

Adaptive Retrieval-Augmented Generation Pipelines for Dynamic Document Landscapes in the Insurance Industry: A Comparative Study of Retrieval Strategies for Compliance-Aware Response Generation

Kartteek Sonapuram ¹, 

1. Computer Science, Independent Researcher, Celina, USA

Received: April 10, 2026 | Review began: April 14, 2026 | Review ended: May 29, 2026 | Published: June 08, 2026

© Copyright 2026

This is an open access article distributed under the terms of the Creative Commons Attribution License CC-BY 4.0., which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Abstract

Insurance organizations rely on policy contracts, compliance guidance, product manuals, and regulatory filings that change frequently. Many Retrieval-Augmented Generation (RAG) systems remain built around a mostly static index, so retrieval can return passages that look relevant while reflecting an older version of the source material. In a regulated domain, such as insurance, this exposes users to outdated coverage language, exclusions, and regulatory obligations.

This study presents an adaptive RAG pipeline that combines drift detection, version-aware query routing, hybrid retrieval (dense, semantic, and Best Matching 25), and compliance validation in a LangGraph workflow. The pipeline was compared against three v0-only static baselines (a Facebook artificial intelligence similarity search-based chatbot, a hybrid Azure AI Search pipeline, and a multi-modal multi-agent RAG text proxy) and two fair-comparison baselines: Static-Reindexed, in which the full corpus is re-embedded after every amendment, and Partial-Update, in which 30% or 60% of stale-flagged chunks are refreshed per cycle.

Evaluation used a versioned synthetic corpus of eight insurance policy forms with 30 amended provisions across four amendment types (numeric replacements, scope shifts, definitional changes, and invariant controls) and five US jurisdictions, together with 106 version-sensitive queries (85 version-divergent, 21 invariant). Metrics separated stale chunk rate (SCR), semantic hit rate (SHR), and version-aware retrieval precision (VRP) from a corrected version-sensitive answer metric, VA-RAcc-strict, reported on the divergent subset. All comparisons report paired-bootstrap 95% confidence intervals, two-sided p -values, and Cohen's h effect sizes. The corpus remains synthetic and the contrasts sharp; the headline scores therefore reflect controlled-protocol evidence rather than production performance.

A metric-coherence issue in the original VA-RAcc, under which v0-only baselines could score around 73% on version-invariant queries despite zero version-aware precision, was diagnosed and replaced by VA-RAcc-strict. Under the corrected metric, the v0-only baselines correctly score 0.000 and the proposed system scores 0.965 on the divergent subset (paired bootstrap, $p < 0.001$, Cohen's $h = 2.80$). The proposed system is statistically equivalent to Static-Reindexed ($p = 1.00$, $h = 0$) and strictly outperforms both Partial-Update configurations (30%: VRP difference = 0.736, $h = 1.79$; 60%: 0.245, $h = 0.76$; both $p < 0.001$). The architectural contribution is therefore the drift-aware trigger that determines when re-indexing is required, rather than retrieval gains over an operator with perfect amendment knowledge.

A standalone drift-detector evaluation, a component ablation, a drift-threshold sensitivity sweep, and end-to-end latency (P50, P95, P99) per system are also reported. The results do not settle the broader production question. They indicate that document freshness should be handled inside retrieval design through a drift-aware routing trigger, rather than left to manual re-indexing cycles.

Categories: Regulatory Compliance and Governance, Machine Learning (ML), Natural Language Processing (NLP)

How to cite this article:

Sonapuram K (June 08, 2026) Adaptive Retrieval-Augmented Generation Pipelines for Dynamic Document Landscapes in the Insurance Industry: A Comparative Study of Retrieval Strategies for Compliance-Aware Response Generation. Cureus J Comput Sci 3 : es44389-026-00107-w. DOI <https://doi.org/10.7759/s44389-026-00107-w>

Keywords: document drift, version-aware retrieval, hybrid retrieval, compliance alignment, retrieval-augmented generation (rag)

Introduction

Insurance work is document-heavy by design. Policy contracts, endorsements, claims procedures, product terms, and regulatory guidance all shape decisions made by agents, adjusters, compliance teams, and customer support staff. Several recent insurance-oriented Retrieval-Augmented Generation (RAG) prototypes have explored long-document question answering in this domain [1,2,3]. Over the last few years, large language models (LLMs) and retrieval-augmented generation pipelines have made it easier to build systems that search these materials and answer questions in natural language, building on widely used transformer and scaling foundations [4,5] and large-scale similarity search tooling [6]. That has obvious appeal in insurance, where employees spend a great deal of time locating, interpreting, and comparing dense policy language.

