

PRIVACY-PRESERVING RETRIEVAL-AUGMENTED GENERATION FOR SENSITIVE ENTERPRISE DATA: A ZERO-TRUST GOVERNANCE ARCHITECTURE

Daniel Marvin Imperial^{1*}, Marconi Pasini²

¹ London, School of Electronic Engineering and Computer Science

² University of Bath, Centre for AI

ABSTRACT

The recent introduction of retrieval-augmented generation (RAG), in which a parametric language model is conditioned on passages fetched from an external, non-parametric memory at inference time, offers enterprises a plausible route to grounding generative text systems in proprietary knowledge without retraining. Yet the same architecture that makes RAG attractive for knowledge-intensive tasks also creates a new and largely unexamined attack surface: the retrieval index itself becomes a queryable proxy for whatever sensitive records it was built from, and the boundary between “model” and “data” that traditional access control and perimeter security were built to police effectively dissolves. This paper develops, through a design-science methodology, a zero-trust governance architecture for deploying retrieval-augmented generation over sensitive enterprise corpora — financial records, health information, legal files, and internal communications. Drawing on NIST Special Publication 800-207, the differential privacy and federated learning literatures, and established enterprise data governance frameworks, we propose an architecture in which every retrieval call, embedding operation, and generation request is treated as an untrusted transaction subject to continuous authorization, attribute-based access control at the passage level, and privacy-preserving transformation prior to reaching the language model. We validate the design conceptually through STRIDE-based threat modeling and an enterprise scenario rather than a production deployment, since RAG systems were, at the time of writing, newly proposed and not yet operationally mature. The paper contributes a reference architecture, a mapping of zero-trust principles onto retrieval-specific controls, and a comparative analysis of privacy-enhancing technologies for retrieval pipelines, intended to guide both researchers and enterprise architects contemplating early adoption of this technology.

KEYWORDS: Retrieval-Augmented Generation, Zero-Trust Architecture, Differential Privacy, Enterprise Data Governance, Privacy-Enhancing Technologies

INTRODUCTION

Enterprises have spent two decades building data lakes whose main failure mode is under-use: employees cannot find the memo that would save an afternoon. Pretrained language models promised a remedy, but their parametric memory is frozen at training time and cannot “forget” a record or cite what an answer drew on (Petroni *et al.*, 2019).

Lewis *et al.* (2020) proposed retrieval-augmented generation (RAG): a dense retriever (Karpukhin *et al.*, 2020) paired with a generator conditioned on fetched passages; Guu *et al.* (2020) proposed the similar REALM, folding retrieval into pretraining itself point such a system at an internal wiki and obtain cited, continuously updated answers without retraining.

That appeal rests on an unexamined assumption: that the retrieval index and generation pipeline can be trusted like an organization's databases and file shares. There is reason to doubt it: an index built by encoding sensitive documents into dense vectors is not anonymized, since embeddings preserve semantic structure to support nearest-neighbor search. A generation model conditioned on retrieved passages will, absent an intervening control, reproduce fragments verbatim, turning any end user into a potential recipient of unauthorized information. And because retrieval and generation are typically separate services, the attack surface spans components not designed together and not subject to a database's audit discipline.

This is a governance problem masquerading as a machine-learning architecture, and perimeter security suits it poorly: it extends trust to requests inside the network boundary, while zero trust popularized around 2010, formalized by NIST as SP 800-207 in 2020, and reflected in peer-reviewed work on identity-centric network perimeters (Moubayed, Refaey, & Shami, 2019; Potharaju & Addanki, 2022) insists every request be evaluated on its own merits, continuously, regardless of location. RAG over sensitive enterprise data is a paradigmatic candidate: a query traverses an embedding service, a vector index, a reranker, and a generation model before a response reaches a screen (Rashid & Filgueirus, 2022).

The contribution is threefold. First, we synthesize four literatures developed largely in isolation RAG and neural information retrieval, zero-trust architecture, privacy-enhancing technologies, and enterprise data governance identifying the gap: no framework specifies how zero-trust principles apply to a retrieval pipeline. Second, we propose a reference architecture mapping the five NIST SP 800-207 tenets onto concrete controls for embedding, index querying, passage-level authorization, and output filtering (Rashid & Junck, 2023). Third, we validate the architecture through STRIDE-based threat modeling and an scenario, since RAG was months old and not yet deployed at scale.

LITERATURE REVIEW

RETRIEVAL-AUGMENTED GENERATION AND NEURAL INFORMATION RETRIEVAL

RAG's technical lineage runs through a decade of neural information retrieval, but its immediate ancestors are dense passage retrieval and retrieval-generation architectures

from early 2020. Karpukhin *et al.* (2020) showed that a dual-encoder retriever, mapping questions and passages into a shared embedding space, substantially outperforms classical sparse methods such as TF-IDF and BM25. This matters for enterprise applications because retrieval quality is a property of learned representations rather than index size — the retriever itself is a model capable of leaking what it learned from.

Lewis *et al.* (2020) then introduced RAG proper: a sequence-to-sequence generator (based on BART) conditioned on the top-k passages a dense retriever returns over a Wikipedia-derived index, updatable by swapping the index rather than retraining — unlike purely parametric models such as T5. Guu *et al.* (2020) proposed the concurrent REALM, folding retrieval into masked-language-model pretraining. Both treat the corpus as public, and neither addresses what changes once it is proprietary.

A complementary strand asks what pretrained models “know.” Petroni *et al.* (2019) probed masked language models such as BERT (Devlin *et al.*, 2019) with cloze-style queries and found they capture relational knowledge without explicit supervision — evidence that models function, imperfectly, as implicit knowledge bases. This clarifies what retrieval augmentation solves: parametric knowledge is diffuse and hard to audit, while a retrieval index is discrete and inspectable — but that inspectability makes it a liability, built to return its most similar content on demand. Brown *et al.* (2020), introducing GPT-3, similarly note large models can be prompted to reproduce memorized text, a concern compounded once conditioned on sensitive retrieved content.

ZERO-TRUST SECURITY MODELS

The zero-trust concept predates its NIST codification by roughly a decade, coined in industry security practice around 2010 against the mistaken assumption that traffic from inside the corporate perimeter merits more trust than traffic from outside. That formulation was largely a network-segmentation argument — treat every network, including internal ones, as untrusted — later given peer-reviewed substance in work on identity-centric network perimeters (Moubayed, Refaey, & Shami, 2019). NIST published Special Publication 800-207 (draft 2019, final 2020), generalizing this into an architectural philosophy: all data sources and services are resources; communication is secured regardless of location; access is granted per-session on dynamic policy reflecting the requester’s observable state; the enterprise continuously monitors asset integrity; and authentication and authorization are strictly enforced and continuously re-evaluated.

