference around security contracts, directional conversions and stateful execution: cost prediction needs a separate admissibility check, quality on one checkpoint can fail on another, and a switch must recover its installation cost.

AI USE STATEMENT

Generative AI assisted with code writing (syntax checking, baseline code reproduction, code architecture optimization), prose grammar checking, reference formatting, paper consistency validation, table formatting, and reference authenticity verification. Empirical values are reanalyzed from supplied records, measured on experimental hosts under shaped loopback, and measured on physical two-host deployments. The authors verified all statements, the bibliography and this disclosure, and take full responsibility for the submitted work.

ETHICS STATEMENT

The study treats output privacy, cryptographic confidentiality, hardware trust, and integrity as distinct guarantees. It uses existing benchmark records and fixed evaluation slices. Permutation-based methods and attacks are discussed under their stated adversary assumptions. No sensitive user prompts and no new human-subject data were collected.

REPRODUCIBILITY STATEMENT

The source package contains the evidence ledger, source URLs, hashed inputs, the 128-entry bibliography with per-record verification confidence, experiment drivers (Appendix E), and editable diagram sources. Appendix E specifies reproduction procedure and execution scope. Historical runs were audited through retained records; charts rebuild from copied inputs. The ICLR 2027 style files are unmodified.

REFERENCES

Yoshimasa Akimoto, Kazuto Fukuchi, Youhei Akimoto, and Jun Sakuma. Privformer: Privacy-preserving transformer with MPC. In *2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P)*, pp. 392–410, Delft, Netherlands, 2023. IEEE. doi: 10.1109/EuroSP57164.2023.00031.

Davide Andreoletti, Alessandro Rudi, Emanuele Carpanzano, Francesco Lelli, and Tiziano Leidi. Privacy-preserving LLM inference in practice: A comparative survey of techniques, trade-offs, and deployability. Cryptology ePrint Archive, Paper 2026/105, 2026. URL https://eprint.iacr.org/2026/105.

Ahmad Al Badawi, Andreea Alexandru, Yuriy Polyakov, and Vinod Vaikuntanathan. SoK: Private LLM inference using approximate homomorphic encryption. Cryptology ePrint Archive, Paper 2026/935, 2026. URL https://eprint.iacr.org/2026/935.

Elette Boyle, Niv Gilboa, and Yuval Ishai. Function secret sharing. In *Advances in Cryptology, EUROCRYPT 2015*, volume 9057 of *Lecture Notes in Computer Science*, pp. 337–367. Springer, 2015. doi: 10.1007/978-3-662-46803-6_12.

Elette Boyle, Niv Gilboa, and Yuval Ishai. Function secret sharing: Improvements and extensions. In *ACM SIGSAC Conference on Computer and Communications Security*, pp. 1292–1303. Association for Computing Machinery, 2016. doi: 10.1145/2976749.2978429. Full version: Cryptology ePrint Archive, Paper 2018/707.

Elette Boyle, Niv Gilboa, and Yuval Ishai. Secure computation with preprocessing via function secret sharing. In *Theory of Cryptography, TCC 2019, Part I*, volume 11891 of *Lecture Notes in Computer Science*, pp. 341–371. Springer, 2019. doi: 10.1007/978-3-030-36030-6_14. Full version: Cryptology ePrint Archive, Paper 2019/1095.

Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. (Leveled) fully homomorphic encryption without bootstrapping. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, pp. 309–325. Association for Computing Machinery, 2012. doi: 10.1145/2090236.2090262. Journal version: ACM Transactions on Computation Theory 6(3), Article 13, 2014.

Rickard Brännvall, Tony Zhang, Henrik Forsgren, Andrei Stoian, Fredrik Sandin, and Marcus Liwicki. Inhibitor transformers and gated RNNs for torus efficient fully homomorphic encryption, 2026. URL https://arxiv.org/abs/2308.05629. arXiv v2, 23 Mar 2026. Consolidated manuscript based on prior workshop contributions (PPAI-24, AAAI-24 student abstract, FHE.org 2024, ICLR 2025 SLLM workshop). Note: v1 of this arXiv ID (Aug 2023) was a different, RNN-only paper titled "ReLU and Addition-based Gated RNN" by Brännvall, Forsgren, Sandin and Liwicki.

Lennart Braun, Daniel Demmler, Thomas Schneider, and Oleksandr Tkachenko. MOTION: A framework for mixed-protocol multi-party computation. *ACM Transactions on Privacy and Security*, 25(2):8:1–8:35, 2022. doi: 10.1145/3490390. URL https://doi.org/10.1145/3490390.

Niklas Büscher, Daniel Demmler, Stefan Katzenbeisser, David Kretzmer, and Thomas Schneider. HyCC: Compilation of hybrid protocols for practical secure computation. In *ACM SIGSAC Conference on Computer and Communications Security*, pp. 847–861, 2018. doi: 10.1145/3243734.3243786. URL https://encrypto.de/papers/BDKKS18.pdf.

Yifei Cai, Zhuoran Li, Yizhou Feng, Qiao Zhang, Hongyi Wu, Danella Zhao, and Chunsheng Xin. SecDTD: Dynamic token drop for secure transformers inference. In *2026 IEEE 11th European Symposium on Security and Privacy (EuroS&P)*, Lisbon, Portugal, 2026. IEEE. URL https://arxiv.org/abs/2603.13670. arXiv:2603.13670; Cryptology ePrint Archive, Paper 2026/731.

Francesco Capano, Jonas Böhler, and Benjamin Weggenmann. SoK: Enhancing cryptographic collaborative learning with differential privacy. In *IEEE Conference on Secure and Trustworthy Machine Learning*. IEEE, 2026a. URL https://arxiv.org/abs/2601.09460. Preprint: arXiv:2601.09460.

Francesco Capano, Jonas Böhler, and Benjamin Weggenmann. SPRINT: Scalable secure and differentially private inference for transformers. *Proceedings on Privacy Enhancing Technologies*, 2026(1):134–153, 2026b. doi: 10.56553/popets-2026-0008. URL https://doi.org/10.56553/popets-2026-0008.

Nishanth Chandran, Divya Gupta, Aseem Rastogi, Rahul Sharma, and Shardul Tripathi. EzPC: Programmable, efficient, and scalable secure two-party computation for machine learning. In *IEEE European Symposium on Security and Privacy*, 2019. URL https://www.microsoft.com/en-us/research/publication/ezpc-programmable-efficient-and-scalable-secure-two-party-computation-for-machine-learning/.

Chenghao Chen, Kailun Qin, Xiaolin Zhang, Chi Zhang, and Dawu Gu. Bifrost: Hybrid TEE-FHE inference for privacy-preserving transformer and LLM serving, 2026a. URL https://arxiv.org/abs/2606.17421.

Guoci Chen, Xiurui Pan, Qiao Li, Bo Mao, Congming Gao, Chengying Huan, Mingzhe Zhang, and Jie Zhang. GPU acceleration of TFHE-based high-precision nonlinear layers for encrypted LLM inference. arXiv preprint arXiv:2604.04783, 2026b. URL https://arxiv.org/abs/2604.04783. v1, 6 Apr 2026; 11 pages, 7 figures.

Jingwei Chen, Linhan Yang, Chen Yang, Shuai Wang, Rui Li, Weijie Miao, Wenyuan Wu, Li Yang, Kang Wu, and Lizhong Dai. Secure transformer-based neural network inference for protein sequence classification. Cryptology ePrint Archive, Paper 2024/1851, 2024. URL https://eprint.iacr.org/2024/1851. Winning solution of the iDASH 2024 Track 1 (homomorphic encryption) competition.

Tianyu Chen, Hangbo Bao, Shaohan Huang, Li Dong, Binxing Jiao, Daxin Jiang, Haoyi Zhou, Jianxin Li, and Furu Wei. THE-X: Privacy-preserving transformer inference with homomorphic encryption. In *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 3510–3520, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-acl.277. URL https://aclanthology.org/2022.findings-acl.277/.

Yuntian Chen, Zhanyong Tang, Tianpei Lu, Bingsheng Zhang, Zhiying Shi, and Zheng Wang. Accelerating private large transformers inference through fine-grained collaborative computation. *IEEE Transactions on Information Forensics and Security*, 20:7482–7497, 2025. doi: 10.1109/TIFS.2025.3584639. arXiv:2412.16537.

Yuntian Chen, Tianpei Lu, Zhanyong Tang, Bingsheng Zhang, Zhiying Shi, Yuxiang Luan, and Zhuzhu Wang. SoK: Private transformer-based model inference. In *35th USENIX Security Symposium*. USENIX Association, 2026c. URL https://www.usenix.org/conference/usenixsecurity26/presentation/chen-yuntian.

Ke Cheng, Yuheng Xia, Anxiao Song, Jiaxuan Fu, Wenjie Qu, Yulong Shen, and Jiaheng Zhang. Mosformer: Maliciously secure three-party inference framework for large transformers. In *Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security (CCS '25)*, New York, NY, USA, 2025. ACM. doi: 10.1145/3719027.3765028. Full version: IACR Cryptology ePrint Archive, Paper 2025/1510, https://eprint.iacr.org/2025/1510.

Jung Hee Cheon, Andrey Kim, Miran Kim, and Yongsoo Song. Homomorphic encryption for arithmetic of approximate numbers. In *Advances in Cryptology, ASIACRYPT 2017*, volume 10624 of *Lecture Notes in Computer Science*, pp. 409–437. Springer, 2017. doi: 10.1007/978-3-319-70694-8_15. Cryptology ePrint Archive, Paper 2016/421.

Marcin Chrapek, Marcin Copik, Etienne Mettaz, and Torsten Hoefler. Confidential LLM inference: Performance and cost across CPU and GPU TEEs. In *IEEE International Symposium on Workload Characterization*, pp. 84–98, Irvine, CA, USA, 2025. IEEE. doi: 10.1109/IISWC66894.2025.00017.

Leo de Castro, Daniel Escudero, Adya Agrawal, Antigoni Polychroniadou, and Manuela Veloso. EncryptedLLM: Privacy-preserving large language model inference via GPU-accelerated fully homomorphic encryption. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 12677–12688. PMLR, 2025. URL https://proceedings.mlr.press/v267/de-castro25a.html.

Daniel Demmler, Thomas Schneider, and Michael Zohner. ABY: A framework for efficient mixed-protocol secure two-party computation. In *Network and Distributed System Security Symposium*, 2015. doi: 10.14722/ndss.2015.23113. URL https://www.ndss-symposium.org/ndss2015/ndss-2015-programme/aby-framework-efficient-mixed-protocol-secure-two-party-computation/.

Naren Dhyani, Jianqiao Cambridge Mo, Patrick Yubeaton, Minsu Cho, Ameya Joshi, Siddharth Garg, Brandon Reagen, and Chinmay Hegde. PriViT: Vision transformers for private inference. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=3CmPvcYJnm. Preprint: arXiv:2310.04604 (Oct. 2023), titled "PriViT: Vision Transformers for Fast Private Inference".

Yuanchao Ding, Hua Guo, Yewei Guan, Weixin Liu, Jiarong Huo, Zhenyu Guan, and Xiyong Zhang. East: Efficient and accurate secure inference framework for transformer. *IEEE Transactions on Services Computing*, 18(4):2038–2046, July 2025. doi: 10.1109/TSC.2025.3577491. Preprint: arXiv:2308.09923 (Aug. 2023), titled "East: Efficient and Accurate Secure Transformer Framework for Inference".

Tian Dong, Yan Meng, Shaofeng Li, Guoxing Chen, Zhen Liu, and Haojin Zhu. Depth gives a false sense of privacy: LLM internal states inversion. In *34th USENIX Security Symposium*, Seattle, WA, 2025a. USENIX Association. URL https://www.usenix.org/conference/usenixsecurity25/presentation/dong-tian. Preprint: arXiv:2507.16372.

Ye Dong, Wen-Jie Lu, Yancheng Zheng, Haoqi Wu, Derun Zhao, Jin Tan, Zhicong Huang, Cheng Hong, Tao Wei, Wen-Guang Chen, and Jianying Zhou. PUMA: Secure inference of LLaMA-7B in five minutes. *Security and Safety*, 4:2025014, 2025b. doi: 10.1051/sands/2025014. URL https://doi.org/10.1051/sands/2025014. Preprint: arXiv:2307.12533.

Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3–4):211–407, 2014. doi: 10.1561/0400000042.

URL https://www.nowpublishers.com/articles/foundations-and-trends-in-theoretical-computer-science/TCS-042.

Junfeng Fan and Frederik Vercauteren. Somewhat practical fully homomorphic encryption. Cryptology ePrint Archive, Paper 2012/144, 2012. URL https://eprint.iacr.org/2012/144.

Mingyuan Fan, Yu Liu, Fuyi Wang, and Cen Chen. What does the server see? understanding privacy leakage from large language models in split inference. In *ACM SIGSAC Conference on Computer and Communications Security*, The Hague, The Netherlands, 2026. Association for Computing Machinery. URL https://arxiv.org/abs/2605.23158. To appear; preprint arXiv:2605.23158.

Jordan Frery, Roman Bredehoft, Jakub Klemsa, Arthur Meyre, and Andrei Stoian. Private LoRA fine-tuning of open-source LLMs with homomorphic encryption, 2025. URL https://arxiv.org/abs/2505.07329. arXiv preprint.

Ran Gilad-Bachrach, Nathan Dowlin, Kim Laine, Kristin Lauter, Michael Naehrig, and John Wernsing. CryptoNets: Applying neural networks to encrypted data with high throughput and accuracy. In *Proceedings of the 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 201–210, 2016. URL https://proceedings.mlr.press/v48/gilad-bachrach16.html.

Zhaoting Gong, Ran Ran, Fan Yao, and Wujie Wen. Scaling long-sequence homomorphic encrypted transformer inference via hybrid parallelism on multi-gpu systems. In *Proceedings of the 40th ACM International Conference on Supercomputing (ICS '26)*, pp. 1206–1219, Belfast, United Kingdom, 2026. ACM. doi: 10.1145/3797905.3800539. arXiv:2604.03425 (titled "AEGIS: ..." on arXiv).

Kanav Gupta, Deepak Kumaraswamy, Nishanth Chandran, and Divya Gupta. LLAMA: A low latency math library for secure inference. *Proceedings on Privacy Enhancing Technologies*, 2022 (4):274–294, 2022. doi: 10.56553/popets-2022-0109. URL https://petsymposium.org/popets/2022/popets-2022-0109.php.

Kanav Gupta, Neha Jawalkar, Ananta Mukherjee, Nishanth Chandran, Divya Gupta, Ashish Panwar, and Rahul Sharma. Sigma: Secure GPT inference with function secret sharing. *Proceedings on Privacy Enhancing Technologies*, 2024(4), 2024. doi: 10.56553/popets-2024-0107. URL https://petsymposium.org/popets/2024/popets-2024-0107.pdf.