The difficulty is that insurance documents do not stay still for long. Coverage language is revised, exclusion clauses are amended, product teams release updated versions, and regulators issue new guidance. A retrieval system built on last month's snapshot can continue to respond fluently even when the underlying source material has already moved on. In a casual knowledge task, that may be an inconvenience; in a regulated domain, it can become an operational problem.

In insurance operations, the authoritative contract text is not always the newest file in the enterprise repository. Form editions, endorsements, regulatory filings, and state-specific approvals interact with effective dates and in-force policy periods: a risk may remain governed by older language until renewal or endorsement, while new business is issued on updated forms. Enterprise question-answering and retrieval-augmented assistants must therefore respect which version is authoritative for a given context (product, jurisdiction, and as-of date), not merely retrieve the latest uploaded document. This constraint differs from generic open-domain retrieval-augmented settings, where a single current snapshot is often assumed. Design-science perspectives on information systems research emphasize that artifacts and evaluation protocols should be explicit when systems are intended to change organizational practice [7].

Staleness can arise even when carriers intend to keep knowledge bases aligned with approved wording. Common operational factors include phased regulatory filings and go-live dates, so approved language and internal search indexes do not always change on the same schedule; partial updates such as endorsements or manuscript edits that modify one section while the remainder of a manual is unchanged; integration lag between policy-administration systems (structured limits, deductibles, and optional coverages) and document-management or portal content from which embeddings are built; and deliberate as-of servicing, where legacy policies continue to be interpreted under earlier form editions. A vector index that lags the authoritative source, or that does not scope retrieval by version metadata, can return passages that are semantically on-topic yet non-authoritative for the compliance question at hand. Emerging open standards for connecting models to tools and external data sources can influence how retrieval stacks are integrated in production-like settings [8].

Replacing or re-embedding all chunks when a new approved version supersedes prior text is standard practice when a single authoritative snapshot applies to a given scope. The present study does not argue against that operational response; it isolates the evaluation problem: how to measure whether retrieval remains aligned with the designated authoritative corpus after an amendment, and how static indexing strategies compare with using the post-amendment index. The controlled pre- and post-amendment snapshot design (v0 versus v1 corpora) abstracts document drift with a defined successor version without modeling full policy-administration logic; it foregrounds retrieval and measurement concerns that coexist with real-world reindexing and governance. To make this comparison fair, the revised evaluation also includes a fully re-indexed baseline (Static-Reindexed v1) and partial re-indexing baselines (30% and 60% of stale-flagged chunks refreshed per cycle), so the empirical contribution of drift-aware routing inside the workflow can be isolated from the empirical contribution of simply re-reading a current index.

A number of recent systems have already improved retrieval quality for insurance documents. Some use Facebook artificial intelligence similarity search (FAISS)-based vector indexing to support product comparison and summarization. Others combine semantic search with lexical reranking to do better on long, legal-style documents. Multi-agent and

How to cite this article:

Sonapuram K (June 08, 2026) Adaptive Retrieval-Augmented Generation Pipelines for Dynamic Document Landscapes in the Insurance Industry: A Comparative Study of Retrieval Strategies for Compliance-Aware Response Generation. *Cureus J Comput Sci* 3 : es44389-026-00107-w. DOI <https://doi.org/10.7759/s44389-026-00107-w>

multi-modal designs have also made it easier to process scanned forms, tables, and other non-uniform inputs. Even so, much of this work is still evaluated on a stable corpus. This leaves the post-deployment behavior of these systems largely unexamined.