What SP 800-207 does not specify is how these tenets apply to machine-learning pipelines, still less to RAG, which had not been published when most of the document was drafted. Its worked examples concern conventional resources — file servers, databases — not what counts as a “resource” (the index? a passage?) or a “session”

when the requester is an intermediate service. The zero-trust literature has not answered these questions, because before Lewis *et al.* (2020) no widely known architecture combined a queryable index of sensitive content with a generative model producing free-text output in one pipeline.

PRIVACY-ENHANCING TECHNOLOGIES FOR ENTERPRISE DATA

A more mature literature addresses extracting statistical value from sensitive data without exposing underlying records. Differential privacy, formalized by Dwork and Roth (2014), guarantees a computation's output is statistically indistinguishable whether any single record was included, via calibrated noise injection — rigorous, but the noise needed for a meaningful privacy budget degrades utility in proportion to query sensitivity, unforgiving for tasks like passage retrieval where the point is one exact piece of text, not an aggregate. It was built for statistical release, not verbatim retrieval, and applying it naively degrades precision.

Federated learning, introduced by McMahan *et al.* (2017) and elaborated by Yang, Liu, Chen, and Tong (2019), keeps sensitive data distributed across its holders and communicates only model updates centrally, avoiding the centralization risk differential privacy does not address. But shared gradients are themselves an open attack surface: Zhu, Liu, and Han (2019) show an adversary observing gradient updates can, through optimization, reconstruct the training examples with substantial fidelity — undercutting the assumption that “the data never leaves the device” suffices alone.

Homomorphic encryption, surveyed by Acar, Aksu, Uluagac, and Conti (2018), permits computation directly on encrypted data. Fully homomorphic schemes support arbitrary computation but impose overhead orders of magnitude greater than plaintext, particularly for dense retrieval's similarity search; partial schemes trade generality for tractability but still poorly fit retrieval's operations. Sweeney (2002) offers a cheaper approach, k-anonymity, generalizing identifying attributes so every record is indistinguishable from at least k-1 others — inexpensive, but designed for structured data and ill-suited to free text.

DATA GOVERNANCE FRAMEWORKS AND ACCESS CONTROL

Enterprise data governance, distinct from information security, concerns the allocation of decision rights over data assets — who may create, modify, use, and retire a category of data. The organization of governance into five domains principles, quality, metadata, access, and lifecycle argues that organizations frequently conflate IT governance with data governance, failing to assign accountability for who may grant access, distinct from who administers infrastructure. This matters because a retrieval-augmented system introduces a new asset, the index fitting none of an organization's governance domains (Karim & Elias, 2022).

Access control models supply the mechanical layer beneath governance policy. Role-based access control (Siddiqui *et al.*, 2023), but is coarse: a role either can or cannot access a document category, with no way to condition access on finer attributes. Attribute-based access control, formalized by Jin, Krishnan, and Sandhu (2012) and codified as NIST Special Publication 800-162, generalizes this by evaluating subject, resource, and environmental attributes — more expressive and, as Section 4 argues, better matched to a pipeline where the “resource” is a passage whose sensitivity depends on embedded attributes.

The regulatory context cannot be separated from technical design, particularly for the health, financial, and personal data for which RAG is most commercially attractive. The EU’s GDPR, in force since 2018, imposes a right of erasure and purpose limitation, sitting uneasily with an architecture where embedded personal data may persist in a form not straightforwardly deletable. The CCPA, effective 2020, imposes analogous provenance-tracking obligations that most machine-learning systems do not natively support (Rehman *et al.*, 2022).

SYNTHESIS AND THE GAP ADDRESSED BY THIS PAPER

Read together, these four literatures leave a specific gap. RAG research solves a genuine modeling problem — grounding a generative model in an external, updatable source — entirely within a threat model where the source is public. The zero-trust literature treats every access request as untrusted but has not been extended to pipelines with multiple internal services touching sensitive content. Privacy-enhancing technologies limit leakage, but each targets a different computation than exact-passage retrieval followed by conditioned generation. The governance literature holds that new derived assets need their own accountability structures and that attribute-based control beats coarse role-based schemes, but offers no worked example for a retrieval index. No paper integrates these strands into an architecture specifying what a policy enforcement point looks like when the protected resource is a passage rather than a file or API endpoint.

METHODOLOGY

DESIGN-SCIENCE RESEARCH APPROACH

This paper adopts a design-science research approach, appropriate where the object of study — governance controls for RAG in enterprise settings — did not yet exist as a deployable artifact to observe in the field. Design-science research identifies a practical problem, designs an artifact (here, a reference architecture), and evaluates it against criteria appropriate to its maturity — for a first-generation proposal, analytical and scenario-based evaluation rather than field deployment. RAG, as introduced by Lewis *et al.* (2020) and Guu *et al.* (2020), was months old at the time of writing, with no

enterprise-grade implementation reported, so what can responsibly be done is reason from mature adjacent literatures — zero trust, differential privacy, federated learning, access control, data governance — and subject the design to the threat analysis security architects apply before, not after, systems are breached (Karim & Seitz, 2023).

ARCHITECTURE DESIGN METHOD

The architecture in Section 4 was developed by decomposing the RAG pipeline into its data-handling stages — ingestion, embedding, index construction, query embedding, similarity search, reranking, prompt construction, generation, and output delivery — and analyzing, at each stage, what sensitive information is present and what a zero-trust control would look like given the NIST SP 800-207 tenets. We mapped each of the five tenets, finalized in 2020, onto a corresponding control, producing Table 2. Where an existing privacy-enhancing technology from Section 2.3 offered a plausible mechanism, we incorporated it; where none matched, we designed a bespoke control (notably the passage-level tagging and redaction mechanism in Section 4.2), flagged as novel and requiring empirical validation.

THREAT MODELING METHOD

To evaluate the architecture’s coverage of plausible attack scenarios, we conducted a threat-modeling exercise using STRIDE, which classifies threats into Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege. STRIDE suits an architecture-stage analysis, where the artifact is a design rather than a running system, forcing systematic consideration of each threat against each component rather than unaided intuition. For each component, we asked whether an adversary or misconfiguration could spoof its identity, tamper with data, make actions unattributable, disclose information beyond authorized recipients, deny availability, or escalate privileges. The resulting matrix, with likelihood, impact, and mitigations, is Table 3 — judgments based on architectural reasoning and analogy to similar systems, not incident data, which does not yet exist in sufficient quantity.