Kanav Gupta, Nishanth Chandran, Divya Gupta, Jonathan Katz, and Rahul Sharma. SHARK: Actively secure inference using function secret sharing. In *2025 IEEE Symposium on Security and Privacy (SP)*, pp. 2472–2490. IEEE, 2025. doi: 10.1109/SP61157.2025.00175. Full version: IACR Cryptology ePrint Archive, Report 2025/716.

Meng Hao, Hongwei Li, Hanxiao Chen, Pengzhi Xing, Guowen Xu, and Tianwei Zhang. Iron: Private inference on transformers. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/64e2449d74f84e5b1a5c96ba7b3d308e-Abstract-Conference.html.

Hanieh Hashemi, Yongqin Wang, and Murali Annavaram. DarKnight: An accelerated framework for privacy and integrity preserving deep learning using trusted hardware. In *MICRO-54: 54th Annual IEEE/ACM International Symposium on Microarchitecture*, pp. 212–224. Association for Computing Machinery, 2021. doi: 10.1145/3466752.3480112.

Jiaxing He, Kang Yang, Guofeng Tang, Zhangjie Huang, Li Lin, Changzheng Wei, Ying Yan, and Wei Wang. Rhombus: Fast homomorphic matrix-vector multiplication for secure two-party inference. In *Proceedings of the 2024 ACM SIGSAC Conference on Computer and Communications Security (CCS '24)*, pp. 2490–2504, Salt Lake City, UT, USA, 2024. ACM. doi: 10.1145/3658644.3690281. Full version: IACR Cryptology ePrint Archive, Paper 2024/1611, https://eprint.iacr.org/2024/1611.

Zhiyu He, Maojiang Wang, Xinwen Gao, Yuchuan Luo, Lin Liu, and Shaojing Fu. ENSI: Efficient non-interactive secure inference for large language models. In *2025 44th International Symposium on Reliable Distributed Systems (SRDS)*, pp. 83–94. IEEE, 2025. doi: 10.1109/SRDS69199.2025.00021. arXiv:2509.09424.

Xiaoyang Hou, Jian Liu, Jingyu Li, Yuhan Li, Jiawen Zhang, Wen-jie Lu, Cheng Hong, and Kui Ren. CipherGPT: Secure two-party GPT inference. *IEEE Transactions on Dependable and Secure Computing*, 23(3):6748–6762, May 2026. doi: 10.1109/TDSC.2026.3667722. Preprint: Cryptology ePrint Archive, Paper 2023/1147, https://eprint.iacr.org/2023/1147.

Zhicong Huang, Wen-jie Lu, Cheng Hong, and Jiansheng Ding. Cheetah: Lean and fast secure two-party deep neural network inference. In *31st USENIX Security Symposium (USENIX Security 22)*, pp. 809–826, Boston, MA, 2022. USENIX Association. ISBN 978-1-939133-31-1. URL https://www.usenix.org/conference/usenixsecurity22/presentation/huang-zhicong.

Neha Jawalkar, Kanav Gupta, Arkaprava Basu, Nishanth Chandran, Divya Gupta, and Rahul Sharma. Orca: FSS-based secure training and inference with GPUs. In *2024 IEEE Symposium on Security and Privacy (SP)*, pp. 597–616, San Francisco, CA, USA, 2024. IEEE. doi: 10.1109/SP54263.2024.00063. Preprint: Cryptology ePrint Archive, Paper 2023/206.

Wen jie Lu, Zhicong Huang, Zhen Gu, Jingyu Li, Jian Liu, Cheng Hong, Kui Ren, Tao Wei, and Wenguang Chen. BumbleBee: Secure two-party inference framework for large transformers. In *Network and Distributed System Security Symposium*, 2025. URL https://www.ndss-symposium.org/ndss-paper/bumblebee-secure-two-party-inference-framework-for-large-transformers/.

Chiraag Juvekar, Vinod Vaikuntanathan, and Anantha Chandrakasan. GAZELLE: A low latency framework for secure neural network inference. In *27th USENIX Security Symposium*, pp. 1651–1669. USENIX Association, 2018. URL https://www.usenix.org/conference/usenixsecurity18/presentation/juvekar.

Andes Y. L. Kei and Sherman S. M. Chow. SHAFT: Secure, handy, accurate, and fast transformer inference. In *Network and Distributed System Security (NDSS) Symposium 2025*, San Diego, CA, USA, 2025. Internet Society. doi: 10.14722/ndss.2025.242287. URL https://www.ndss-symposium.org/ndss-paper/shaft-secure-handy-accurate-and-fast-transformer-inference/. NDSS 2025 Artifact Evaluation Award (Distinguished Artifact Award).

Marcel Keller. MP-SPDZ: A versatile framework for multi-party computation. In *ACM SIGSAC Conference on Computer and Communications Security*, pp. 1575–1590. Association for Computing Machinery, 2020. doi: 10.1145/3372297.3417872. URL https://doi.org/10.1145/3372297.3417872. Cryptology ePrint Archive, Paper 2020/521.

Brian Knott, Shobha Venkataraman, Awni Hannun, Shubho Sengupta, Mark Ibrahim, and Laurens van der Maaten. CrypTen: Secure multi-party computation meets machine learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 4961–4973. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/hash/2754518221cfbc8d25c13a06a4cb8421-Abstract.html.

Nishant Kumar, Mayank Rathee, Nishanth Chandran, Divya Gupta, Aseem Rastogi, and Rahul Sharma. CrypTFlow: Secure TensorFlow inference. In *IEEE Symposium on Security and Privacy*, pp. 336–353. IEEE, 2020. doi: 10.1109/SP40000.2020.00092. Preprint: arXiv:1909.07814.

Caihua Li, In Gim, and Lin Zhong. Confidential prompting: Privacy-preserving LLM inference on cloud, 2024a. URL https://arxiv.org/abs/2409.19134. arXiv preprint; latest version v5, 19 Nov 2025.

Dacheng Li, Hongyi Wang, Rulin Shao, Han Guo, Eric Xing, and Hao Zhang. MPCFormer: Fast, performant and private transformer inference with MPC. In *International Conference on Learning Representations*, 2023. URL https://iclr.cc/virtual/2023/poster/12110.

Ding Li, Ziqi Zhang, Mengyu Yao, Yifeng Cai, Yao Guo, and Xiangqun Chen. TEESlice: Protecting sensitive neural network models in trusted execution environments when attackers have pre-trained models. *ACM Transactions on Software Engineering and Methodology*, 34(6):1–49, 2025a. doi: 10.1145/3707453. URL https://doi.org/10.1145/3707453.

Xuechen Li, Florian Tramèr, Percy Liang, and Tatsunori Hashimoto. Large language models can be strong differentially private learners. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2110.05679.

Zhengyi Li, Kang Yang, Jin Tan, Wen-jie Lu, Haoqi Wu, Xiao Wang, Yu Yu, Derun Zhao, Yancheng Zheng, Minyi Guo, and Jingwen Leng. Nimbus: Secure and efficient two-party inference for transformers. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*, 2024b. doi: 10.52202/079017-0680. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/264a9b3ce46abdf572dcfe0401141989-Abstract-Conference.html. arXiv:2411.15707.

Zhengyi Li, Yue Guan, Kang Yang, Yu Feng, Ning Liu, Yu Yu, Jingwen Leng, and Minyi Guo. An efficient private GPT never autoregressively decodes. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 34410–34428. PMLR, 2025b. URL https://proceedings.mlr.press/v267/li25q.html.

Zhengyi Li, Yakai Wang, Jingwen Leng, Kang Yang, Yu Yu, Jiaping Gui, Yu Feng, Ning Liu, and Minyi Guo. On the (in-)security of the shuffling defense in the transformer secure inference. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 29084–29098. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.1341. URL https://aclanthology.org/2026.acl-long.1341/.

Zi Liang, Pinghui Wang, Ruofei Zhang, Nuo Xu, Shuo Zhang, Lifeng Xing, Haitao Bai, and Ziyang Zhou. MERGE: Fast private text generation. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI-24)*, volume 38, pp. 19884–19892. AAAI Press, 2024. doi: 10.1609/aaai.v38i18.29964. URL https://ojs.aaai.org/index.php/AAAI/article/view/29964. Issue 18.

Jinglong Luo, Yehong Zhang, Zhuo Zhang, Jiaqi Zhang, Xin Mu, Hui Wang, Yue Yu, and Zenglin Xu. SecFormer: Fast and accurate privacy-preserving inference for transformer models via SMPC. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 13333–13348, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.790. URL https://aclanthology.org/2024.findings-acl.790/.

Jinglong Luo, Guanzhong Chen, Yehong Zhang, Shiyu Liu, Hui Wang, Yue Yu, Xun Zhou, Yuan Qi, and Zenglin Xu. CENTAUR: Bridging the impossible trinity of privacy, efficiency, and performance in privacy-preserving transformer inference. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 22751–22770. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.1109. URL https://aclanthology.org/2025.acl-long.1109/.

Junming Ma, Yancheng Zheng, Jun Feng, Derun Zhao, Haoqi Wu, Wenjing Fang, Jin Tan, Chaofan Yu, Benyu Zhang, and Lei Wang. SecretFlow-SPU: A performant and User-Friendly framework for Privacy-Preserving machine learning. In *2023 USENIX Annual Technical Conference (USENIX ATC 23)*, pp. 17–33, Boston, MA, July 2023. USENIX Association. ISBN 978-1-939133-35-9. URL https://www.usenix.org/conference/atc23/presentation/ma.

Yuhan Ma, Yong Li, and Stefan Schmid. FuseFSS: Efficient secure LLM inference with function secret sharing. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*, volume 306 of *Proceedings of Machine Learning Research*, Seoul, South Korea, 2026. PMLR. URL https://openreview.net/forum?id=WpUpj8DrVB. arXiv:2606.09551.

Pratyush Mishra, Ryan Lehmkuhl, Akshayaram Srinivasan, Wenting Zheng, and Raluca Ada Popa. Delphi: A cryptographic inference service for neural networks. In *29th USENIX Security Symposium (USENIX Security 20)*, pp. 2505–2522. USENIX Association, August 2020. ISBN 978-1-939133-17-5. URL https://www.usenix.org/conference/usenixsecurity20/presentation/mishra.

Payman Mohassel and Peter Rindal. ABY3: A mixed protocol framework for machine learning. In *ACM SIGSAC Conference on Computer and Communications Security*, 2018. URL https://eprint.iacr.org/2018/403.

Payman Mohassel and Yupeng Zhang. SecureML: A system for scalable privacy-preserving machine learning. In *IEEE Symposium on Security and Privacy*, pp. 19–38. IEEE, 2017. doi: 10.1109/SP.2017.12. Cryptology ePrint Archive, Paper 2017/396.

Jungho Moon, Dongwoo Yoo, Xiaoqian Jiang, and Miran Kim. THOR: Secure transformer inference with homomorphic encryption. In *ACM SIGSAC Conference on Computer and Communications Security*, pp. 3765–3779, 2025. doi: 10.1145/3719027.3765150. URL https://doi.org/10.1145/3719027.3765150.

Muqsit Nawaz, Aditya Gulati, Kunlong Liu, Vishwajeet Agrawal, Prabhanjan Ananth, and Trinabh Gupta. Accelerating 2PC-based ML with limited trusted hardware. arXiv preprint arXiv:2009.05566, 2020. URL https://arxiv.org/abs/2009.05566.

Tushar Nayan, Ziqi Zhang, and Ruimin Sun. SecureInfer: Heterogeneous TEE-GPU architecture for privacy-critical tensors for large language model deployment. In *2025 First International Conference on Intelligent Computing and Systems at the Edge (ICEdge)*, pp. 1–7, Bangalore, India, 2025. IEEE. doi: 10.1109/ICEdge67252.2025.11412803. URL https://doi.org/10.1109/ICEdge67252.2025.11412803. Preprint available as arXiv:2510.19979.

Lucien K. L. Ng and Sherman S. M. Chow. SoK: Cryptographic neural-network computation. In *IEEE Symposium on Security and Privacy*, pp. 497–514. IEEE, 2023. doi: 10.1109/SP46215.2023.10179483. URL https://doi.org/10.1109/SP46215.2023.10179483.

Nges Brian Njungle, Eric Jahns, Luigi Mastromauro, Edwin P. Kayang, Milan Stojkov, and Michel A. Kinsy. Prismo: A decision support system for privacy-preserving ML framework selection, 2025. URL https://arxiv.org/abs/2510.09985.

Qi Pang, Jinhao Zhu, Helen Möllering, Wenting Zheng, and Thomas Schneider. BOLT: Privacy-preserving, accurate and efficient inference for transformers. In *IEEE Symposium on Security and Privacy*, pp. 4753–4771, 2024. doi: 10.1109/SP54263.2024.00130. URL https://doi.org/10.1109/SP54263.2024.00130.

Dongjin Park, Eunsang Lee, and Joon-Woo Lee. Powerformer: Efficient and high-accuracy privacy-preserving language model with homomorphic encryption. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11090–11111, Vienna, Austria, July 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.543. URL https://aclanthology.org/2025.acl-long.543/.

Hanjun Park, Byeongseo Min, Jiheon Woo, Min-Wook Jeong, Jongho Shin, Yongwoo Lee, Young-Sik Kim, and Yongjune Kim. CGF-Softmax: A cumulant-based softmax reformulation for efficient inference under homomorphic encryption. arXiv preprint arXiv:2602.01621, 2026a. URL https://arxiv.org/abs/2602.01621. v3, 8 May 2026. v1 (2 Feb 2026) was titled "Efficient Softmax Reformulation for Homomorphic Encryption via Moment Generating Function" (MGF-softmax).

Jaiyoung Park, Sejin Park, Jai Hyun Park, Jung Ho Ahn, Jung Hee Cheon, Guillaume Hanrot, Jung Woo Kim, Minje Park, and Damien Stehlé. Scaling up FHE-based privacy-preserving ML: Higher throughput, longer inputs for LLama-3-8B, 2026b. URL https://arxiv.org/abs/2601.18511. arXiv preprint, v2 (17 Jun 2026).

Arpita Patra, Thomas Schneider, Ajith Suresh, and Hossein Yalame. ABY2.0: Improved mixed-protocol secure two-party computation. In *30th USENIX Security Symposium*, pp. 2165–2182.

USENIX Association, 2021. ISBN 978-1-939133-24-3. URL https://www.usenix.org/conference/usenixsecurity21/presentation/patra. Cryptology ePrint Archive, Paper 2020/1225.

Robert Podschwadt, Daniel Takabi, Peizhao Hu, Mohammad H. Rafiei, and Zhipeng Cai. A survey of deep learning architectures for privacy-preserving machine learning with fully homomorphic encryption. *IEEE Access*, 10:117477–117500, 2022. doi: 10.1109/ACCESS.2022.3219049. URL https://doi.org/10.1109/ACCESS.2022.3219049. Peer-reviewed and expanded version of the preprint arXiv:2112.12855 (2021).