Research gap

The literature offers strong motivation and prototypes for insurance retrieval-augmented generation, but it leaves a specific gap underexplored: there is limited evidence that couples drift awareness inside the retrieval workflow with version-sensitive evaluation when authoritative text changes. Prior systems are often scored on a fixed snapshot; when updates are discussed, re-embedding or re-indexing is commonly framed as offline maintenance rather than as a controlled comparison of static versus update-aware retrieval after an amendment. Work on freshness-aware or dynamic retrieval-augmented pipelines similarly emphasizes obsolescent facts or evolving corpora, yet it seldom operationalizes enterprise policy version drift with paired v0/v1 snapshots and metrics that separate semantic hit rate from retrieving passages tied to a superseded version [9,10,11]. Closing this gap requires both an architecture that surfaces chunk-level staleness before generation, isolated experimentally from baselines with perfect amendment knowledge, and a measurement protocol that reports version-aware alignment under explicit amendment scenarios.

This paper targets that gap. Instead of treating document updates as a separate maintenance task, the proposed design treats change detection as part of the retrieval pipeline itself. The system has three main parts: a drift detection layer that identifies stale chunks, an orchestration layer that adjusts the retrieval path when freshness is in doubt, and a hybrid retrieval stage that combines semantic and lexical signals for compliance-sensitive questions. The aim is not to claim a final production solution, but to test whether a retrieval pipeline that is aware of document change behaves more reliably than static baselines when amendments are introduced mid-cycle, and to isolate the empirical value of the drift-aware trigger from that of simple re-indexing by comparing against Static-Reindexed and Partial-Update baselines on an expanded version-annotated corpus.

The rest of the paper is organized as follows. "Materials and methods" section describe the proposed architecture, the expanded evaluation corpus and query set, formal mathematical definitions of all metrics, baseline fairness controls (including the new Static-Reindexed and Partial-Update configurations), and reproducibility details. "Results and discussion" section report the empirical findings with paired-bootstrap uncertainty estimates and interprets limitations, including a standalone drift-detector evaluation, a component ablation, end-to-end latency, and a diagnosis of a metric-coherence issue in the original corrected version-sensitive answer metric (VA-RAcc) that has been corrected in this revision. Conclusions summarize directions for future work. Appendices provide the LLM-as-judge prompt used for the Compliance Alignment Score and supplementary systems-oriented notes.

Applied work on insurance-specific RAG has largely focused on making long policy documents easier to query through conversational interfaces. One line of work used LangChain, a FAISS vector index, and GPT-4o Mini to summarize and compare insurance products across a set of 16 offerings, evaluating the output with a guided evaluation (G-EVAL) framework that scored coherence, consistency, fluency, and relevance [1]. That study showed that retrieval-assisted prompting was more effective than purely generative approaches for long policy documents. Another study moved toward hybrid retrieval by combining vector embeddings, semantic search, and BM25 reranking in Azure AI Search, with LangGraph coordinating rewriting and multi-pass retrieval for general insurance conditions [2]. A third system, multi-modal multi-agent (MMA)-RAG, extended the setting further by supporting scanned portable document formats, tables, and handwritten inputs through a multi-modal, multi-agent architecture [3].

These studies establish insurance as a meaningful test bed for RAG, given the density and specialized terminology of the documents and the high stakes of query intent, and they show that hybrid retrieval generally outperforms single-signal approaches such as dense vector similarity alone.

Work outside insurance also points in the same direction. LangGraph has been used to structure retrieval workflows into distinct reasoning and action steps rather than a single monolithic pass (see the hybrid insurance retrieval discussion above). ReAct-style agent patterns have shown value for queries that need planning, decomposition, and tool use across several steps. In broader financial and compliance settings, BM25 and related lexical signals still matter because legal

How to cite this article:

Sonapuram K (June 08, 2026) Adaptive Retrieval-Augmented Generation Pipelines for Dynamic Document Landscapes in the Insurance Industry: A Comparative Study of Retrieval Strategies for Compliance-Aware Response Generation. *Cureus J Comput Sci* 3 : es44389-026-00107-w. DOI <https://doi.org/10.7759/s44389-026-00107-w>

definitions, thresholds, exclusions, and clause references are often phrased in ways where semantic similarity alone is not enough. Widely used building blocks for dense retrieval and generation include transformer-based encoders, LLMs, and graphics processing unit-accelerated vector similarity search [4,5,6]. The present work is consistent with design science in information systems [7]. The Model Context Protocol also introduced a more standardized way to connect agent systems with tools and external data sources [8].