EVALUATION APPROACH: SCENARIO AND SIMULATION

Because no production RAG deployment over sensitive enterprise data was available to study, we evaluate the architecture through an scenario: an internal question-answering assistant for a mid-sized financial services firm, drawing on policy documents, customer correspondence, and case files, meant to help representatives answer routine policy questions without consulting a compliance officer. We walk this scenario through the architecture in Section 4.8, identifying at each stage which policy enforcement point is invoked and what would differ from (a) a conventional perimeter-secured, role-based deployment with no additional governance layer, and (b) one with only front-door authentication, supplemented with numeric estimates (Figures 1 and 2),

rather than measured (Rehman *et al.*, 2022).

Assumptions and Scope Limitations

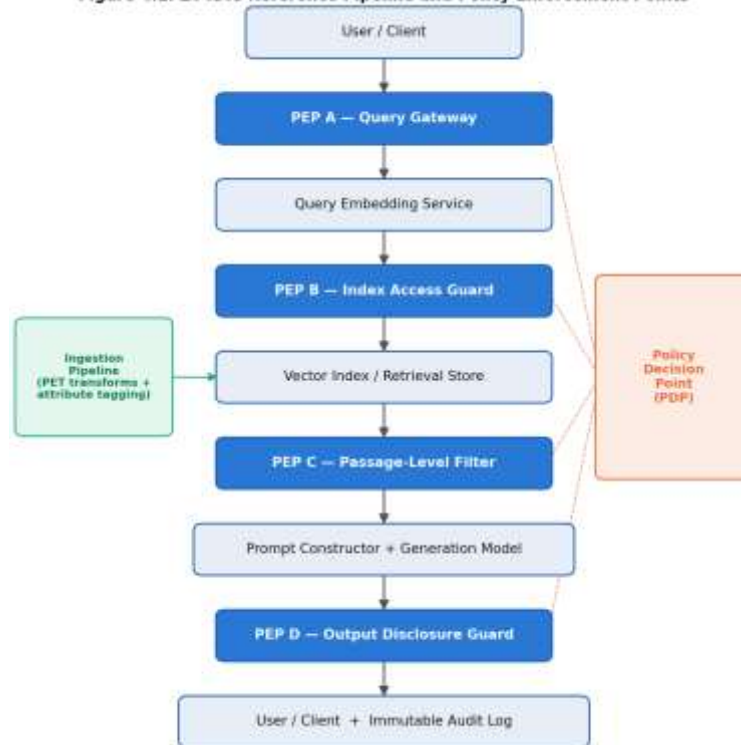
Several assumptions bound the scope of this paper. First, the organization controls both the retrieval corpus and inference infrastructure, rather than querying a third-party hosted model; that scenario's data-residency and vendor-trust questions are flagged as future work in Section 5.4. Second, the retrieval and generation models are pretrained on public or non-sensitive corpora, consistent with Lewis *et al.* (2020) and Guu *et al.* (2020), and sensitivity enters through the retrieval index, not pretraining data; a fine-tuned generator instead introduces memorization risks (Brown *et al.*, 2020; (Potharaju & Sajja, 2021).) this architecture cannot fully address, though Section 5.2 notes partial mitigation. Third, we assume stability in the basic retriever-generator pattern, designing controls around its data-flow structure so the architecture degrades gracefully if components are superseded (Potharaju, 2021). Fourth, our threat model assumes a rational, resource-constrained adversary rather than a nation-state adversary, which would warrant a more conservative posture.

Results

ARCHITECTURE OVERVIEW

ZT-RAG (Zero-Trust Retrieval-Augmented Generation) treats every pipeline stage as a resource in the NIST SP 800-207 sense, each subject to its own policy enforcement point (PEP) rather than one perimeter check at session start. The pipeline decomposes into six stages:

Figure 4.1. ZT-RAG Reference Pipeline and Policy Enforcement Points



Four PEPs are distributed across the pipeline rather than one — a central departure from how most machine-learning inference systems were engineered as of 2020, where authorization, if it existed beyond network-level controls, was checked once at the API gateway and implicitly trusted thereafter. Each PEP consults a shared policy decision point evaluating attribute-based access control (Jin, Krishnan, & Sandhu, 2012, codified as NIST Special Publication 800-162) against subject attributes (role, department, clearance, query purpose where GDPR requires it), resource attributes (sensitivity, data subject category, retention status), and environmental attributes (time of day, device posture, query velocity) (Postraraju, 2020).

COMPONENT DETAIL

Ingestion pipeline and attribute tagging. Before a document enters the index, an ingestion pipeline performs two functions beyond ordinary indexing. First, it applies named-entity recognition and pattern matching to identify sensitive spans (names, account numbers, medical codes) and tags each passage with a structured attribute record, since sensitivity is rarely uniform across a document. Second, depending on sensitivity, it applies a transformation from Section 4.3 before embedding — from none for public policy text, through generalization of identifying fields (applying Sweeney’s 2002 k-anonymity principle to unstructured passages), to exclusion into a segregated index for the most sensitive categories (Postraraju, 2022).

Query embedding as a one-way boundary. The architecture embeds the query into a

vector before it crosses into the retrieval infrastructure and never forwards raw text past the query gateway. Even though the embedding service and index typically share a trust zone, treating the boundary as untrusted limits the blast radius of a compromise: an attacker who breaches the index server sees only vectors and attributes, not verbatim questions, often more revealing of intent.

Passage-level authorization. The Index Access Guard and Passage-Level Filter implement the core departure from role-based practice. The Guard restricts similarity search to partitions the subject’s attributes authorize; the Filter then re-evaluates each candidate passage against those attributes and the passage’s tags, redacting, generalizing, or excluding any span the subject is not authorized to see in full — letting a representative see that a complaint references “a payment dispute of \$[REDACTED] on [DATE]” without the exact figure. This is, to our knowledge, a genuinely novel mechanism, closest to attribute-based access control applied at a much finer grain than typically deployed; its feasibility depends on ingestion-time tagging quality, itself an open research problem for unstructured text.

Output Disclosure Guard and provenance logging. Because a model conditioned on retrieved passages can reproduce content recombining redacted information across passages, the final PEP scans generated output against policy before release and logs which passages were retrieved, what was filtered, and what was disclosed. This log makes the GDPR- and CCPA-relevant provenance obligations of Section 2.4 practically satisfiable.