Deevashwer Rathee, Mayank Rathee, Nishant Kumar, Nishanth Chandran, Divya Gupta, Aseem Rastogi, and Rahul Sharma. CrypTFlow2: Practical 2-party secure inference. In *ACM SIGSAC Conference on Computer and Communications Security*. Association for Computing Machinery, 2020. doi: 10.1145/3372297.3417274. Cryptology ePrint Archive, Paper 2020/1002; arXiv:2010.06457.

Deevashwer Rathee, Mayank Rathee, Rahul Kranti Kiran Goli, Divya Gupta, Rahul Sharma, Nishanth Chandran, and Aseem Rastogi. SiRnn: A math library for secure RNN inference. In *2021 IEEE Symposium on Security and Privacy (SP)*, pp. 1003–1020, San Francisco, CA, USA, 2021. IEEE. doi: 10.1109/SP40001.2021.00086. arXiv:2105.04236.

Deevashwer Rathee, Dacheng Li, Ion Stoica, Hao Zhang, and Raluca Ada Popa. MPC-minimized secure LLM inference. arXiv preprint arXiv:2408.03561, 2024. URL https://arxiv.org/abs/2408.03561.

Donghwan Rho, Taeseong Kim, Minje Park, Jung Woo Kim, Hyunsik Chae, Ernest K. Ryu, and Jung Hee Cheon. Encryption-friendly LLM architecture. In *International Conference on Learning Representations*, 2025a. URL https://arxiv.org/abs/2410.02486.

Donghwan Rho, Sieun Seo, Hyewon Sung, Chohong Min, and Ernest K. Ryu. Traveling salesman-based token ordering improves stability in homomorphically encrypted language models. arXiv preprint arXiv:2510.12343, 2025b. URL https://arxiv.org/abs/2510.12343.

Abhishek Saini, Haolin Jiang, and Hang Liu. Vulnerabilities in partial TEE-shielded LLM inference with precomputed noise, 2026. URL https://arxiv.org/abs/2602.11088.

Manuel B. Santos, Dimitris Mouris, Mehmet Ugurbil, Stanislaw Jarecki, José Reis, Shubho Sengupta, and Miguel de Vega. Curl: Private LLMs through wavelet-encoded look-up tables. In Rachel Allen, Sagar Samtani, Edward Raff, and Ethan Rudd (eds.), *Proceedings of the Conference on Applied Machine Learning in Information Security (CAMLIS 2024)*, volume 3920 of *CEUR Workshop Proceedings*, pp. 16–47, Arlington, Virginia, USA, October 2024. CEUR-WS.org. URL https://ceur-ws.org/Vol-3920/paper02.pdf. Preprint: Cryptology ePrint Archive, Paper 2024/1127, https://eprint.iacr.org/2024/1127.

Robert Schambach, Quoc Do Le, Sergei Arnautov, and Christof Fetzer. EnclaveX: End-to-end confidential AI with CPU/GPU TEEs, 2026. URL https://arxiv.org/abs/2606.31408.

Kyle Storrier, Adithya Vadapalli, Allan Lyons, and Ryan Henry. Grotto: Screaming fast $(2+1)$-PC for $\mathbb{Z}_{2^n}$ via $(2,2)$-DPFs. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security (CCS '23)*, pp. 2143–2157, Copenhagen, Denmark, 2023. ACM. doi: 10.1145/3576915.3623147. Full version: IACR Cryptology ePrint Archive, Report 2023/108, https://eprint.iacr.org/2023/108.

Yifan Tan, Cheng Tan, Zeyu Mi, and Haibo Chen. PipeLLM: Fast and confidential large language model services with speculative pipelined encryption. In *Proceedings of the 30th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 1*, pp. 843–857. Association for Computing Machinery, 2025. doi: 10.1145/3669940.3707224. Preprint: arXiv:2411.03357.

Marcus Taubert, Adam Skuta, and Thomas Lorünser. A pragmatic comparison of cryptographic computation technologies for machine learning. In *Proceedings of the 12th International Conference on Information Systems Security and Privacy, Volume 2*, pp. 422–430, Marbella, Spain,

2026. SciTePress. ISBN 978-989-758-800-6. doi: 10.5220/0014567700004061. Preprint: arXiv:2605.04858.

Rahul Krishna Thomas, Louai Zahran, Erica Choi, Akilesh Potti, Micah Goldblum, and Arka Pal. Hidden no more: Attacking and defending private third-party LLM inference. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 59434–59469, 2025. URL https://proceedings.mlr.press/v267/thomas25b.html.

Florian Tramèr and Dan Boneh. Slalom: Fast, verifiable and private execution of neural networks in trusted hardware. In *International Conference on Learning Representations*, 2019. URL https://www.floriantramer.com/docs/papers/iclr19slalom.pdf.

Mehmet Ugurbil, Dimitris Mouris, Manuel B. Santos, José Cabrero-Holgueras, Miguel de Vega, and Shubho Sengupta. Fission: Distributed privacy-preserving large language model inference. In *2025 IEEE Conference on Communications and Network Security (CNS)*, pp. 1–9, Avignon, France, 2025. IEEE. doi: 10.1109/CNS66487.2025.11194960. Preprint: Cryptology ePrint Archive, Paper 2025/653, https://eprint.iacr.org/2025/653.

Sameer Wagh. Pika: Secure computation using function secret sharing over rings. *Proceedings on Privacy Enhancing Technologies*, 2022(4):351–377, 2022. doi: 10.56553/popets-2022-0113. URL https://petsymposium.org/popets/2022/popets-2022-0113.php.

Wei Wang, Abdul Hyee Waqas, and Burns Smith. Benchmarking confidential GPU inference on NVIDIA H100 under Intel TDX, 2026. URL https://arxiv.org/abs/2607.19353.

Yongqin Wang, G. Edward Suh, Wenjie Xiong, Benjamin Lefaudeux, Brian Knott, Murali Annavaram, and Hsien-Hsin S. Lee. Characterization of MPC-based private inference for transformer-based models. In *2022 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*, pp. 187–197. IEEE, 2022. doi: 10.1109/ISPASS55109.2022.00025. URL https://ieeexplore.ieee.org/document/9804616/.

Jean-Luc Watson, Sameer Wagh, and Raluca Ada Popa. Piranha: A GPU platform for secure computation. In *31st USENIX Security Symposium (USENIX Security 22)*, pp. 827–844, Boston, MA, August 2022. USENIX Association. ISBN 978-1-939133-31-1. URL https://www.usenix.org/conference/usenixsecurity22/presentation/watson.

Haoqi Wu, Wenjing Fang, Yancheng Zheng, Junming Ma, Jin Tan, and Lei Wang. Ditto: Quantization-aware secure inference of transformers upon MPC. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 53346–53365, 2024. URL https://proceedings.mlr.press/v235/wu24d.html.

Pan Xiao, Rending Ouyang, Heng Zhang, Jiawen Zhang, and Jian Liu. Bootstrapping is all you need: Secure transformer inference via improved CKKS functional bootstrapping. Cryptology ePrint Archive, Paper 2026/1255, 2026. URL https://eprint.iacr.org/2026/1255. Preprint.

Jianhang Xie, Sicheng Tan, Vishnu Naresh Boddeti, and Zhichao Lu. ATLAS: Automated approximation of transformers for efficient homomorphic inference in one hour. arXiv preprint arXiv:2607.23478, 2026. URL https://arxiv.org/abs/2607.23478. Also IACR Cryptology ePrint Archive, Report 2026/1817. Code: https://github.com/jianhayes/ATLAS.

Tianshi Xu, Wen jie Lu, Jiangrui Yu, Yi Chen, Chenqi Lin, Runsheng Wang, and Meng Li. Breaking the layer barrier: Remodeling private transformer inference with hybrid CKKS and MPC. In *34th USENIX Security Symposium*, pp. 2653–2672. USENIX Association, 2025. URL https://www.usenix.org/conference/usenixsecurity25/presentation/xu-tianshi.

Xiangrui Xu, Qiao Zhang, Rui Ning, Chunsheng Xin, and Hongyi Wu. Comet: A communication-efficient and performant approximation for private transformer inference. arXiv preprint arXiv:2405.17485, 2024. URL https://arxiv.org/abs/2405.17485. v2, 7 September 2024.

Jiaqi Xue, Xin Xin, Wei Zhang, Mengxin Zheng, Qianqian Song, Minxuan Zhou, Yushun Dong, Dongjie Wang, Xun Chen, Jiafeng Xie, Liqiang Wang, David Mohaisen, Hongyi Wu, and Qian Lou. SoK: Can fully homomorphic encryption support general AI computation? a functional and cost analysis. *Proceedings on Privacy Enhancing Technologies*, 2026(2):680–696, 2026. doi: 10.56553/popets-2026-0066. URL https://petsymposium.org/popets/2026/popets-2026-0066.php. arXiv:2504.11604.

Timofey Yaluhin. SoK: Confidential transformer inference and retrieval-augmented generation. Cryptology ePrint Archive, Paper 2026/1544, 2026. URL https://eprint.iacr.org/2026/1544.

Guang Yan, Yuhui Zhang, Zimu Guo, Lutan Zhao, Xiaojun Chen, Chen Wang, Wenhao Wang, Dan Meng, and Rui Hou. Comet: Accelerating private inference for large language model by predicting activation sparsity. In *2025 IEEE Symposium on Security and Privacy (SP)*, pp. 2827–2845, San Francisco, CA, USA, 2025. IEEE. doi: 10.1109/SP61157.2025.00182. URL https://doi.org/10.1109/SP61157.2025.00182. arXiv:2505.07239.

Huan Yang, Deyu Zhang, Yudong Zhao, Yuanchun Li, and Yunxin Liu. A first look at efficient and secure on-device LLM inference against KV leakage. In *Proceedings of the 19th Workshop on Mobility in the Evolving Internet Architecture*, Washington, DC, USA, 2024. Association for Computing Machinery. doi: 10.1145/3691555.3696827. Preprint: arXiv:2409.04040.

Linhan Yang, Jingwei Chen, Wangchen Dai, Shuai Wang, Wenyuan Wu, and Yong Feng. ARION: Attention-optimized transformer inference on encrypted data. Cryptology ePrint Archive, Paper 2025/2271, 2025. URL https://eprint.iacr.org/2025/2271. Preprint, dated 18 December 2025.

Wenshao Yang, Zhenhua Liu, and Dongdong Yao. General techniques for reducing key-switching overhead in privacy-preserving two-party transformer inference, 2026. URL https://arxiv.org/abs/2606.25349.

Ye Yu, Yifan Zhou, Yi Chen, Pedro Soto, Wenjie Xiong, and Meng Li. Cachemir: Fully homomorphic encrypted inference of generative large language model with KV cache. arXiv preprint arXiv:2602.11470, 2026. URL https://arxiv.org/abs/2602.11470. Under review.

Mu Yuan, Lan Zhang, Yihang Cheng, Miao-Hui Song, Guoliang Xing, and Xiang-Yang Li. STIP: Three-party privacy-preserving and lossless inference for large transformers in production. In *Network and Distributed System Security Symposium*, 2026. doi: 10.14722/ndss.2026.230035. URL https://www.ndss-symposium.org/ndss-paper/stip-three-party-privacy-preserving-and-lossless-inference-for-large-transformers-in-production/.

Patrick Yubeaton, Jianqiao Cambridge Mo, Karthik Garimella, Nandan Kumar Jha, Brandon Reagen, Chinmay Hegde, and Siddharth Garg. Truncformer: Private LLM inference using only truncations, 2024. URL https://arxiv.org/abs/2412.01042. arXiv preprint arXiv:2412.01042.

Wenxuan Zeng, Meng Li, Wenjie Xiong, Tong Tong, Wen-jie Lu, Jin Tan, Runsheng Wang, and Ru Huang. MPCViT: Searching for accurate and efficient MPC-friendly vision transformer with heterogeneous attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5052–5063, 2023. arXiv:2211.13955.

Wenxuan Zeng, Ye Dong, Jinjin Zhou, Jin Tan, Lei Wang, Tao Wei, Runsheng Wang, and Meng Li. MPCache: MPC-friendly KV cache eviction for efficient private LLM inference. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025a. URL https://neurips.cc/virtual/2025/poster/116363. arXiv:2501.06807.

Wenxuan Zeng, Tianshi Xu, Yi Chen, Yifan Zhou, Mingzhe Zhang, Jin Tan, Cheng Hong, and Meng Li. Towards efficient privacy-preserving machine learning: A systematic review from protocol, model, and system perspectives, 2025b. URL https://arxiv.org/abs/2507.14519. Also Cryptology ePrint Archive, Paper 2025/1365.

Ziqian Zeng, Jianwei Wang, Junyao Yang, Zhengdong Lu, Haoran Li, Huiping Zhuang, and Cen Chen. PrivacyRestore: Privacy-preserving inference in large language models via privacy removal and restoration. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10821–10855, Vienna, Austria, 2025c. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.532. URL https://aclanthology.org/2025.acl-long.532/.

Hedong Zhang, Neusha Javidnia, Shweta Pardeshi, Qian Lou, and Farinaz Koushanfar. CryptoGen: Secure transformer generation with encrypted KV-cache reuse, 2026a. URL https://arxiv.org/abs/2602.08798.

Jiawen Zhang, Xinpeng Yang, Lipeng He, Kejia Chen, Wen jie Lu, Yinghao Wang, Xiaoyang Hou, Jian Liu, Kui Ren, and Xiaohu Yang. Secure transformer inference made non-interactive. In *Network and Distributed System Security Symposium*, 2025a. URL https://www.ndss-symposium.org/ndss-paper/secure-transformer-inference-made-non-interactive/.

Linru Zhang, Xiangning Wang, Jun Jie Sim, Zhicong Huang, Jiahao Zhong, Huaxiong Wang, Pu Duan, and Kwok Yan Lam. MOAI: Module-optimizing architecture for non-interactive secure transformer inference. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026b. URL https://openreview.net/forum?id=qJn4HtTzhH. Poster. Preprint: Cryptology ePrint Archive, Paper 2025/991.

Xiaojin Zhang, Yahao Pang, Yan Kang, Wei Chen, Lixin Fan, Hai Jin, and Qiang Yang. No free lunch theorem for privacy-preserving LLM inference. *Artificial Intelligence*, 341:104293, 2025b. doi: 10.1016/j.artint.2025.104293. URL https://doi.org/10.1016/j.artint.2025.104293.

Yancheng Zhang, Jiaqi Xue, Mengxin Zheng, Mimi Xie, Mingzhe Zhang, Lei Jiang, and Qian Lou. CipherPrune: Efficient and scalable private transformer inference. In *International Conference on Learning Representations*, 2025c. URL https://arxiv.org/abs/2502.16782.