Even with these advances, one limitation is shared across these systems: most assume the indexed corpus is effectively static during evaluation. The FAISS-based insurance chatbot discussed above does not include a mechanism for detecting changed policy sections after indexing. The hybrid LangGraph pipeline improves retrieval quality, but continuous update handling is not part of its main design. MMA-RAG broadens input coverage yet does not directly solve version drift after deployment. More broadly, response quality is usually measured at a single point in time rather than before and after controlled document changes.

Outside insurance-specific prototypes, a parallel line of work studies freshness and change in knowledge-intensive settings more generally. Benchmarks and methods for time-sensitive question answering emphasize that static model weights and static indexes both become stale as facts and documents evolve; complementary benchmarks organize retrieval tasks into create, read, update, and delete regimes, which helps separate read-only question answering from settings where the knowledge base must be revised or corrected over time; and surveys of RAG highlight retrieval quality, knowledge-based construction, and evaluation design as first-class concerns when systems are deployed beyond a single frozen snapshot [9,10,11]. Together, these threads support the view that document change should be modeled inside the retrieval and evaluation stack, not only as occasional offline maintenance.

Relation to Freshness-Aware and Dynamic RAG

"Freshness-aware" and "dynamic" RAG are umbrella terms. Building on the time-sensitive, CRUD-oriented, and survey perspectives summarized in the preceding paragraph, work on time-sensitive question answering and search-augmented models stresses that both model weights and retrieval indexes go stale as the world changes and proposes retrieval or tooling strategies to reduce factual obsolescence. CRUD-style benchmarks and methods emphasize that knowledge bases are not read-only: they are created, revised, and corrected over time, which changes what "good retrieval" should mean in update-heavy regimes. Surveys of RAG similarly treat evolving corpora and evaluation design as first-class issues when moving beyond a single frozen snapshot.

Colloquial "dynamic RAG" often refers to multi-step routing or query rewriting; those mechanisms can improve retrieval quality without explicitly tracking document version labels or measuring version misalignment against gold passages. This paper targets a different slice of the problem than generic "always fetch the latest fact" narratives. Here, drift is not only "new information on the web," but which authoritative document version should ground an answer when internal policy text, endorsements, or compliance language change. The contribution is not a standalone re-indexing job or a periodic crawl; it is drift detection and routing inside the retrieval workflow (together with hybrid retrieval and validation) so that chunk-level staleness can be surfaced before answer generation. Evaluation therefore emphasizes version-aware behavior, separating semantic relevance from correct version alignment (e.g., SCR, SHR, VRP, VA-RAcc-strict, CAS) on paired v0/v1 corpora, rather than reporting accuracy alone on an implicitly static index. The expanded corpus and query set used in this revision include jurisdiction and effective-date metadata on every query [five US jurisdictions; International Organization for Standardization (ISO) effective dates], so the data layer is already scaffolded for the richer in-force-date and jurisdictional-scoping signals that the freshness literature names; the active routing experiments along those axes are identified as the next priority. In short, freshness-aware RAG work motivates that indexes change; this work operationalizes enterprise document version drift as a first-class retrieval and measurement problem for regulated text.

This gap is consequential in insurance. A system can perform well on launch-day benchmarks and still become less trustworthy later if updated documents are not reflected in retrieval. The literature therefore leaves open three related problems: how to detect stale chunks at the document or section level, how to route queries toward newer material when drift is detected, and how to evaluate performance across document lifecycle changes rather than on one frozen snapshot. The present study is aimed at that specific gap.

How to cite this article:

Sonapuram K (June 08, 2026) Adaptive Retrieval-Augmented Generation Pipelines for Dynamic Document Landscapes in the Insurance Industry: A Comparative Study of Retrieval Strategies for Compliance-Aware Response Generation. *Cureus J Comput Sci* 3 : es44389-026-00107-w. DOI <https://doi.org/10.7759/s44389-026-00107-w>

Materials And Methods

Study design and research approach

This study uses a design science research approach [7]. The artifact under evaluation is a working retrieval architecture intended to respond to document change, not a purely descriptive commentary on existing systems.