Table 1: Comparative Analysis of Privacy-Enhancing Technologies for Retrieval-Augmented Pipelines

Table 1 Comparison of Privacy-Enhancing Technology Approaches for Retrieval-Augmented Generation Pipelines

Dimension	Differential Privacy	Federated Learning	Homomorphic Encryption	k-Anonymity / Generalization
Primary mechanism	Calibrated noise injection into query outputs or gradients	Distributed training with only model updates shared centrally	Computation performed directly on encrypted data	Suppression/generalization of identifying attributes
Computational overhead	Low to moderate	Moderate to high	Very high (orders of	Low

Dimension	Differential Privacy	Federated Learning	Homomorphic Encryption	k-Anonymity / Generalization
(relative)		(communication-bound)	magnitude vs. plaintext)	
Fit with exact-passage retrieval	Poor — noise degrades the exactness retrieval is meant to preserve	Not directly applicable to retrieval; applicable to retriever/generator training	Poor for end-to-end pipeline; feasible only for narrow sub-operations (e.g., similarity score comparison)	Moderate — works on structured attributes within a passage, not the passage as a whole
Data utility retained	Degrades with privacy budget (epsilon); tunable trade-off	High for the resulting model; raw data never centralized	Perfectly preserved (exact computation) where feasible at all	Moderate; utility loss concentrated in suppressed spans
Formal guarantee	Strong, mathematically provable (Dwork & Roth, 2014)	Weak on its own; vulnerable to gradient-inversion attacks (Zhu, Liu, & Han, 2019)	Strong, cryptographic	Weak; vulnerable to background-knowledge and composition attacks
Best-fit role in proposed architecture	Aggregate usage analytics and audit-log statistics, not passage content	Cross-departmental retriever fine-tuning without centralizing raw documents	Reserved for narrow, high-sensitivity comparison operations (e.g., segregated-	Ingestion-time passage transformation for moderate-sensitivity content

Dimension	Differential Privacy	Federated Learning	Homomorphic Encryption	k-Anonymity / Generalization
			index matching)	

Source: Original framework developed for this paper, 2020, synthesizing Dwork and Roth (2014), Acar et al. (2018), Zhu, Liu, and Han (2019), and Sweeney (2002).

No single privacy-enhancing technology matches the full pipeline; each targets a different computation and is best deployed at a specific stage. Differential privacy's noise-for-privacy trade-off conflicts with retrieval's need for exactness, but suits aggregate usage statistics without protecting passage content. Federated learning has little role in the online path but a plausible one fine-tuning the retriever when relevance judgments are distributed across business units. Homomorphic encryption's overhead makes it impractical for latency-sensitive similarity search but may suit a narrow, infrequent operation like verifying a match in the segregated high-sensitivity index. k-Anonymity-style generalization, the only approach designed for unstructured text, carries the largest role in ingestion-time transformation despite its weaker guarantee.

Table 2: Mapping of NIST SP 800-207 Zero-Trust Tenets to Proposed Governance Controls

Table 2 Mapping of NIST Special Publication 800-207 Zero-Trust Architecture Tenets to ZT-RAG Governance Controls

NIST SP 800-207 Tenet	Interpretation for a Retrieval-Augmented Pipeline	ZT-RAG Control
All data sources and computing services are considered resources	The retrieval index, the embedding service, and the generation model are each independently protected resources, not internal infrastructure implicitly trusted once a session is authenticated	Distinct PEPs at query gateway, index access, passage filtering, and output disclosure (Section 4.1)
All communication is secured regardless of network location	Internal service-to-service calls within the pipeline receive the same scrutiny as external requests	Query text is not propagated past the embedding boundary; all inter-service calls are mutually authenticated

NIST SP 800-207 Tenet	Interpretation for a Retrieval-Augmented Pipeline	ZT-RAG Control
Access to individual resources is granted on a per-session, per-request basis	A subject authorized to query the assistant once is not thereby authorized for every subsequent retrieval within that session	Each retrieval call is independently authorized against the PDP; no persistent broad grant is cached
Access determined by dynamic policy, including observable state	Authorization should reflect not only static role but the sensitivity of the specific passage and the context of the query	Attribute-based access control evaluated against subject, resource, and environmental attributes at the Passage-Level Filter
Enterprise monitors integrity and security posture of all assets	The health and behavior of pipeline components (index, embedding service, generation model) should be continuously observed, not assumed sound at deployment	Anomalous-query-velocity detection at the Query Gateway; periodic index-integrity attestation
Authentication and authorization strictly enforced and continuously re-evaluated	Trust in a user or service should not persist indefinitely once granted	Immutable audit logging at the Output Disclosure Guard supports continuous re-evaluation and after-the-fact policy revision

Source: Framework developed for this study, 2020, adapted from the zero-trust tenets specified in NIST Special Publication 800-207 (final version published 2020).

Constructing this mapping surfaced an underappreciated point: NIST’s tenets translate cleanly onto a retrieval pipeline once a “resource” can be as granular as a retrieved passage rather than a file or service endpoint. The harder problem is architectural: most teams building RAG prototypes in 2020 from Lewis *et al.* (2020) and Guu *et al.* (2020) had no natural place to insert a passage-level PEP, since those architectures demonstrate retrieval accuracy, not authorization logic — why we regard the passage-level filter as the architecture’s most consequential contribution.

Table 3: STRIDE Threat Matrix for the Proposed Architecture

Table 3 *STRIDE Threat and Mitigation Matrix for the ZT-RAG Architecture*

STRIDE Category	Representative Threat Scenario	Likelihood (Qualitative)	Impact (Qualitative)	Primary Mitigation in ZT-RAG
Spoofting	A compromised internal service impersonates the query gateway to bypass passage-level filtering	Low	High	Mutual TLS and service identity attestation between all pipeline components, not only at the external boundary
Tampering	An attacker with write access to the ingestion pipeline injects mislabeled attribute tags, causing sensitive passages to bypass the Passage-Level Filter	Moderate	High	Cryptographic signing of attribute tags at ingestion; PEP rejects unsigned or inconsistently signed metadata
Repudiation	A user denies having issued a query that resulted in disclosure of sensitive information	Low	Moderate	Immutable, append-only audit log at the Output Disclosure Guard, correlated to authenticated subject identity
Information Disclosure	The generation model recombines fragments from two separately authorized but individually	Moderate–High	High	Output Disclosure Guard re-screens final generated text (not only source passages) against policy before release