Yuke Zhang, Dake Chen, Souvik Kundu, Chenghao Li, and Peter A. Beerel. SAL-ViT: Towards latency efficient private inference on ViT using selective attention search with a learnable softmax approximation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5116–5125. IEEE, October 2023. doi: 10.1109/ICCV51070.2023.00472.

Fei Zheng, Chaochao Chen, Zhongxuan Han, and Xiaolin Zheng. PermLLM: Private inference of large language models within 3 seconds under WAN. arXiv preprint arXiv:2405.18744v1, 2024. URL https://arxiv.org/abs/2405.18744v1.

Mengxin Zheng, Qian Lou, and Lei Jiang. Primer: Fast private transformer inference on encrypted data. In *2023 60th ACM/IEEE Design Automation Conference (DAC)*, pp. 1–6, San Francisco, CA, USA, July 2023. IEEE. doi: 10.1109/DAC56929.2023.10247719. arXiv:2303.13679.

Haoling Zhou, Shixuan Zhao, Chao Wang, and Zhiqiang Lin. OpenPCC: Open and confidential LLM serving on commodity TEEs, 2026. URL https://arxiv.org/abs/2606.11145.

Yifan Zhou, Tianshi Xu, Jue Hong, Ye Wu, and Meng Li. CryptoMoE: Privacy-preserving and scalable mixture of experts inference via balanced expert routing. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://neurips.cc/virtual/2025/poster/119592. Preprint: arXiv:2511.01197.

Jianwei Zhu, Hang Yin, Peng Deng, Aline Almeida, and Shunfan Zhou. Confidential computing on NVIDIA Hopper GPUs: A performance benchmark study, 2024. URL https://arxiv.org/abs/2409.03992.

Yufan Zhu, Chao Jin, Khin Mi Mi Aung, and Xiaokui Xiao. EncFormer: Secure and efficient transformer inference over encrypted data, 2026. URL https://arxiv.org/abs/2604.09975. Manuscript submitted to IEEE TDSC.

Itamar Zimerman, Moran Baruch, Nir Drucker, Gilad Ezov, Omri Soceanu, and Lior Wolf. Converting transformers to polynomial form for secure inference over homomorphic encryption. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 62803–62814. PMLR, 2024. URL https://proceedings.mlr.press/v235/zimerman24a.html.

Itamar Zimerman, Allon Adir, Ehud Aharoni, Matan Avitan, Moran Baruch, Nir Drucker, Jenny Lerner, Ramy Masalha, Reut Moshe, and Omri Soceanu. PowerSoftmax: Towards secure LLM inference over encrypted data. In Emtiyaz Khan, Yingzhen Li, Arno Solin, and Aaditya Ramdas (eds.), *Proceedings of The 29th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 300 of *Proceedings of Machine Learning Research*, pp. 5203–5211. PMLR, May 2026. URL https://proceedings.mlr.press/v300/zimerman26a.html. arXiv:2410.09457.

Alessandro Zirilli, Davide Marincione, Evgenios M. Kornaropoulos, Giuseppe Ateniese, and Emanuele Rodolà. HEAT: Faster fully homomorphic inference via approximations-weights co-adaptation. arXiv preprint arXiv:2609.01730; also IACR Cryptology ePrint Archive, Report 2026/1862, 2026. URL https://arxiv.org/abs/2609.01730. 4 pages. Preprint.

A Review protocol and corpus

The search combines five facets, not one query. No single vocabulary covers these literatures, so the search combines five facets pairwise. The facets are privacy mechanism (MPC, secret sharing, FSS, homomorphic encryption, trusted execution, differential privacy); workload (Transformer, KV cache, decoding); task framing (private, secure or oblivious inference); component (GELU, softmax, truncation); and evidence type (SoK, benchmark, attack). It sweeps security, cryptography, machine-learning and systems venues, the Cryptology ePrint Archive and arXiv, cut off at 11 September 2026.

Inclusion requires a visited record and a line of descent. A paper is included when it proposes, analyses, benchmarks or supplies an essential primitive for inference under a privacy mechanism; is retrievable at a URL actually opened; and evaluates a Transformer workload, underpins one, or bears on its privacy contract. Secure training and private fine-tuning are excluded except as three contrast anchors, as are primitives with no descent into secure inference. Search sweeps return 169 records and the previous corpus 28, for 197 candidates. Deduplication removes 39 exact duplicate records and then collapses the remaining title variants of the same work, leaving 130 unique titles; chaining adds 3; the 133 titles verified in full text produce 134 records. Dropping three version duplicates and merging four pairs leaves 127 verified papers plus one carried over unverified: the 128 rows of the released corpus file (Appendix E), 85 of them Transformer inference.

The corpus is a purposive sample, not a census. Foundational frameworks control novelty statements, the MPC-based, HE-based and hybrid systems establish workload relevance, the DP and TEE tracks separate security semantics. Note that the corpus is not exhaustive and supports no performance meta-analysis.

Table 4: The reviewed corpus, grouped by family. Primitive gives the leading clause of the primitive description recorded for each entry, truncated for width. Net: which network setting the record reports. Off: whether it reports offline cost separately. Adaptivity: how its execution plan is chosen. LLM: whether it evaluates a Transformer language model. A dash marks a property that does not apply or that the source does not state. Party counts, threat models and nonlinear methods are recorded per record in the released corpus file (Appendix E), which is authoritative for every coding here.

Record	Year	Primitive	Net	Off	Adaptivity	LLM
MPC-based secure inference (23 records)						
MPCFormer (Li et al., 2023)	2023	2PC additive secret sharing…	LAN	no	co-design	yes

Table 4 continued from the previous page.

Record	Year	Primitive	Net	Off	Adaptivity	LLM
MPCViT (Zeng et al., 2023)	2023	2PC additive secret sharing…	WAN	no	co-design	–
Privformer (Akimoto et al., 2023)	2023	3PC 2-out-of-3 replicated secret…	LAN	–	co-design	yes
PUMA (Dong et al., 2025b)	2023	3PC 2-out-of-3 replicated secret…	LAN	no	fixed	yes
SAL-ViT (Zhang et al., 2023)	2023	2PC arithmetic secret sharing…	none	no	co-design	yes
Curl (Santos et al., 2024)	2024	2PC additive SS over $\mathbb{Z}_{2^{64}}$ + binary…	LAN	no	manual	yes
Ditto (Wu et al., 2024)	2024	3PC 2-out-of-3 replicated secret…	LAN+WAN	no	typed	yes
MARILL (Rathee et al., 2024)	2024	Arithmetic secret sharing…	LAN	no	co-design	yes
MERGE (Liang et al., 2024)	2024	2PC additive secret sharing…	LAN	no	co-design	yes
Orca (Jawalkar et al., 2024)	2024	2PC FSS…	LAN+WAN	yes	fixed	–
PriViT (Dhyani et al., 2024)	2024	No new cryptographic protocol	LAN	no	co-design	–
SecFormer (Luo et al., 2024)	2024	2PC additive (arithmetic) secret…	LAN	no	co-design	yes
SIGMA (Gupta et al., 2024)	2024	2PC additive secret sharing + FSS…	LAN	yes	fixed	yes
TruncFormer (Yubeaton et al., 2024)	2024	2PC additive secret sharing…	none	yes	compiler	yes
Comet (predictive) (Yan et al., 2025)	2025	2PC additive secret sharing…	LAN+WAN	no	input	yes
CryptoMoE (Zhou et al., 2025)	2025	Two-party secret sharing…	LAN+WAN	no	co-design	yes
Mosformer (Cheng et al., 2025)	2025	3PC replicated secret sharing…	LAN+WAN	yes	fixed	yes
MPCache (Zeng et al., 2025a)	2025	3PC 2-out-of-3 replicated secret…	LAN	no	input	yes
SHAFT (Kei & Chow, 2025)	2025	2-out-of-2 arithmetic secret…	LAN+WAN	no	fixed	yes
SHARK (Gupta et al., 2025)	2025	2PC authenticated additive secret…	LAN+WAN	yes	fixed	yes
FuseFSS (Ma et al., 2026)	2026	2PC masked/additive secret sharing…	LAN+WAN	yes	fixed	yes
SecDTD (Cai et al., 2026)	2026	Secure median selection…	none	–	input	yes
SPRINT (Capano et al., 2026b)	2026	2-out-of-2 additive and binary…	LAN+WAN	no	co-design	yes
HE-based secure inference (24 records)						
CryptoNets (Gilad-Bachrach et al., 2016)	2016	Leveled RLWE/NTRU-style…	none	–	co-design	–
THE-X (Chen et al., 2022)	2022	CKKS (Microsoft SEAL 3.6)…	none	–	co-design	yes
Poly-Transformers (Zimerman et al., 2024)	2024	CKKS…	none	–	co-design	yes
ProteinTI (Chen et al., 2024)	2024	CKKS (leveled, no bootstrapping)…	none	no	fixed	yes
ARION (Yang et al., 2025)	2025	CKKS (Lattigo v6.1.1)…	none	–	fixed	yes
EncryptedLLM (de Castro et al., 2025)	2025	CKKS approximate FHE in RNS form…	none	no	fixed	yes
Encryption-Friendly LLM (Rho et al., 2025a)	2025	CKKS…	none	–	co-design	yes
ENSI (He et al., 2025)	2025	RNS-CKKS only…	none	no	co-design	yes

Table 4 continued from the previous page.

Record	Year	Primitive	Net	Off	Adaptivity	LLM
HE-LoRA (Frery et al., 2025)	2025	RLWE-based HE "similar to TFHE"…	none	no	fixed	–
NEXUS (Zhang et al., 2025a)	2025	RNS-CKKS FHE with bootstrapping…	LAN+WAN	no	fixed	yes
Powerformer (Park et al., 2025)	2025	RNS-CKKS…	none	–	co-design	yes
THOR (Moon et al., 2025)	2025	CKKS…	none	–	fixed	yes
TSP-Token-Order (Rho et al., 2025b)	2025	CKKS…	none	–	co-design	yes
AEGIS (Gong et al., 2026)	2026	CKKS (RNS, with bootstrapping)…	none	no	compiler	yes
ATLAS (Xie et al., 2026)	2026	CKKS…	none	no	compiler	yes
Bootstrap-TI (Xiao et al., 2026)	2026	RNS-CKKS FHE with functional…	none	no	co-design	yes
Cachemir (Yu et al., 2026)	2026	RNS-CKKS FHE with bootstrapping…	none	no	fixed	yes
CGF-Softmax (Park et al., 2026a)	2026	CKKS (desilofhe Python library)	none	–	co-design	yes
FHE-Scaling (Park et al., 2026b)	2026	CKKS (HEaaN2 library)…	none	no	co-design	yes
HEAT (Zirilli et al., 2026)	2026	CKKS…	none	–	co-design	yes
Inhibitor-TFHE (Brännvall et al., 2026)	2026	TFHE…	none	no	co-design	yes
MOAI (Zhang et al., 2026b)	2026	CKKS FHE with bootstrapping…	none	no	fixed	yes
PowerSoftmax (Zimerman et al., 2026)	2026	CKKS approximate HE…	–	–	co-design	yes
TFHE-GPU-NL (Chen et al., 2026b)	2026	TFHE…	none	no	manual	yes
Hybrid secure inference (25 records)						
Slalom (Tramèr & Boneh, 2019)	2019	TEE (Intel SGX enclave)…	none	no	fixed	–
2PC-LTH (Nawaz et al., 2020)	2020	2PC additive secret sharing…	WAN	yes	fixed	–
DarKnight (Hashemi et al., 2021)	2021	Intel SGX enclave…	–	–	manual	–
Iron (Hao et al., 2022)	2022	BFV HE…	LAN	no	fixed	yes
Primer (Zheng et al., 2023)	2023	Additive/packed HE…	LAN	yes	fixed	yes
BOLT (Pang et al., 2024)	2024	BFV HE…	LAN+WAN	–	manual	yes
Comet (2PC) (Xu et al., 2024)	2024	2PC additive secret sharing (ASS)…	none	no	co-design	yes
KV-Shield (Yang et al., 2024)	2024	No cryptography: random…	–	yes	fixed	yes
Nimbus (Li et al., 2024b)	2024	RLWE-based additive HE…	LAN+WAN	no	manual	yes
PermLLM (Zheng et al., 2024)	2024	Additive secret sharing…	LAN+WAN	yes	manual	yes
BLB (Xu et al., 2025)	2025	CKKS HE for fused linear operators	LAN+WAN	no	manual	yes
BumbleBee (jie Lu et al., 2025)	2025	2PC additive secret sharing…	LAN+WAN	no	manual	yes
CENTAUR (Luo et al., 2025)	2025	Random permutation matrices…	LAN+WAN	no	fixed	yes
CipherPrune (Zhang et al., 2025c)	2025	BFV HE for linear layers…	LAN+WAN	no	input	yes
East (Ding et al., 2025)	2025	2-out-of-2 arithmetic secret…	LAN	yes	manual	yes
FastLMPI (Chen et al., 2025)	2025	BFV HE…	LAN+WAN	no	fixed	yes

Table 4 continued from the previous page.