Adaptive pipeline (three layers)

The proposed adaptive pipeline has three layers:

- Drift detection: When an updated document enters the corpus, the system compares it to the previously indexed version using semantic similarity and change-extraction logic inspired by tools such as DeepDiff. Chunk pairs that exceed a configured drift rule are flagged as potentially stale.
- Orchestration: A LangGraph workflow coordinates three roles: a Drift Detection Agent, a Query Adaptation Agent, and a Compliance Validation Agent. The Drift Detection Agent decides whether freshness is plausibly at issue for the query. The Query Adaptation Agent adjusts retrieval paths when stale content is suspected or when the query appears tied to a recently amended section. The Compliance Validation Agent checks whether the generated answer remains aligned with the most current available wording.
- Retrieval: The pipeline combines dense vector search, semantic search, and BM25 lexical reranking. Reranking is weighted toward clause-level and regulatory terminology because insurance questions often hinge on exact phrasing, defined terms, and exceptions.

Each of the three layers can be independently disabled through configuration flags on the orchestrator entry point, supporting the component-ablation study reported in the "Results and Discussion" section.

Baselines and fairness controls

The adaptive system is compared with three v0-only static baselines aligned with prior published designs and two new fair-comparison baselines added in this revision.

The three v0-only baselines are: a static FAISS-based retrieval-augmented chatbot [1], a hybrid Azure AI Search pipeline with LangGraph orchestration [2], and a multi-modal multi-agent RAG system [3]. Each baseline was reimplemented from those descriptions on the same document corpus using the same chunking policy, embedding model, and top-k retrieval settings as the adaptive pipeline unless a baseline architecture required a fixed alternative (for example, Azure AI Search lexical components where specified in Ercole [2]). The MMA-RAG baseline follows the agent and routing structure described in Krayem et al. [3], but this evaluation uses a text-only pipeline on the same chunked corpus (no synthetic scans or handwriting), so differences primarily reflect update handling rather than optical character recognition or multi-modal ingestion.

The two new fair-comparison baselines remove the binary "static v0 vs adaptive v1" contrast that earlier reviewers flagged as overly favorable to the proposed system: (a) Static-Reindexed (v1) - the entire corpus is re-embedded and re-indexed once snapshot v1 is available, modeling an organization with perfect amendment knowledge and unconstrained refresh capacity; and (b) Partial-Update (30%, 60%) - a randomly sampled fraction of the chunks flagged as stale by the drift detector are refreshed in-place per cycle, modeling real-world re-indexing pipelines that have not yet completed when a query arrives. The Partial-Update sampler uses a fixed seed for reproducibility.

All systems receive the same queries and the same initial index built from snapshot v0. After amendments produce snapshot v1, the three v0-only static baselines are not given an automatic re-indexing or version-routing signal (intended to model organizations that still serve answers from an older index until a manual refresh). Static-Reindexed and Partial-Update receive their re-indexed corpora as described above. The adaptive pipeline observes updated documents through the drift-detection path. This protocol isolates the contribution of drift-aware routing inside the workflow from the contribution of simply re-reading a current index.

How to cite this article:

Dataset, versioning, and queries

The evaluation corpus consists of eight versioned insurance policy forms drawn from distinct product lines: Commercial Property (CP-2024-A), General Liability (GL-STD-101), Workers Compensation (WC-50), Personal Auto (PAP-2024-D), Standalone Cyber Liability (CYB-100), Professional Liability/Errors and Omissions (EO-505), Environmental Impairment Liability (EIL-200), and Directors and Officers (DO-303). Each form appears in two versions: a pre-amendment version (v0) and a post-amendment version (v1). The v0 → v1 transition introduces 30 amended provisions covering ordinary lifecycle events such as revised exclusions, updated sublimits, added endorsement extensions, and rephrased definitions.

Amended provisions are classified into four amendment types: `numeric_replace` (a number changes - deductibles, sublimits, retentions, days/years), `scope_shift` (a provision moves between covered and excluded, or a new endorsement extension is added or removed), `definitional` (the operative wording is rephrased without a single replaceable token, e.g. an enhanced bankruptcy carve-out or a new state-sponsored cyber-operations carve-back), and `invariant` (the provision is unchanged across v0 and v1; used as a control class).