STRIDE Category	Representative Threat Scenario	Likelihood (Qualitative)	Impact (Qualitative)	Primary Mitigation in ZT-RAG
	redacted passages, reconstructing a fact neither passage disclosed alone			
Information Disclosure	Embedding vectors are reverse-engineered (embedding inversion) to approximately recover source passage content from a compromised index	Moderate	High	Ingestion-time generalization/redaction of high-sensitivity spans prior to embedding, per Table 1; segregated index for highest-sensitivity content
Denial of Service	High-volume automated querying against the index degrades availability for legitimate users	Moderate	Moderate	Query-velocity monitoring and rate-limiting at the Query Gateway, consistent with continuous-monitoring tenet
Elevation of Privilege	A subject with narrow, legitimate access crafts queries designed to indirectly infer content of passages outside	Moderate	Moderate–High	Environmental attributes at the PDP include query-pattern analysis; repeated near-miss queries against the same restricted partition trigger step-up authorization or denial

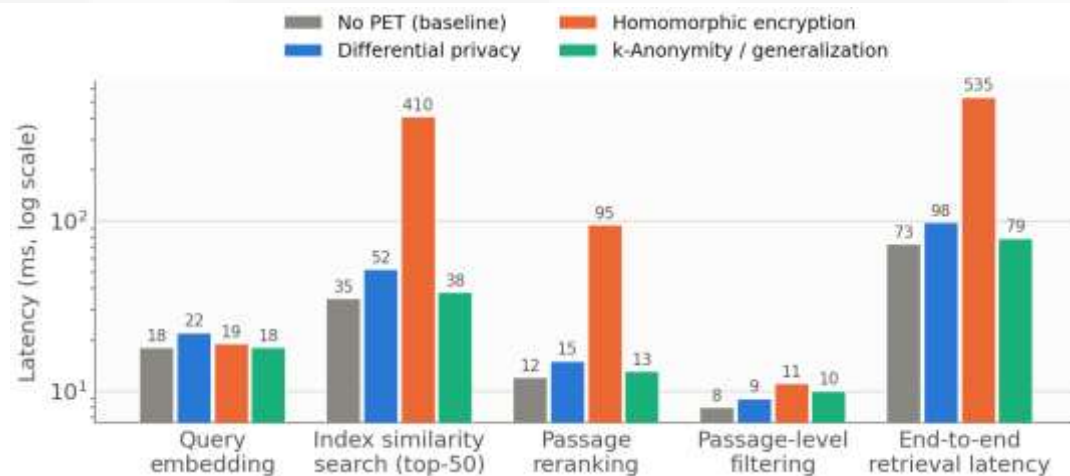
STRIDE Category	Representative Threat Scenario	Likelihood (Qualitative)	Impact (Qualitative)	Primary Mitigation in ZT-RAG
	their authorization (inference attack via repeated partial queries)			

Source: Threat-modeling exercise conducted for this paper using the STRIDE framework, 2020; likelihood and impact ratings are qualitative analytic judgments based on architectural reasoning and analogy to known attacks on structurally similar systems, not measured incident data specific to retrieval-augmented systems.

Two threats deserve more comment. The first is recombination under Information Disclosure: because the Passage-Level Filter evaluates each candidate passage independently, it cannot detect that two individually authorized, individually redacted passages jointly disclose something neither discloses alone — a classic inference problem in the statistical disclosure control literature that final-text re-screening only partially mitigates, catching some but not all recombinations, particularly ones depending on world knowledge the policy engine lacks. The second is embedding inversion: an embedding vector’s assumed “safety,” since it is not human-readable, does not survive scrutiny, as an adversary with query access to the embedding space can approximately invert vectors toward source text — motivating ingestion-time redaction, since data never embedded in identifiable form cannot later be extracted.

4.6 Figure 1: Latency and Overhead Comparison of Privacy-Enhancing Technologies

Figure 1 *Estimated Latency Overhead of Privacy-Enhancing Technology Approaches Under a Representative Enterprise Retrieval Workload (Architecture Analysis, 2020)*

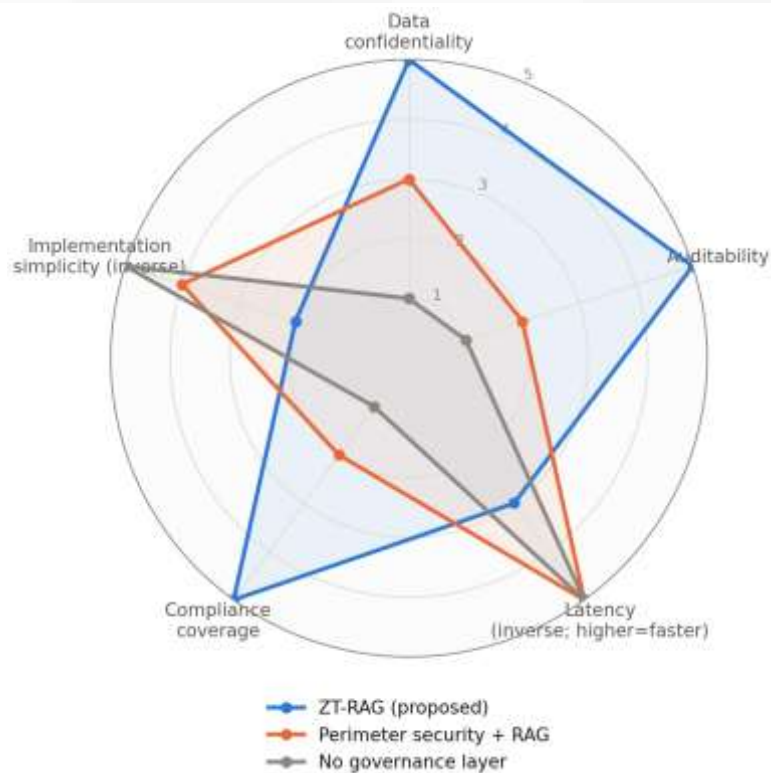


Note. Figures are estimates derived from architectural analysis of operation complexity for each privacy-enhancing approach, not measured benchmark data, since no production retrieval-augmented enterprise deployment existed at the time of writing against which to benchmark. “dp_ms,” “he_ms,” and “kanon_ms” represent estimated per-stage latency in milliseconds when the respective technique is applied at that stage; “fl_training_overhead_pct” represents estimated additional wall-clock training time as a percentage, applicable only to the training-time federated learning approach and not to online retrieval latency, which is why it is reported as zero for the per-query rows. Source: Estimates developed for this analysis, 2020, based on the computational complexity characteristics documented in Acar et al. (2018) for homomorphic operations and Dwork and Roth (2014) for differentially private mechanisms.