Record	Year	Primitive	Net	Off	Adaptivity	LLM
Fission (Ugurbil et al., 2025)	2025	M-party additive secret sharing…	none	no	manual	yes
SecureInfer (Nayan et al., 2025)	2025	Intel SGX (SGX2) enclave…	none	no	fixed	yes
SpecPrivGPT (Li et al., 2025b)	2025	Lattice-based additive HE under…	LAN+WAN	no	input	yes
Bifrost (Chen et al., 2026a)	2026	CKKS FHE on a GPU accelerator…	none	yes	manual	yes
CipherGPT (Hou et al., 2026)	2026	2PC additive SS over $\mathbb{Z}_{2^l}$ + OT…	LAN	no	fixed	yes
CryptoGen (Zhang et al., 2026a)	2026	Leveled BFV homomorphic encryption…	none	no	manual	yes
EncFormer (Zhu et al., 2026)	2026	RNS-CKKS FHE on GPU…	LAN+WAN	no	compiler	yes
KS-Reduce-2PC (Yang et al., 2026)	2026	RNS-CKKS approximate FHE…	none	yes	environment	yes
STIP (Yuan et al., 2026)	2026	No cryptographic primitives…	LAN+WAN	no	fixed	yes
Frameworks, compilers and primitives (27 records)						
BFV (Fan & Vercauteren, 2012)	2012	The canonical BFV scheme…	–	–	–	–
BGV (Brakerski et al., 2012)	2012	BGV leveled FHE from LWE…	–	–	–	–
ABY (Demmler et al., 2015)	2015	Mixed-protocol 2PC over Arithmetic	LAN+WAN	yes	manual	–
FSS (Boyle et al., 2015)	2015	Function secret sharing	–	–	–	–
FSS-Improvements (Boyle et al., 2016)	2016	FSS and distributed point…	–	–	–	–
CKKS (Cheon et al., 2017)	2017	RLWE-based leveled approximate…	–	–	–	–
SecureML (Mohassel & Zhang, 2017)	2017	2-out-of-2 additive secret sharing…	LAN+WAN	yes	manual	–
ABY3 (Mohassel & Rindal, 2018)	2018	2-out-of-3 replicated secret…	LAN+WAN	yes	manual	–
GAZELLE (Juvekar et al., 2018)	2018	Packed additively homomorphic…	LAN	yes	manual	–
HyCC (Büscher et al., 2018)	2018	Mixed-protocol 2PC over arithmetic…	LAN+WAN	yes	compiler	–
EzPC (Chandran et al., 2019)	2019	Arithmetic secret sharing…	LAN+WAN	no	compiler	–
FSS-Preprocessing (Boyle et al., 2019)	2019	FSS for offset function families…	none	yes	–	–
CrypTFlow (Kumar et al., 2020)	2020	Secret sharing over rings…	LAN	no	compiler	–
CrypTFlow2 (Rathee et al., 2020)	2020	2-out-of-2 additive secret sharing…	LAN+WAN	no	manual	–
Delphi (Mishra et al., 2020)	2020	Linearly homomorphic encryption…	WAN	yes	co-design	–
MP-SPDZ (Keller, 2020)	2020	Secret sharing…	LAN+WAN	yes	manual	–
ABY2.0 (Patra et al., 2021)	2021	Mixed-protocol 2PC over 64-bit…	LAN+WAN	yes	manual	–
CrypTen (Knott et al., 2021)	2021	N-party additive (arithmetic)…	none	no	fixed	–
SiRnn (Rathee et al., 2021)	2021	2PC additive secret sharing…	LAN	no	fixed	–
Cheetah (Huang et al., 2022)	2022	Lattice-based HE…	LAN+WAN	no	fixed	–
LLAMA (Gupta et al., 2022)	2022	2PC masked/additive secret sharing…	LAN+WAN	yes	fixed	–
MOTION (Braun et al., 2022)	2022	Arithmetic GMW over rings	LAN+WAN	yes	manual	–

Table 4 continued from the previous page.

Record	Year	Primitive	Net	Off	Adaptivity	LLM
Pika (Wagh, 2022)	2022	2-out-of-2 additive secret sharing…	LAN+WAN	no	fixed	–
Piranha (Watson et al., 2022)	2022	Linear secret sharing (LSSS)…	LAN+WAN	no	fixed	–
Grotto (Storrier et al., 2023)	2023	(2,2)-additive secret sharing…	none	yes	fixed	–
SecretFlow-SPU (Ma et al., 2023)	2023	Arithmetic secret sharing…	LAN+WAN	no	compiler	–
Rhombus (He et al., 2024)	2024	RLWE HE…	LAN+WAN	no	manual	–
Analysis, systematization and attacks (16 records)						
SoK-HE-DL (Podschwadt et al., 2022)	2021	Surveys BGV	none	–	–	–
MPC-Char (Wang et al., 2022)	2022	2PC additive (arithmetic) + binary…	LAN	no	fixed	yes
SoK-CryptoNN (Ng & Chow, 2023)	2023	Systematizes 53 crypto-only…	LAN+WAN	–	–	–
Depth-Inversion (Dong et al., 2025a)	2025	No cryptography	–	–	–	yes
Hidden No More / Cascade (Thomas et al., 2025)	2025	Hidden-state reconstruction…	LAN	–	fixed	yes
No-Free-Lunch PPI (Zhang et al., 2025b)	2025	No cryptography	–	–	–	yes
PPML-Systematic-Review (Zeng et al., 2025b)	2025	Surveys garbled circuits	LAN+WAN	yes	–	yes
ActInv/PriPert (Fan et al., 2026)	2026	No cryptography	none	–	–	yes
PPLLM-Survey (Andreoletti et al., 2026)	2026	Surveys MPC…	none	–	–	yes
PrecompNoise-Attack (Saini et al., 2026)	2026	Attack on non-cryptographic…	–	–	–	yes
Shuffling (In-)Security (Li et al., 2026)	2026	Attack on linear-only-encryption…	none	–	–	yes
SoK-CKKS-LLM (Badawi et al., 2026)	2026	CKKS approximate HE	–	–	–	yes
SoK-ConfTI (Yaluhin, 2026)	2026	Systematizes five mechanism…	LAN+WAN	yes	–	yes
SoK-CPCL (Capano et al., 2026a)	2026	Systematizes MPC…	LAN+WAN	yes	–	–
SoK-FHE-AI (Xue et al., 2026)	2026	FHE only: word-wise BGV/BFV/CKKS…	none	no	–	–
SoK-PTI (Chen et al., 2026c)	2026	Survey of HE (CKKS/BFV/BGV/TFHE)	LAN+WAN	yes	–	yes
Trusted-execution track (8 records)						
H100-CC-Bench (Zhu et al., 2024)	2024	NVIDIA Hopper GPU…	–	–	–	yes
Petridish (Li et al., 2024a)	2024	AMD SEV-SNP confidential VM…	none	no	fixed	yes
CC-LLM-CPU-GPU (Chrapek et al., 2025)	2025	Intel TDX confidential VMs	–	–	–	yes
PipeLLM (Tan et al., 2025)	2025	No cryptographic-inference…	–	–	–	yes
TEESlice (Li et al., 2025a)	2025	Intel SGX enclave with an…	–	no	co-design	yes
EnclaveX (Schambach et al., 2026)	2026	Intel TDX confidential VM with an…	–	–	fixed	yes
H100-TDX-Bench (Wang et al., 2026)	2026	Intel TDX confidential VM hosting…	–	–	–	yes

Table 4 continued from the previous page.

Record	Year	Primitive	Net	Off	Adaptivity	LLM
OpenPCC (Zhou et al., 2026)	2026	Intel TDX confidential VM…	–	–	–	yes
Differential-privacy track (3 records)						
DP Foundations (Dwork & Roth, 2014)	2014	Differential privacy: randomized…	–	–	–	–
DP LM fine-tuning (Li et al., 2022)	2022	DP-SGD and DP-Adam…	–	–	–	–
PrivacyRestore (Zeng et al., 2025c)	2025	No cryptography: client-side…	none	–	–	yes
Benchmarks and selection tooling (2 records)						
Prismo (Njungle et al., 2025)	2025	Catalogues 74 PPML frameworks…	none	no	environment	–
SMPC-vs-FHE-Bench (Taubert et al., 2026)	2026	Benchmarks secret-sharing MPC…	none	no	–	–

Codes describe a reported capability, not an absence. Each row records eleven fields, the primary URL and a verification confidence, coded from the paper itself. Unstated properties read `unclear` or `none-reported` and are counted, not imputed. A code states that a paper reports a capability; it is not proof that other mechanisms are absent. Coding was AI-assisted and source-checked, without independent dual human coding. Counts are therefore descriptive: they give neither confidence intervals nor the fraction of all papers in the field. Table 4 lists all 128 records with the canonical fields coded for each; the per-record party counts, threat models and nonlinear methods are in the released corpus file (Appendix E), which is authoritative for every coding the review reports.

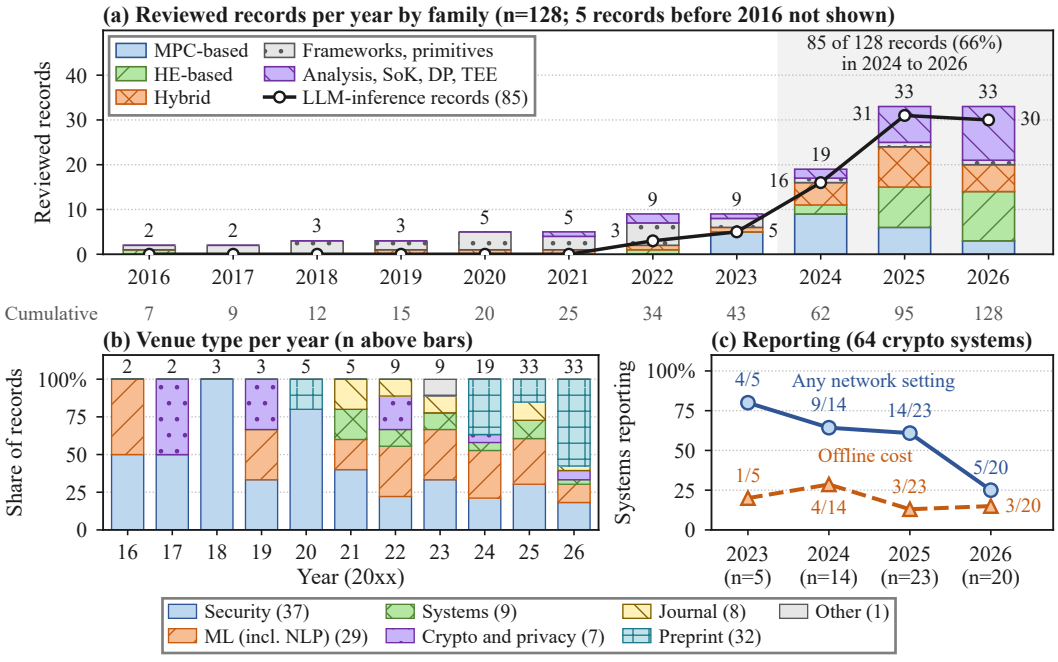


Figure 5: The reviewed corpus over time (record counts, purposive sample). (a) Records per year by family group ($n = 128$; five pre-2016 records not shown; benchmark records counted with frameworks and primitives), the 85 LLM-inference records per year (black line), cumulative corpus (grey row); 85 of 128 records (66%) fall in 2024 to 2026 (shaded). (b) Venue-type shares per year, n above bars; NLP folded into ML, legend counts span 2016 to 2026. (c) Share of the 64 cryptographic LLM systems reporting any network setting or offline cost (k/n labels), years with at least five systems; not-applicable rows stay in the denominator.

Table 5: Each mechanism imposes different comparison conditions; bottlenecks are design-dependent and no row dominates another. DP is a separate security track, not an execution backend.

Track	Representative evidence	Limitation and benchmark consequence
MPC-based	PUMA; SIGMA (Dong et al., 2025b; Gupta et al., 2024)	Interaction and preprocessing can dominate; compare only matched parties, corruption thresholds, and numeric precision. FSS is an MPC technique, not a family; it can shift cost into correlated randomness, so declare dealer trust and price offline material.
HE-based	CryptoNets; NEXUS; THOR (Gilad-Bachrach et al., 2016; Zhang et al., 2025a; Moon et al., 2025)	Encrypted arithmetic, key and memory costs, and non-linear approximation constrain execution; match security, packing and model visibility.
Hybrid	Iron; BOLT; BumbleBee; BLB; Slalom; PermLLM; STIP (Hao et al., 2022; Pang et al., 2024; jie Lu et al., 2025; Xu et al., 2025; Tramèr & Boneh, 2019; Zheng et al., 2024; Yuan et al., 2026)	Conversion and truncation can erase kernel-level savings: price directional boundaries and verify representation correctness. Permuting alone preserves value invariants, so name the observing parties, collusion assumptions, and any added protection such as a TEE, whose integrity, side-channel and transition assumptions must be stated.
DP track	DP foundations; private language-model learning (Dwork & Roth, 2014; Li et al., 2022)	Noise and privacy-budget choices affect utility; training-record privacy alone does not hide inference prompts. Specify adjacency and releases.

This appendix expands Section 2's observations with record-level evidence: one block per family, then adaptivity.

Privacy mechanisms answer different questions. Each mechanism protects a different observation: DP bounds what a release reveals under neighboring-data changes (Dwork & Roth, 2014), but alone does not hide a prompt from an inference server; a confidential execution does not bound what a released answer reveals about training records. Note that DP is not synonymous with unusable accuracy: pretrained language models substantially improve private learning (Li et al., 2022). Each comparison unit must name a protected observation and an adversarial model, not a scalar "privacy level".

MPC-based: FSS moves work into correlated randomness. MPC protocols exchange messages, so their cost follows communication volume and interaction rounds. Function secret sharing (FSS) is an MPC technique, not a family of its own: it instantiates the gates secret sharing handles badly, and FSS for a gate's offset family is exactly correlated randomness for a one-round online phase (Boyle et al., 2019). The trade-off is explicit: SIGMA's cheap online phase is bought with dealer-supplied material that cannot disappear when its online GPU latency is compared with others (Gupta et al., 2024).

HE-based: interaction falls, the deciding axis goes unreported. HE can reduce interaction: NEXUS evaluates encrypted Transformer inputs without intermediate client communication, and THOR develops packed matrix operations and adaptive numerical approximations (Zhang et al., 2025a; Moon et al., 2025), yet both still require measuring encrypted arithmetic, numerical error, and resource costs. Note that the family rarely reports the axis on which it is strongest: experiments confirm that non-interactive HE wins where bandwidth is scarce and latency high (Chen et al., 2026c; Ng & Chow, 2023), yet 20 of our 22 HE-based Transformer records report no network setting and NEXUS alone reports LAN and WAN.

Hybrid: crossing representations is work, and permutation needs a leakage contract. BLB's profiling identifies conversion and truncation as substantial communication sources in BumbleBee and reduces boundaries through operator fusion (Xu et al., 2025; jie Lu et al., 2025). BOLT names automated, network-dependent linear-protocol selection as future work (Pang et al., 2024). A benchmark must measure whole execution graphs, boundary costs included. Note that we do not pool reported cross-paper speedups. This is because model versions, threat assumptions, hardware, input lengths, and offline accounting differ across papers. Here "matrix pseudorandom permutation"

denotes a secret, possibly pseudorandomly generated index permutation given by a permutation matrix; it implies no cryptographic PRP guarantee for whole tensors. PermLLM, CENTAUR, and STIP assign visibility and trust differently; the final STIP version includes TEE protection (Yuan et al., 2026). Thomas et al. (2025) show why access to hidden states and weights matters, but do not establish that every permutation protocol, party assignment, or later version is broken. With access to shuffled intermediate activations, Li et al. (2026) show alignment and model-weight extraction on Pythia-70m and GPT-2. This is a visibility-specific attack, not a break of every protocol that uses permutations. The benchmark must preserve these distinctions and cannot treat permutation-obfuscated tensors as ordinary semantically secure ciphertexts.

Adaptivity is multidimensional. MPCFormer and an encryption-friendly architecture adapt models and approximations before serving (Li et al., 2023; Rho et al., 2025a). Ditto combines static quantization with dynamic ring dispatch and automatic type conversions (Wu et al., 2024). CipherPrune and SecDTD perform input-aware token reduction inside secure inference (Zhang et al., 2025c; Cai et al., 2026). These mechanisms complement environment-aware protocol selection; they are not evidence of a wholly static field. The benchmark question is narrower and testable: under a fixed security contract, when do deployment or workload changes justify a different operator-to-backend assignment once conversion, precision, and switching costs are included? Answering it requires common workloads, measured cost profiles, controlled ablations, and regret against feasible reference plans.