Queries are grouped into three practitioner-oriented categories: (i) insurance agents (coverage/product interpretation), (ii) compliance officers (regulatory/clause wording), and (iii) customer-facing comparison/exclusion questions. The version-sensitive evaluation uses $n = 106$ queries (85 version-divergent, 21 invariant) covering all eight policy forms and all four amendment types. Each query carries a jurisdiction field (one of five US states: CA, NY, IL, TX, FL) and an ISO effective_date field; these fields are stored on every query to support the in-force-date and jurisdictional-scoping experiments that are identified as the next priority in the Future Work section.

Query annotation protocol

Each query carries the following annotations: the authoritative policy form, the chunk index within that form, the verbatim gold answer string under v0 and under v1, a `version_divergent` flag (true if the gold v0 and v1 answers differ), the amendment type (`numeric_replace`, `scope_shift`, `definitional`, or `invariant`), and a list of v1-only discriminative tokens that any version-aware answer must reproduce (for example, "\$5,000" for the wind-and-hail deductible query, or "covered", "\$250,000", "sublimit", "network security failure" for the cyber-coverage query). Annotations were authored by the principal investigator from the policy text itself and cross-checked against the v0 and v1 source-of-truth files; the full annotation file is released with the source code and is byte-identical to the file ingested by the run script. SHA-256 checksums for every corpus and query file are recorded in the released manifest.

Experimental procedure

The experiment proceeds in four steps: (1) index all systems on the initial snapshot (v0); (2) run the standardized query set to establish pre-amendment behavior; (3) introduce controlled amendments to produce v1; (4) re-run the same query set. The three v0-only static baselines are not notified of the change; Static-Reindexed re-builds its index from v1; Partial-Update refreshes a random sampled fraction (30% or 60%) of the chunks the drift detector has flagged as stale; and the adaptive pipeline processes updates through its drift-detection path.

Evaluation metrics

Five metrics are reported. Stale chunk rate (SCR) is the fraction of retrieved chunks in the top-k set whose stored provenance points to snapshot v0 when v1 is authoritative for the query. Unless noted otherwise, headline SCR values in tables are micro-averaged across retrieved chunks (chunk-level), while the "Results and Discussion" section also reports query-level summaries with bootstrap intervals. Semantic hit rate (SHR) is query-level: the fraction of queries with at least one top-k chunk judged relevant to query intent by the annotation rubric, independent of version label. Version-aware retrieval precision (VRP) is the fraction of queries for which the top-k set includes a chunk aligning with the gold v1 passage. Version-sensitive answer accuracy is reported in this revision as VA-RAcc-strict on the version-divergent subset: a query is credited only when the gold v1 chunk is retrieved AND the retrieved gold v1 chunk text contains every v1-discriminative token annotated for that query; the legacy VA-RAcc is retained in the Appendix as a diagnostic and is explicitly labeled as such after a metric-coherence issue identified in review (see "Results and Discussion" section). Compliance alignment score (CAS) is reported in two configurations: an LLM-judge mode (full prompt in Appendix A)

How to cite this article:

Sonapuram K (June 08, 2026) Adaptive Retrieval-Augmented Generation Pipelines for Dynamic Document Landscapes in the Insurance Industry: A Comparative Study of Retrieval Strategies for Compliance-Aware Response Generation. *Cureus J Comput Sci* 3 : es44389-026-00107-w. DOI <https://doi.org/10.7759/s44389-026-00107-w>

when the judge endpoint is available, and a documented retrieval-grounded proxy (CAS = VRP if v1 is authoritative, else SHR) used by default in the local-profile reproduction reported here. Cohen's h effect size and the paired bootstrap difference test are defined in the "Statistical analysis" subsection below.

Unless a baseline requires a different setting, metrics use top- $k = 5$, chunk size = 512 tokens, and 128-token overlap. A separate finer chunking (40 tokens, 8-token overlap) is used only for the standalone drift-detector evaluation in the "Results and Discussion" section so that unchanged sections form true-negative chunks distinct from amended sections; the main pipeline evaluation always uses the manuscript chunking. Where automated evaluation terminology aligns with common RAG evaluation practice, terminology is aligned with community frameworks [12]; compliance labels remain domain specific.