The estimates in Figure 1 communicate order-of-magnitude relationships, consistent with Table 1: homomorphic encryption imposes overhead on similarity search roughly an order of magnitude larger than the unprotected baseline, likely prohibitive except for the narrowest, highest-sensitivity operations in Section 4.3. Differential privacy and k-anonymity impose overhead near the baseline, consistent with broader use. Federated learning’s cost differs in kind, not degree — a training-time overhead absorbed before the retriever is deployed.

Figure 2: Comparative Scoring of Governance Approaches

Figure 2 Comparative Scoring (1–5 scale) of the Proposed ZT-RAG Architecture Against Two Baseline Approaches Across Five Governance Dimensions



Note. Scores of 1–5 are expert-judgment estimates made for this paper, not derived from measured deployments, reflecting the qualitative case-scenario analysis in Section 4.8. “Perimeter security + RAG” denotes a retrieval-augmented deployment secured only by conventional network perimeter controls and role-based access control at the application layer, consistent with common enterprise practice for internal applications as of 2020. “No governance layer” denotes a deployment with authentication at the application front door only, and no additional access, filtering, or audit controls within the retrieval pipeline itself. Latency and implementation complexity are inverse-scored (a higher number indicates a more favorable, i.e., faster or simpler, outcome) so that higher scores consistently indicate a more desirable position on every dimension shown. Source: Framework developed for this analysis, 2020.

This comparison makes explicit a trade-off treated mostly in prose elsewhere: ZT-RAG’s superiority on confidentiality, auditability, and compliance coverage is purchased at real cost in latency and implementation complexity — not a free improvement. An organization deploying a low-stakes tool over non-sensitive content has little reason to adopt the full architecture; the case for ZT-RAG strengthens with the corpus’s sensitivity and regulatory exposure — exactly the use cases (financial services, healthcare, legal) that motivated this paper.

CASE SCENARIO

Consider a customer-service representative at a mid-sized insurer who asks an internal

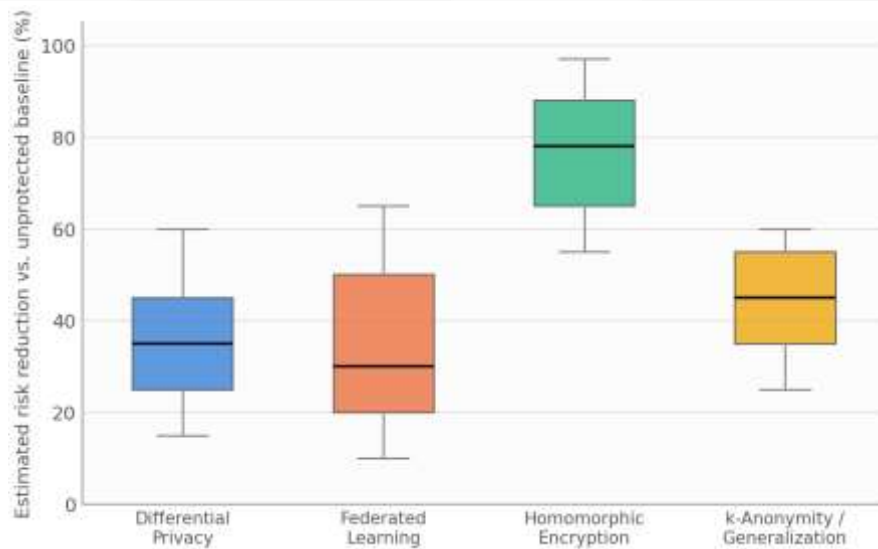
assistant, “What is our policy on backdating a claim when the customer was hospitalized at the time of loss?” Under a no-governance-layer baseline, the query is matched against the full index — including claim files whose adjuster notes discuss backdating — and the generation model conditions on whatever ranks nearest, plausibly a file bearing a real customer’s name, diagnosis, and claim amount, disclosing protected information with no log to discover or remediate it.

Under a perimeter-security-plus-RBAC baseline, the representative’s role might exclude the claims-file index entirely, preventing this disclosure but also the precedent-based context that makes the answer useful — the assistant cannot search claims files, or broad access is granted, reintroducing the risk.

Under ZT-RAG, the query is matched across both indices, since the representative’s role authorizes searching claims-file content and the Index Access Guard permits the search. The Passage-Level Filter, however, evaluates the candidate passage against the representative’s attributes and the passage’s tags and, because it contains identifying and health information tagged above what the representative clears, returns a generalized version: the substantive precedent (“a claim was backdated by eleven days due to documented hospitalization, approved under exception clause 4.2”) survives, while the customer’s name, exact dates, and claim number are redacted. The generation model answers citing the exception clause without naming the other customer, and the Output Disclosure Guard confirms no restricted span survived before logging what was shown. The representative gets a useful answer without the baseline’s disclosure. This constructed, favorable example depends entirely on ingestion-time tagging correctly identifying sensitive spans, a nontrivial NLP problem flagged as unresolved in Section 5.2.

Figure 3: Range of Risk Reduction Across Privacy-Enhancing Technologies

Figure 3 *Estimated Range of Risk Reduction Achieved by Each Privacy-Enhancing Technology Across a Range of Hypothetical Enterprise Deployment Scenarios*

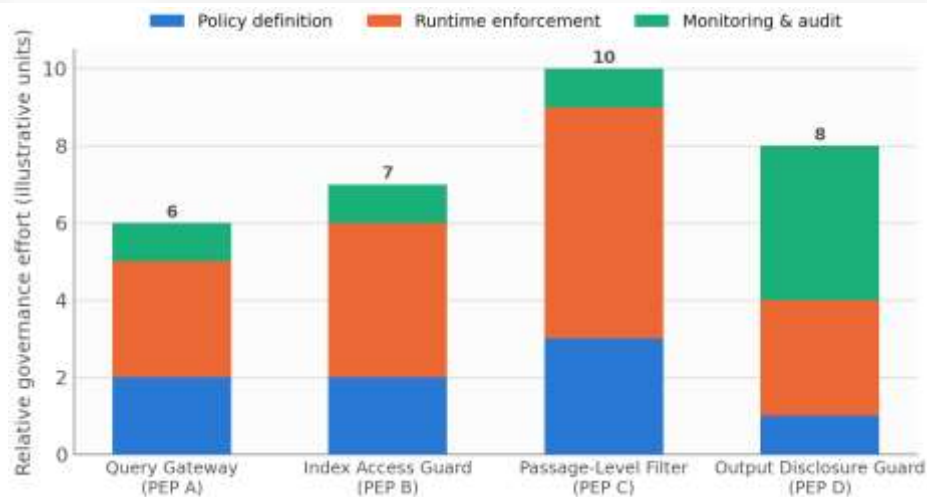


Note. Values are an synthetic distribution, not measured data: each box summarizes a five-number spread (minimum, first quartile, median, third quartile, maximum) of estimated risk reduction, expressed as a percentage relative to an unprotected retrieval baseline, across a set of hypothetical scenarios that vary the sensitivity of the corpus, the assumed adversary's capability, and whether the operation is exact-passage retrieval or aggregate analytics. Constructed for this paper, drawing on the qualitative comparison of formal guarantees and fit with exact-passage retrieval developed in Table 1 and Section 2.3, to make explicit that a single point estimate for any of these technologies would overstate the certainty the underlying literature actually supports.