B Benchmark architecture and representation contract

Three objects of decreasing scope. Figure 6 gives the task-level view: a request and its contract enter the typed registry, the planner prices directional conversions, and the controller switches plans at token boundaries, with every run returning a record. Figure 7 separates the same three objects by tier and shows which campaign produces each evidence class (NATIVE, E2E, SF, ANALYTIC). Measurements flow back into planning only as the fitted cost model and range-violation flags. Every record carries an origin (author artifact, local reimplementation or reported-only) and a scope (analytic, functional, microbenchmark, local protocol or deployed model). In the harness, party 0 (the server) owns the model weights and party 1 (the client) holds token ids and keys and learns the outputs; the dealer is emulated in-process with a common seed; layer profiles use Beaver or FSS gates; the LM head runs in the weight-mask lane on $\mathbb{Z}_{2^{64}}$ shares; the KV cache stays as shares and never migrates at a switch. The ANALYTIC chip holds modeled costs of large models never executed and has no measured input. Note that neither a registry check nor a drawn box establishes a guarantee.

A conversion is a protocol, not a cast. An adapter descriptor includes its input/output representation, party ownership, modulus, signed interpretation, fractional precision, HE scheme and level, layout, secret/public dimensions, error contract, material lifecycle and admissible threat profiles. Two tensors of the same shape but different rings or encryption schemes are not interchangeable. A conversion is therefore a protocol with a pre- and a postcondition, not a cast between arrays, and a hybrid pipeline pays at every HE/MPC boundary. One generic lift rule fails (Appendix C), showing why the distinction must be checked independently of latency.

C Conversion contracts and implementation audit

Break-even for a constant regime. **Proposition (constant-regime break-even).** If staying costs c_p per epoch, switching costs $K \geq 0$ once and then c_q per epoch, and these costs remain constant for n epochs, switching strictly improves total cost iff $n(c_p - c_q) > K$. For $c_p > c_q$, the first profitable integer horizon is $\lfloor K/(c_p - c_q) \rfloor + 1$. *Proof.* Subtract $K + nc_q$ from nc_p. □

The proposition motivates Eq. (3) but guarantees nothing under incorrect cost or dwell predictions; the uncertainty allowance is heuristic, not a calibrated confidence bound.

The supplied modulus conversion has a deterministic counterexample. The native conversion module (Appendix E) implements A2H, the arithmetic-to-HE conversion a hybrid pipeline needs at its HE linear layers, in three steps: each party maps its $\mathbb{Z}_{2^{64}}$ share to its signed integer interpretation

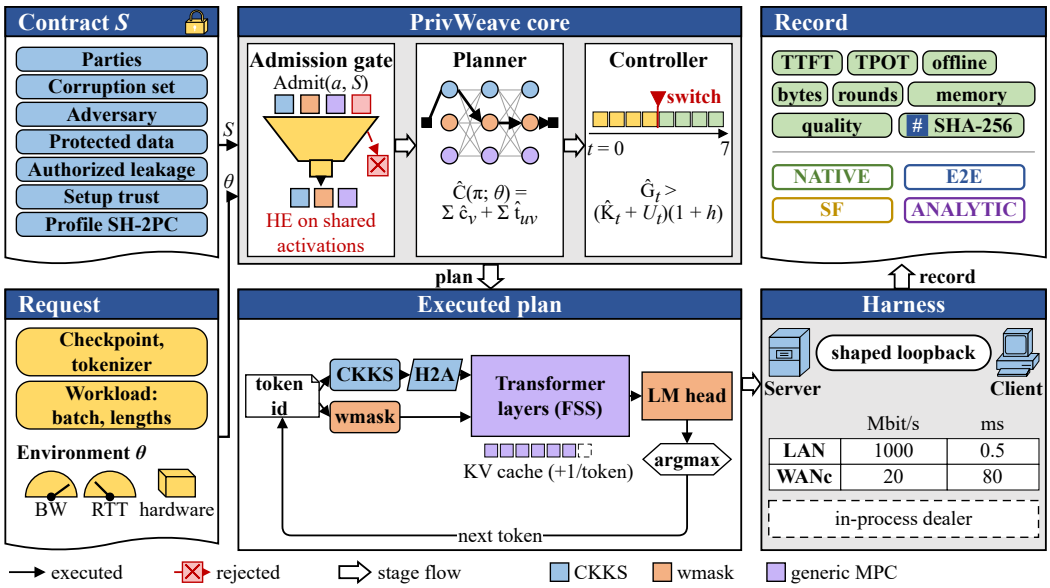


Figure 6: PRIVWEAVE overview. Contract S and request (environment θ: bandwidth BW, RTT, hardware) enter the core: admission $\mathsf{Admit}(a, S)$ drops HE on shared activations; the planner prices operators and conversions (black: cheapest admitted chain); the controller switches only at token boundaries under Eq. (3) (marker: switch at epoch 4, before output token 5, of eight). Lane colours per legend; H2A applies to the CKKS lane; the KV cache gains one share cell per token. Plans run on the two-process shaped-loopback harness (server: weights; client: token ids, keys; table: the two link profiles; dashed: in-process dealer). Records carry cost, quality, SHA-256 provenance and an evidence class.

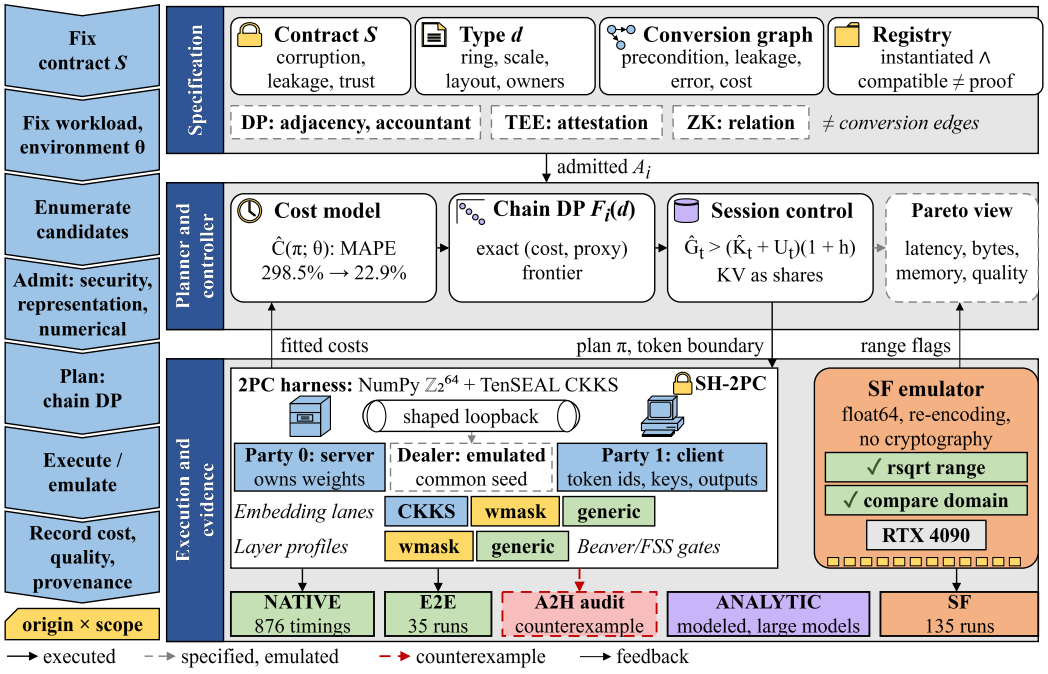


Figure 7: PRIVWEAVE as a stack. The chevron rail is the workflow, ending in the origin-by-scope tag. Dashed strata are comparison strata, not conversion edges; the session card abbreviates Eq. (3) (policy-pair arguments dropped). The execution tier holds the semi-honest harness (party chips name each party's data) and the GPU-shaped emulator: check marks are its two range checks (rsqrt range, comparison domain), the yellow strip its per-run flag row. Only fitted costs and range flags feed back.

and reduces it modulo the BFV plaintext prime $t = 68720050177$; the two residues are added homomorphically. Let $q = 2^{64}$, let $\widetilde{s}_i$ be the signed interpretation of share s_i, and x that of $(s_0 + s_1) \bmod q$. There is an integer $k \in \{-1, 0, 1\}$ such that

$$\widetilde{s}_0 + \widetilde{s}_1 = x + kq. \tag{5}$$

The routine therefore returns an encryption of $(x + kq) \bmod t$, which equals $x \bmod t$ iff $kq \equiv 0 \pmod t$. This is because t is an odd prime, so $\gcd(q, t) = 1$, and for the stated range of k equality requires $k = 0$. The condition $|x| < t/2$ does not guarantee $k = 0$.

For example, $s_0 = s_1 = 2^{63} - 1$ is a valid sharing of $x = -2$. The source routine yields residue 1016072382, whereas the required residue is $t - 2 = 68720050175$; their modular difference is $q \bmod t = 1016072384$. The analogous negative signed-wrap case also fails, and both secrets meet the routine's stated small-value condition; Figure 3 draws this edge as failing. Note that uniform random shares of small secrets seldom hit these narrow wrap boundaries, so ordinary random correctness tests can miss it. We do not measure failure frequency in a deployed workload.

The audit reproduces the defect with real BFV encryption. The audit script (Appendix E) invokes the unchanged original client and server functions under real TenSEAL BFV encryption. It verifies two signed-wrap counterexamples and two nonwrapping controls, then checks that the original type checker rejects arithmetic-to-CKKS and arithmetic-to-BFV-to-CKKS plans. The audit record (Appendix E) records the residues, rejection reasons, and SHA-256 hashes of all four relevant source files. This audit is a correctness reproduction, distinct from the inherited session measurements, and does not modify the original artifact. A general conversion would need a secure lift/carry correction or a justified restricted sharing contract, followed by a compatible BFV consumer; the benchmark does not assert that such a correction is implemented.

H2A is also parameter-specific. The executed CKKS-to-arithmetic routine adds a wide plaintext mask and rounds the decrypted masked value, with mask slack capped at 20 bits in the source. This is not the active-profile, ciphertext-noise-flooding construction described elsewhere in the manuscripts. A deployment guarantee would need bounds on each secret, the mask range, the CKKS error, accumulated statistical leakage across slots and calls, and circuit privacy where required. A recorded parameter object is not a proof of these properties, so our benchmark keeps the observed conversion cost separate from such unvalidated guarantees.

The exact-chain argument covers serialized prefixes only. For Eq. (2), induct on prefix length. Every feasible length-i plan consists of a feasible length-$(i-1)$ prefix, one valid conversion, and one admitted operator; minimizing over exactly these choices proves the recurrence. With an additive error proxy, a prefix can be discarded only if another prefix with the same future-relevant state has no higher cost and no larger proxy. Note that this reasoning does not permit merging states with different live tensors, installed material, ciphertext levels, or correlated approximation effects. Nor does a serialized chain make a branching execution graph globally optimal: shared conversion reuse, joins, parallel schedules, and peak memory can couple nonadjacent decisions.

Contracts do not collapse distinct privacy goals. DP evaluation must identify whether adjacency is over training records, prompts, or another unit, and track repeated releases under an accountant; a DP utility curve is not a communication-latency comparison with an MPC-based system. Similarly, a matrix permutation or invertible transform (Zheng et al., 2024; Yuan et al., 2026; Luo et al., 2025) may preserve statistics or invariants unless additional protection is supplied, so its descriptor must state what is hidden under which attack model. Note that Section 2 requires the privacy contract to be matched before the cost, and labels the heuristic tier that prompt-reconstruction and weight-extraction attacks reach (Thomas et al., 2025; Zhang et al., 2025b). A TEE route requires a permitted hardware trust boundary, while a ZK route specifies the proved relation and any exposed statement. Neither can be admitted solely because its modeled cost is lower than an MPC-based or HE-based backend's.

D EVALUATION MATRIX: EXECUTED AND CONSTRAINED MEASURED AXES

Requirements and completed results. In Table 6, each axis's implementation category carries corresponding experimental requirements; inherited evidence is presented precisely within its actual scope in the last column.

Table 6: Evaluation matrix: the axes executed here with their campaign block, the local axes under constrained-resource measurement, and the axes remaining for server-side equivalent emulation. The driver script of each server-side row is listed in Appendix E; the middle block gives the required wall-clock cost.

Axis	Required variation and test	Measured evidence chain
Executed in this revision		
Workload	Encoder classification and decoder perplexity, contexts 64–1024; whether operator preferences transfer across task families	Six causal models and three SST-2 classifiers: the accuracy-extension perplexity and classifier blocks (driver: Appendix E)
Quality	Same checkpoint, tokenizer, slice and precision; predeclared thresholds and range flags	72 evaluations, three rounding seeds, 16 fractional bits, GELU degree 12 (Table 7)
Constrained-resource measurement		
Accounting	Cold start, warm start, amortization over $N \in \{1, 10, 100\}$ requests	Driver: Appendix E: about 0.7 h of harness wall at four layers on LAN, scaled from the recorded session cells
Statistics	At least five sessions per cell; fixed warm-up; no pseudoreplication	Driver: Appendix E; the session-replicate block completed both seeds, 20 cells $\times$ 2 seeds $=$ 40 runs ($n = 2$ per replicate cell, $n = 1$ for the frozen headline cells); each further seed costs 2.6 to 4.0 h of harness wall and raises n by one, so $n = 5$ needs 10 h
Server-side, equivalent emulation		
Model size	Two sizes beyond local memory; size-dependent ranking	Driver: Appendix E, under its RSS guard; 48.9 GB per process for gpt2-xl, extrapolated crudely from the frozen 12-layer GPT-2 decode record's measured 14.2 GB peak
Execution	HE-based, MPC-based and hybrid routes, each in its own stratum	Driver: Appendix E
Network	A physical two-host link, not a shaper	Driver: Appendix E; the shaper measures 762.1 s for the adaptive eight-token stable-WANc decode (session summary, Appendix E); the physical link is tested
Hardware	Two hosts, one held out from fitting; transfer error and realized regret	Driver: Appendix E; the two-host fit holds 22.9% held-out MAPE and 7.1–35.8% per lane

Ablations must keep their own accounting. Each ablation removes one ingredient from the decision while the measurement keeps every cost. (1) Drop conversion cost from the planning objective alone, then execute the plan with its actual conversions; otherwise the ablation hides its error. (2) Drop the uncertainty margin, not the accounting. (3) Compare stateful and stateless installation under one inventory. (4) Drop the security filter only as a rejected-plan diagnostic, never as an accepted insecure baseline. (5) Compare fitted and analytic costs on one admissible candidate set. (6) For numerical error, hold the task threshold fixed, vary the planner's proxy, and report task failure separately from optimization success.

Calibration must be tested on what it has not seen. Split by complete configurations; hold out model shapes, network regimes and physical hosts before fitting, keeping repeats of a configuration in one partition. Fit latency, bytes, rounds and setup separately, and inspect residuals per component as well as overall MAPE. Report rank errors near measured crossover boundaries and their regret. Note that an always-winning lane can make selection accuracy perfect despite a poor model, so

candidate pairs must also be evaluated without it. Do not infer transfer from repeated timings on one machine.

DP, TEE, and integrity are contracts, not conversion edges. DP records specify adjacency, privacy unit, mechanism, accountant, (ε, δ), privacy loss under repeated release, utility and attack-specific risk; TEE records the trusted computing base, attestation, side-channel assumptions and hardware generation; integrity records the statement verified, setup, soundness error and proof cost. They may augment confidential execution but are no arithmetic domain joined by a conversion edge, which is why the review keeps them as tracks, not families.

E Reproduction and evidence inventory

The analysis is deterministic given its hashed inputs. The package copies the numeric inputs it consumes and records their original paths and SHA-256 hashes; given those, the analysis is deterministic. This revision regenerated the figures, tables and statistics; the historical protocol runs were not re-executed. The bibliography rests on externally verified primary sources, not solely on inherited reference lists.

Commands. From the source-package directory, run the package scripts listed in the artifact index to verify the copied inputs and regenerate outputs, then compile the main document with latexmk; the build-check directory holds the build and rendering checks, and the artifact audit record (Appendix E) records which inherited results were retained, corrected or excluded. Paths here are package-relative; the originals are unchanged.

Artifact index. Package-relative names of every artifact the text, tables and captions refer to by plain name.

- Review data: the released corpus file `data/corpus_v4.csv` and the curated review statistics `data/review_statistics.json`; chart number sheets `data/review_properties_counts.json` (Figure 2) and `data/review_year_counts.json` (Figure 5).
- Reanalysis outputs: the computed summary `data/computed_summary.json`; the session summary `data/session_summary.csv` and the session trajectory `data/session_trajectory.csv`; the accuracy configurations `data/accuracy_configurations.csv`; the calibration configurations `data/calibration_configurations.csv` and the per-component calibration summary `data/calibration_component_summary.csv`; the audit record `data/conversion_contract_audit.json`; the provenance ledger `data/provenance.csv`; chart number sheets `data/accuracy_extension_summary.csv` (Figure 4), `data/session_trajectory_L12.csv` (Figure 9), `data/calibration_panel_summary.json` (Figure 11) and `data/sessions_panel_summary.json` (Figure 10).
- Campaign records copied under `data/runs_v4`: the accuracy-extension perplexity and classifier blocks `acc_ext_ppl` and `acc_ext_cls`, each with `campaign_extension.csv` and `summary_extension.json`; the session-replicate block `sess_rep` with `replicates_runs.csv` and `replicates_summary.json`; the campaign status record `campaign_status.json`.
- Run directories: `runs_v4/acc_ext_ppl`, `runs_v4/acc_ext_cls` and `runs_v4/sess_rep`.
- Experiment drivers under `scripts/experiments`: `run_accuracy_extension.py` (workload and quality axes), `run_session_replicates.py` (statistics), `run_session_chain.py` (accounting), `run_decode_scale.py` (scale), `run_two_host_program.py` (isolation), `run_cross_hardware_calibration.py` (hardware), `run_active_security.py` (malicious security),

`run_external_frameworks.py` (external frameworks) and `run_campaign_v4.py` (campaign orchestration).

- Table generators under `scripts`: `build_table_families.py` (Table 1), `build_table_setup.py` (Table 2), `build_table3_sessions.py` (Table 3) and `build_table7_acc_extension.py` (Table 7).

F Extended results and reanalysis details

Cost trajectory. Figure 8 places the four accounted timers within an epoch of the four-layer WANc-to-LAN trace; Figure 9 accumulates them. In panel (a), P1 and Adaptive cost alike while the link is constrained (480.58 against 481.29 seconds at token 4); afterwards Adaptive finishes at 516.30 seconds against 604.43 for P1 and 682.39 for P2. Panel (b) resolves the per-token increments: the prefill token costs 275 seconds (P1), 523 (P2) and 275 (Adaptive), of which 185, 435 and 185 seconds are installation and about 89 seconds execution; steady WAN-constrained tokens cost 68 to 70 seconds for P1 and Adaptive but 44 to 46 for P2; the switch at token 5 charges Adaptive 8.22 seconds of installation, after which its LAN tokens cost 6.1 to 6.2 seconds, level with P2's 5.9 to 6.2, while P1 stays at 29 to 32. Sensing plus planning peaks at 0.61 seconds per token at four layers (0.64 at twelve; under 0.06 on LAN). Panel (c) repeats the pattern on the twelve-layer trace: the prefill token costs 805.15, 1055.29 and 806.54 seconds, the switch 8.31, LAN tokens 34 to 36 (P1) against 12.8 to 13.0 (P2 and Adaptive), and the runs end at 1361.40 (P1), 1457.83 (P2) and 1285.39 seconds (Adaptive), 5.58% below P1 against 14.58% at four layers. Every epoch sum equals its run's accounted total. Note that these points expose the divergence without a variance model for correlated tokens.

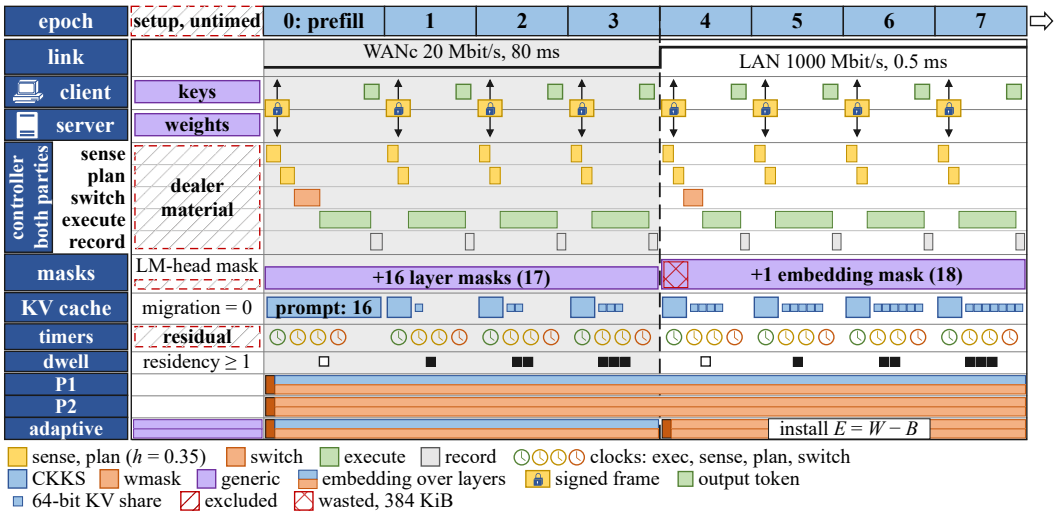
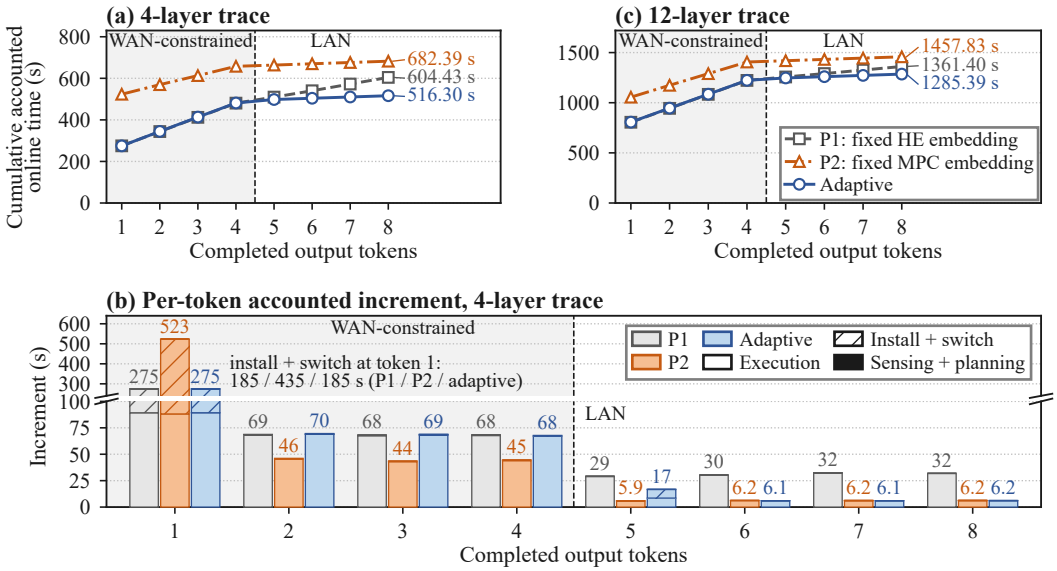


Figure 8: Session lifecycle on the four-layer WANc-to-LAN trace. Columns: untimed setup, epoch 0 (16-token prefill), epochs 1 to 7 (one token each); grey columns: constrained regime; dashed line: regime change. Lanes: shaped link; signed telemetry frames between the parties (green chips: output tokens); both parties' identical deterministic controller (challenger q against plan p, Eq. (3)); installed masks (discarded material cross-hatched); KV cache as $\mathbb{Z}_{2^{64}}$ shares; clocks (execution, sensing, planning, switch); dwell ticks (hollow: reset); two-band plan strips P1, P2, adaptive (embedding over layers; dark bars: charged installs). Hatched: excluded from the metric (dealer material, common setup, timing residual).

Execution configuration. The recorded native host (§5) runs Python 3.12.10, NumPy 2.5.2 and TenSEAL 0.3.17; numerical campaigns use the RTX 4090. Session settings are 16 fractional bits, 40 logical bits, a 32-bit DCF domain and statistical parameter 40; the CKKS embedding profile uses polynomial degree 8,192, coefficient-modulus bit sizes $(60, 40, 40, 60)$ and log scale 40. Shaping profiles (Mbit/s over RTT in ms) are LAN 1000/0.5, WAN 100/40, WAN-constrained 20/80, WAN-mid calibration holdout 50/20.

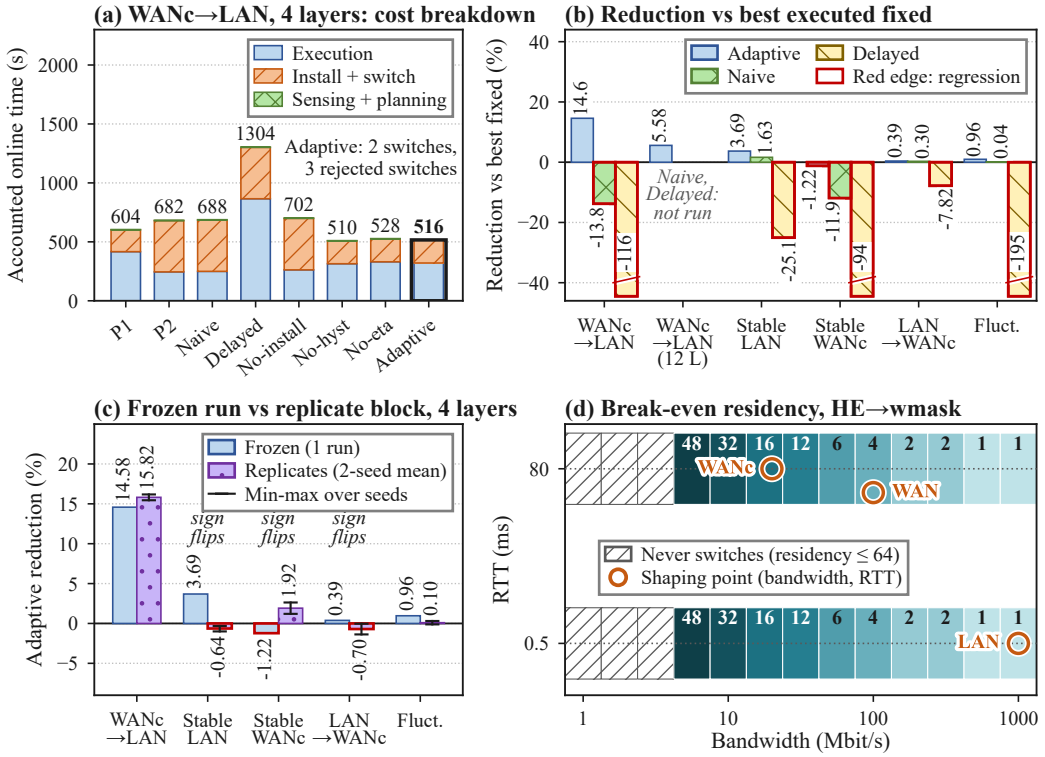


One run per policy; E2E runs on shaped loopback, dealer emulated in-process. Markers: observations; segments: guides. Sensing + planning peaks at 0.61 s per token (four layers) and 0.64 s (twelve), under 0.06 s on LAN: too thin to see.

Figure 9: Accounted online cost on the WANc-to-LAN trace, one run per policy (E2E protocol executions over shaped loopback, not process-isolated deployment). Shading marks the WAN-constrained regime and the dashed line the regime change before output token 5. (a) Cumulative accounted online time at the eight recorded token boundaries of the four-layer trace; the first point includes prefill and installation, exclusions as in Figure 8. (b) Per-token increment of the same runs, broken axis, split into execution, install + switch, and sensing + planning (peak 0.61 s, not visible). (c) The twelve-layer trace with final totals; legend shared with (a). Markers are observations; segments are guides, not fits; values are descriptive, not confidence intervals.

Accounted online metric. The session metric sums the execution, sensing, planning and switch timers, excluding dealer provisioning and common LM-head setup. Note that digest exchange and some bookkeeping fall between these timers. Across 35 runs, wall time minus dealer time minus this four-part metric leaves 0.0106–0.6516 seconds, at most 0.150% of accounted online time. We retain the original metric rather than change the denominator silently; released tables keep dealer-inclusive session times.