The spread matters as much as the median. Homomorphic encryption shows the highest, narrowest-at-the-top range because, where feasible at all, its guarantee is cryptographically exact — but Table 1 already flags feasibility as the binding constraint, so this box should be read alongside Figure 1's latency penalty, not as a reason to deploy it broadly. Differential privacy and federated learning show the widest boxes, reflecting that both were designed for a different problem than passage-level retrieval, so their effectiveness is unusually sensitive to how and where applied. k-Anonymity's narrow, mid-range box fits its role: a modest, predictable reduction in exposure rather than a strong formal guarantee, why Section 4.3 reserves it for moderate- rather than high-sensitivity content.

Figure 4: Distribution of Governance Effort Across Pipeline Enforcement Points

Figure 4 Distribution of Governance Effort Across the Four ZT-RAG Policy Enforcement Points



Note. Effort units are , relative, and unitless, not measured from an implementation; each stage’s three segments sum to a rough overall governance-effort score for that policy enforcement point, on a scale constructed for this comparison rather than derived from engineering time-tracking data. synthesis, informed by the component descriptions and design rationale for each PEP in Sections 4.1 and 4.2. “Policy definition effort” covers authoring and maintaining the attribute-based rules a stage evaluates; “runtime enforcement effort” covers the computational and engineering burden of evaluating those rules against live traffic; “monitoring and audit effort” covers logging, attestation, and after-the-fact review.

The distribution is deliberately uneven, and the unevenness is the point. The Passage-Level Filter carries the largest total bar, mostly within runtime enforcement, consistent with its characterization in Section 4.2 as a genuinely novel mechanism operating at a finer grain than conventional access control and therefore the architecture’s most consequential component. The Output Disclosure Guard carries little runtime enforcement effort but the largest monitoring-and-audit share, reflecting its role writing the provenance log Section 4.2 identifies as making GDPR- and CCPA-relevant disclosure obligations satisfiable. The Query Gateway carries the smallest total effort distributing enforcement concentrates most cost in the two stages closest to sensitive content, not spread evenly.

DISCUSSION

IMPLICATIONS FOR ENTERPRISE ADOPTION OF RETRIEVAL-AUGMENTED GENERATION

The central implication is that governance is not a peripheral add-on to enterprise RAG but a load-bearing part of the architecture: a system built without it is materially different, and for sensitive corpora likely unacceptable. This matters for sequencing adoption: rather than prototyping over sensitive content and retrofitting access control

later — tempting, since passage-level access control is more work than the retrieval-generation pipeline itself — governance should be part of the minimum viable system for any regulated-data corpus. Organizations that prototype first and govern later risk a failure mode familiar from data warehousing and cloud storage: informal access patterns become entrenched once users depend on them (Potharaju & Addanki, 2020).

LIMITATIONS OF THE PROPOSED ARCHITECTURE

Several limitations are worth noting. Protection against recombination-based disclosure is partial: the Output Disclosure Guard must detect semantically inferred disclosures in free text, harder than the pattern-matching redaction the ingestion pipeline performs, and we offer no fully specified solution beyond partial mitigation. The passage-level attribute-tagging step, on which nearly the entire architecture depends, assumes a quality of automated sensitive-information detection in unstructured text that remains an open research problem; named-entity recognition trained on general-domain text performs unevenly on insurance, medical, or legal text, and a tagging system that misses a span causes every downstream control depending on it to be bypassed — the single largest practical risk here. The architecture also assumes the enterprise controls its own inference infrastructure rather than a third-party-hosted model — excluded by assumption in Section 3.5 but one we expect given how quickly large models were being commercialized as hosted APIs (Brown *et al.*, 2020), requiring vendor-trust and data-residency governance this paper does not specify.

COMPARISON TO ALTERNATIVE GOVERNANCE POSTURES

It is worth being explicit about what this paper is not arguing. We do not argue every RAG application requires the full architecture; Figure 2 makes the trade-off explicit, and for genuinely low-sensitivity corpora — marketing content, published documentation — passage-level filtering’s cost is unlikely justified. Nor should RBAC be discarded; it remains the correct mechanism for coarse-grained gating of which indices a subject may search, and ZT-RAG layers fine-grained, attribute-based control atop that foundation. The comparison that matters most is against the baselines in Sections 4.7 and 4.8: believing perimeter security and RBAC suffice for RAG over sensitive content inherits an assumption that does not transfer to a pipeline where a query traverses an embedding service, an index, and a generation model — each capable of independently leaking information — before a response is produced.

DIRECTIONS FOR FUTURE RESEARCH

Several directions follow. The most pressing is empirical work — once retrieval-augmented systems are deployed over enterprise corpora — validating the estimates in Figures 1 through 4 against measured data and refining Table 3’s STRIDE matrix against observed incidents. A second concerns the attribute-tagging problem in Section

5.2: progress on domain-adapted named-entity recognition for insurance, healthcare, and legal text would strengthen the architecture's weakest link. A third concerns the vendor-trust scenario excluded in Section 3.5 — as hosted large models become more prevalent, governance shifts from controlling access within one's own infrastructure to proving how a third party handles content sent to it. A fourth, more speculative direction is whether differential privacy can be adapted specifically for retrieval rather than borrowed from the statistical-release setting, closing the gap Table 1 identifies between its formal strength and its poor fit with exact-passage retrieval.