Ablations and replicates. Figure 10(a) lists every policy executed on the four-layer WANc-to-LAN trace. The fixed plans P1 and P2 total 604.43 and 682.39 seconds with one installation each. Naive re-planning, which ignores installation when it selects but pays it, totals 687.90 seconds (one switch, none rejected); delayed re-planning totals 1303.80 seconds and discards 536 MB of mask material. Of the ablations, no-install (installation dropped from the objective) totals 701.67 seconds with one switch; no-hysteresis totals 510.37 and no-eta 527.55 seconds, each with the same two switches and three rejected switches as Adaptive (516.30 seconds, 393 KB discarded), so on this single trace dropping hysteresis saves 5.94 seconds and dropping the uncertainty margin costs 11.25 seconds; on the fluctuating trace the three ablations lie within 2.1 seconds of Adaptive (222.99 to 225.18 against 223.10). Panel (b) states each policy's reduction against the best executed fixed plan: Adaptive gains 14.58% (four layers) and 5.58% (twelve layers) on WANc-to-LAN, 3.69% on stable LAN, 0.39% on LAN-to-WANc and 0.96% on the fluctuating trace, and loses 1.22% on stable WANc; naive re-planning loses 13.81% and 11.94% on WANc-to-LAN and stable WANc and stays within 1.63% of zero elsewhere; delayed re-planning loses on every trace (7.82% to 194.61%). Panel (c) sets the frozen adaptive margins beside the session-replicate block (two dealer seeds on the 32-logical-CPU host): the WANc-to-LAN margin holds at 15.82% (15.45% and 16.18% per seed) and the fluctuating margin shrinks to 0.10%, but three margins reverse sign, stable LAN from $+3.69\%$ to -0.64%, stable WANc from -1.22% to $+1.92\%$ and LAN-to-WANc from $+0.39\%$ to -0.70%, so the small-margin cells carry no sign. Panel (d) is the modeled break-even map of the native cost model (312 decisions): for each bandwidth and RTT it gives the smallest residency



Four-layer GPT-2 prefix unless stated; one run per frozen policy; (c) replicate block: two seeds per cell; E2E protocol executions over shaped loopback with in-process dealer emulation, not process-isolated deployment; dealer provisioning and common LM-head setup excluded; (d) is a modeled decision map, not a measurement; no confidence intervals.

Figure 10: Session case study: (a)–(c) E2E, (d) ANALYTIC modeled decision map. (a) Four-layer WANc-to-LAN accounted online time for fixed plans, baselines, ablations and Adaptive (outlined). (b) Reduction versus best executed fixed plan; red edges mark regressions, truncated bars labelled; 12-layer cell: Adaptive only. (c) Frozen adaptive reduction versus the session-replicate block (two-seed mean, min-max whiskers); three signs flip. (d) Smallest residency (epochs) installing the HE-to-weight-mask embedding conversion; hatched cells never switch within 64 epochs; hollow circles: LAN, WAN, WANc shaping points at their bandwidth/RTT coordinates; dotted lines: row RTTs.

at which switching the embedding from the client-input HE lane to weight-mask MPC pays back its installation. The residency falls from 48 epochs at 5.62 Mbit/s to 16 at 17.8, 4 at 100 and 1 at 562 Mbit/s and above; at 3.16 Mbit/s and below the switch never pays within 64 epochs; the 0.5 ms and 80 ms rows are identical, so bandwidth, not latency, sets the decision; the companion P3-to-P2 decision switches at residency 2 in all 26 cells and is not plotted. Per-run timers of all 35 frozen and 40 replicate runs, including P3, delayed re-planning and the three ablations, are in the session summary and the replicate run sheet (Appendix E).

New measurements in this revision. The campaign status record (Appendix E) lists three completed blocks, each with return code 0 and no unfinished cell. Block R1a (0.614 h) extends the quality grid to six causal language models at contexts 64, 256 and 1024 and writes 54 per-seed raw records, a 54-row campaign sheet and an 18-cell summary to the accuracy-extension perplexity block (Appendix E). Block R1b (0.029 h) extends it to three SST-2 classifiers at contexts 64 and 128, writing 18 raw records, an 18-row sheet and a 6-cell summary to the classifier block. Each carries a run manifest with the driver command, the environment and SHA-256 digests of every output and of the 35 artifact sources; both run the accuracy-extension driver listed in Appendix E in float64 on the RTX 4090. Note that both emulate the declared gadgets in plaintext under the predeclared contract: no cryptography executes, and seeds vary rounding only. Table 7 lists every cell; its Wall column times the scoring run only, so the cells sum to 0.554 h and 0.023 h, below the block totals that include model loading. Block R2 (6.659 h) is the session-replicate block: it repeats the four-layer session matrix (five traces, policies P1, P2, naive re-planning and Adaptive) under two dealer seeds, 20 cells $\times$ 2 seeds $=$ 40 runs, all complete, on the 32-logical-CPU host, with tokens identical

across seeds and matching the float reference in every run. Because host and seed change together, its magnitudes are not comparable with the frozen campaign, only its margins (Table 3).

Reanalysis script. The released script groups operator measurements by component, row count, input/output dimension and network, averages the three timings per group and computes percentage errors on the frozen test split, reproducing the source summary to within six-decimal CSV rounding. It also recomputes all 45 model/precision/degree quality outcomes from 135 seed-level rows and checks each session total against its raw epoch costs. The source snapshot records SHA-256 digests for 217 inputs; all 135 numerical-campaign raw-file references and the session manifest's copied output hashes match.

Evidence labels. The calibration files carry the broad label E2E; we narrow them to NATIVE because they measure operators, not complete models. The accuracy files, old and new, carry EMUL, mapped to SF. Large-model matrices and historical large-model cost models remain ANALYTIC, whatever their filenames. Relabeling must not silently change the type of evidence.

Per-lane calibration error. The fitted predictor keeps a long error tail: held-out MAPE is 7.14% for generic MPC linear operations, 35.79% for weight-mask MPC linear operations and 29.72% for the client-input HE linear lane (54 configurations each); remaining test configurations cover GELU, softmax, LayerNorm and argmax. Figure 11(c) shows those remaining components: the fitted predictor errs by 1.2% on GELU, 1.1% on LayerNorm and 0.5% on argmax (six, six and one configurations) but 33.9% on softmax (six), where the handwritten model errs by 48% to 780% per component and the robust predictor by 7% to 45%. Panel (d) splits the same 181 configurations by network: fitted MAPE is 45.6% on LAN, 21.1% on WAN and 14.5% on WAN-constrained (36 configurations each) and 16.7% on the WAN-mid holdout (73), 124 of 181 fitted-to-measured ratios (68.5%) lie inside the $\pm 25\%$ band, and every ratio lies between 0.33 and 2.84. Note that these costs may be combined only after input-domain and numerical-contract admission: a measured conversion does not establish correctness for all rings, cryptosystems, scales or later operators.

Extended contract. The extended contract stays separate from the source-declared original; these labels identify archived configurations, and independently attested preregistration is not asserted. The extension increases logical range and changes reciprocal-square-root normalization; its minimum-passing SmolLM2 point has +1.6125 perplexity by empirical measurement. The historical calibration route that decrypted a shared intermediate before CKKS reencryption is likewise superseded and excluded. All retained HE calibration rows consume client-owned inputs with zero A2H time, and all retained session transformer layers execute in MPC.

Reproduction. Running the chart scripts and number sheets listed in Appendix E verifies the frozen input hashes and regenerates the PDF/PNG figures, the per-configuration sheets and the computed summary; this reruns the analysis, not the original secure-inference campaign. Raw host manifests and source snippets expose same-host TCP shaping, trusted-dealer emulation, online/offline exclusions and the lack of checkpoint process isolation.

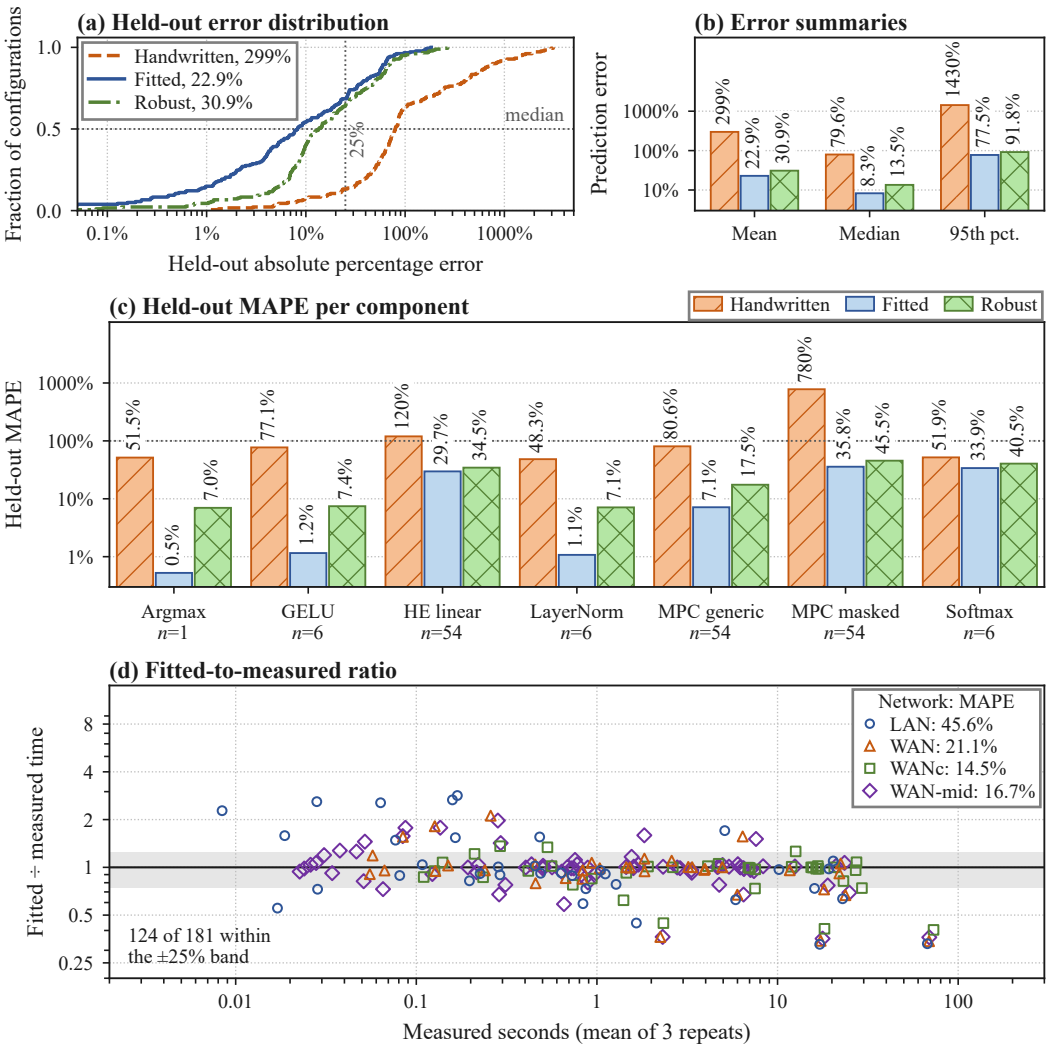


Figure 11: NATIVE prediction errors for the 181 held-out configurations (means of three repeats, same-host TCP with synthetic shaping). (a) ECDF of absolute percentage error per predictor, held-out MAPE in the legend; dotted guides mark 25% error and the median. (b) Mean, median and 95th percentile, log axis, bar legend as in (c). (c) Held-out MAPE per component with its held-out count n; dotted line at 100%. (d) Ratio of fitted modeled time to measured time against measured time, one hollow marker per network with its MAPE, line at ratio 1 and shaded ±25% band (124 of 181 inside). "Robust" is the uncertainty-aware upper predictor. Finite-grid summaries, descriptive, not confidence intervals.

Table 7: Accuracy extension at 16 fractional bits and GELU degree 12, three rounding seeds per cell. Δ is emulated minus plaintext (negative accuracy Δ is a drop; ∞: divergence); Agree: prediction agreement with plaintext. Budgets: +2.0 perplexity, 0.5 points of drop. Violation: rsqrt range, comparison (cmp) domain, both or none. Met: budget met (y/n); Viol.: violation. Shading: budget missed or any violation. Scored WikiText-2 tokens: 2,016, 2,040 and 2,046 at contexts 64, 256 and 1024. Wall: mean seconds per seed; dash: not applicable.

		Score		Δ over seeds		Agree	Contract		Wall
Model (params)	**Ctx**	**Plain**	**Emul.**	**Mean $\pm$ std**	**Max $\lvert\Delta\rvert$**	**(%)**	**Met**	**Viol.**	**(s)**
WikiText-2 perplexity (lower is better), budget +2.0									
GPT-2 (124M)	64	89.2213	89.5015	$+0.2802 \pm 0.0010$	0.2811	–	y	none	6.20
	256	47.0920	47.2435	$+0.1515 \pm 0.0005$	0.1520	–	y	none	1.90
	1024	23.2367	23.2781	$+0.0414 \pm 0.0008$	0.0422	–	y	none	1.90
SmolLM2 (135M)	64	51.3519	∞	∞	∞	–	n	both	14.76
	256	25.8682	∞	∞	∞	–	n	both	4.18
	1024	14.2529	∞	∞	∞	–	n	both	3.38
Qwen2.5 (0.5B)	64	39.7705	39.7090	-0.0615 ± 0.0324	0.0879	–	y	none	13.62
	256	22.4466	22.3843	-0.0624 ± 0.0101	0.0738	–	y	none	4.89
	1024	12.2046	12.2083	$+0.0037 \pm 0.0067$	0.0107	–	y	none	5.38
TinyLlama v1.1 (1.1B)	64	21.1603	21.8838	$+0.7235 \pm 0.0758$	0.7823	–	y	rsqrt	13.60
	256	11.5150	11.9357	$+0.4206 \pm 0.0352$	0.4610	–	y	rsqrt	112.34
	1024	7.2038	7.4828	$+0.2791 \pm 0.0078$	0.2868	–	y	rsqrt	230.43
Llama-3.2 (1B)	64	32.0531	33.1048	$+1.0517 \pm 0.0279$	1.0798	–	y	none	118.63
	256	15.4835	15.9404	$+0.4569 \pm 0.0088$	0.4627	–	y	none	12.73
	1024	8.6473	8.8700	$+0.2227 \pm 0.0037$	0.2261	–	y	none	110.00
Pythia (160M)	64	89.8711	89.9936	$+0.1225 \pm 0.0062$	0.1295	–	y	cmp	6.39
	256	46.3292	46.3836	$+0.0544 \pm 0.0058$	0.0594	–	y	cmp	2.06
	1024	24.0343	24.0209	-0.0134 ± 0.0078	0.0220	–	y	cmp	2.45
SST-2 accuracy (%, higher is better), budget 0.5 points of drop									
BERT-base (110M)	64	92.19	91.80	-0.39 ± 0.00	0.39	99.61	y	none	5.13
	128	92.19	91.80	-0.39 ± 0.00	0.39	99.61	y	none	5.13
DistilBERT (66M)	64	91.41	91.41	0.00 ± 0.00	0.00	100.00	y	none	2.65
	128	91.41	91.41	0.00 ± 0.00	0.00	100.00	y	none	3.09
RoBERTa-base (125M)	64	94.53	94.53	0.00 ± 0.00	0.00	100.00	y	none	5.84
	128	94.53	94.53	0.00 ± 0.00	0.00	100.00	y	none	5.85