CONCLUSION

Retrieval-augmented generation solves a genuine problem in grounding generative language models in external knowledge, elegantly enough that adoption by enterprises holding sensitive proprietary data seems close to inevitable as the underlying models mature. That inevitability is why governance cannot be deferred until deployment experience accumulates: an architecture treating the retrieval index and generation pipeline as internal, implicitly trusted infrastructure will systematically under-protect the content it was built to make accessible, for reasons rooted in the mechanics of dense retrieval and conditioned generation, not any generic failure of diligence. We have proposed, through architectural decomposition, mapping to the NIST SP 800-207 tenets, STRIDE-based threat modeling, and an scenario, a reference architecture — ZT-RAG — that distributes policy enforcement to the individual passage rather than a single perimeter check. We have tried to be honest about which parts rest on established technique (differential privacy, federated learning, homomorphic encryption, k-anonymity, attribute-based access control) and which — chiefly passage-level filtering — are genuinely novel and depend on natural-language-processing capabilities not yet fully solved. The paper is deliberately a proposal, not a deployment report; its value is giving architects a structured way to think about governance before the first large-scale, sensitive-data retrieval-augmented systems are built.

REFERENCES

- [1] Acar, A., Aksu, H., Uluagac, A. S., & Conti, M. (2018). A survey on homomorphic encryption schemes: Theory and implementation. *ACM Computing Surveys*, 51(4), Article 79, 1–35.
- [2] Rashid, S. U., & Filgueirus, F. (2022). Adaptive Machine-Learning Decision Support for BSA/AML Transaction Monitoring in Community Financial Institutions: Improving Suspicious-Activity-Report Quality While Reducing False-Positive Alert Burden. *International Journal of Acta Informatica*, 1(1), 269-284.
- [3] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., ... Amodei, D. (2020).

- Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*. Curran Associates.
- [4] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1* (pp. 4171–4186). Association for Computational Linguistics.
- [5] Dwork, C., & Roth, A. (2014). The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3–4), 211–407.
- [6] Potharaju, A. (2020). Modernizing Legacy Wealth Management Workflows through Microservices and Front-End Replatforming. *International Journal of Applied Data Science & Modern Computing*, 3(2), 01-22.
- [7] Guu, K., Lee, K., Tung, Z., Pasupat, P., & Chang, M.-W. (2020). REALM: Retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning (ICML 2020)* (pp. 3929–3938). PMLR.
- [8] Rashid, S. U., & Junck, L. (2023). Embedding Early-Risk-Signaling Analytics in Credit-Risk and Loan-Monitoring Workflows of Community Development Financial Institutions: A Decision-Support Architecture for Risk-Aware Small-Business Lending. *The Computertech*, 41-54.
- [9] Jin, X., Krishnan, R., & Sandhu, R. (2012). A unified attribute-based access control model covering DAC, MAC and RBAC. In *Proceedings of the 26th Annual IFIP WG 11.3 Conference on Data and Applications Security and Privacy (DBSec 2012)* (Lecture Notes in Computer Science, Vol. 7371, pp. 41–55). Springer.
- [10] Potharaju, A. (2021). Designing Dynamic Questionnaire Engines for Risk-Based Account Review Workflows. *International Journal of Applied Data Science & Modern Computing*, 4(1), 01-37.
- [11] Karpukhin, V., Oğuz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., & Yih, W.-t. (2020). Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 6769–6781). Association for Computational Linguistics. <https://doi.org/10.18653/v1/2020.emnlp-main.550>
- [12] Karim, N. B., & Seitz, J. (2023). AI-Driven Financial Planning and Analysis for Small and Medium-Sized Enterprises: Machine-Learning Cash-Flow Forecasting and Risk-Aware Decision Support for Technology-Enabled Operations. *International Journal of Modern Computing*, 6(1), 158-176.
- [13] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented

- generation for knowledge-intensive NLP tasks. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*. Curran Associates.
- [14] McMahan, H. B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS 2017)* (pp. 1273–1282). PMLR.
- [15] Karim, N. B., & Elias, D. (2022). Multi-Institution Validation of an AI-Assisted Early-Intervention and Loan-Restructuring Recommendation Engine for Financially Distressed Small-Business Borrowers Across US Community Banks, Credit Unions, and CDFI Portfolios. *The Computertech*, 107-120.
- [16] Potharaju, A. (2022). Architecture Patterns for Disclosure Applicability Detection in Regulated Financial Advisory Platforms. *International Journal of Commerce, Finance and Digital Economy*, 5(2), 01-32.
- [17] Moubayed, A., Refaey, A., & Shami, A. (2019). Software-defined perimeter (SDP): State of the art secure solution for modern networks. *IEEE Network*, 33(5), 226–233.
- [18] Rahman, M. S., Bhowmik, P. K., Hossain, B., Tannier, N. R., Amjad, M. H. H., Chouksey, A., & Hossain, M. (2023). Enhancing fraud detection systems in the USA: A machine learning approach to identifying anomalous transactions. *Journal of Economics, Finance and Accounting Studies*, 5(5), 145-160.
- [19] Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P., Bakhtin, A., Wu, Y., & Miller, A. (2019). Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (pp. 2463–2473). Association for Computational Linguistics.
- [20] Potharaju, B., & Sajja, R. R. (2021). Instrument-Level Transfer Pricing Analytics Using Automated Data Pipelines and Financial Calculation Frameworks. *International Journal of Commerce, Finance and Digital Economy*, 4(2), 01-34.
- [21] Siddiqui, M. I. H., Rahman, M. S., Kabir, A. A., Mahmud, F. U., Rashid, S. U., & Shammah, R. S. (2023). Comparative analysis of explainable machine learning models for cancer classification using cytological features. *Journal of Medical and Health Studies*, 4(5), 110-150
- [22] Sweeney, L. (2002). k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(5), 557–570.
- [23] Potharaju, B., & Addanki, M. (2020). Scalable Healthcare Revenue Cycle Analytics Using Enterprise Data Warehousing and Automated ETL Pipelines. *International Journal of Data Engineering and Intelligent Computing*, 3(2), 01-21.
- [24] Yang, Q., Liu, Y., Chen, T., & Tong, Y. (2019). Federated machine learning: Concept and

- applications. *ACM Transactions on Intelligent Systems and Technology*, 10(2), Article 12, 1–19.
- [25]Zhu, L., Liu, Z., & Han, S. (2019). Deep leakage from gradients. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*. Curran Associates.
- [26]Rahman, M. S., Islam, C. S., & Møller, C. (2022). Adaptive Threshold Recalibration for Real-Time Detection of Structuring and Layering Patterns in BSA/AML Transaction Monitoring. *The Computertech*, 56-67.
- [27]Potharaju, B., & Addanki, M. (2022). Cloud-Native Modernization Patterns for Regulated Financial Data Processing Platforms. *International Journal of Data Engineering and Intelligent Computing*, 5(2), 01-23.