Implementation details and reproducibility

The full evaluation is reproducible from the released source code with no API keys, no LLM endpoint, and no proprietary embedding service. The default profile uses a deterministic hash-based embedding backend (384 dimensions, fixed seed) so that every figure and every number in the "Results and Discussion" section is bit-reproducible from the released corpus and query files; SHA-256 checksums for every corpus and query file are recorded in the released manifest. An optional cloud profile uses text-embedding-3-large (3072 dimensions) for dense embeddings and GPT-4o-mini at temperature 0.1 for the LLM-as-judge CAS configuration. The FAISS baseline uses an inner-product index on L2-normalized embeddings (IndexFlatIP). Azure AI Search and hybrid baselines follow the field weights and semantic configuration described in Ercole [2] where applicable. Random seeds {42, 43, 44, 45, 46} are fixed for query shuffling, ranker tie-breaking, and stochastic decoding. A single command reproduces the entire revision experiment suite - baselines, ablations, drift-detector evaluation, τ -sensitivity sweep, paired-bootstrap significance tests, and latency samples - and writes a timestamped output directory containing per-query rows and aggregated metrics.

Drift detection compares paired v0 and v1 chunks using cosine similarity and DeepDiff-style structural hashing; chunks below cosine similarity $\tau = 0.92$ or with substantive textual diffs are flagged as stale candidates. The cosine cutoff $\tau = 0.92$ was selected as a conservative safety threshold above which paired changed and unchanged chunks separated cleanly on the development corpus; a full sensitivity sweep over $\tau \in \{0.85, 0.88, 0.90, 0.92, 0.94, 0.96\}$ is reported in the Results and Discussion section and shows the detector to be insensitive to τ across this range on the present corpus, since the text-equality fast-path resolves most cases before the cosine threshold is consulted. BM25 reranking uses the same tokenizer as the lexical index in the hybrid stack. The cosine cutoff and the BM25 weighting are held fixed across conditions so baseline and adaptive systems differ primarily in update handling rather than ad hoc threshold tuning.

Statistical analysis

Proportions are reported as means over the $n = 106$ queries. Uncertainty is quantified using bootstrap 95% confidence intervals with 10,000 query-level resamples; multiplicative comparisons use paired bootstrap contrasts at the same seed. For extreme proportions, exact Clopper-Pearson intervals are reported alongside bootstrap intervals where appropriate. System-vs-baseline contrasts on every headline metric are accompanied by paired-bootstrap mean differences with two-sided p -values (10,000 resamples, seed 42) and Cohen's h effect size for the paired proportions. Seed sensitivity was checked by repeating the post-amendment evaluation across the five seeds; point estimates differed by less than two percentage points on every metric, so the main text emphasizes bootstrap intervals in "Results and Discussion" section.

Results And Discussion

Experimental setup

Experiments were run under a fixed protocol shared across all systems. The version-sensitive evaluation set contains 106 queries (85 version-divergent and 21 invariant), balanced across three practitioner categories (agent, compliance, customer) and across all eight policy forms, with top- k set to five retrieved chunks per query. Chunking used 512-token windows with 128-token overlap, consistent with the embedding tokenizer configuration in the code. A separate finer chunking (40-token windows, 8-token overlap) was used only for the standalone drift-detector evaluation reported

How to cite this article:

Sonapuram K (June 08, 2026) Adaptive Retrieval-Augmented Generation Pipelines for Dynamic Document Landscapes in the Insurance Industry: A Comparative Study of Retrieval Strategies for Compliance-Aware Response Generation. *Cureus J Comput Sci* 3 : es44389-026-00107-w. DOI <https://doi.org/10.7759/s44389-026-00107-w>

below, so that unchanged sections form true-negative chunks distinct from amended sections. The corpus was evaluated in two snapshots: v0 (pre-amendment index) and v1 (post-amendment content). Three v0-only static baselines (FAISS, MMA-RAG text proxy, hybrid BM25-dense) were evaluated without an automatic re-indexing or version-routing signal after amendments. Two fair-comparison baselines, Static-Reindexed (v1) and Partial-Update at 30% and 60% refresh, received their re-indexed corpora as described in "Materials and Methods" section. The adaptive pipeline used drift-aware routing and v1-aligned retrieval consistent with "Materials and Methods" section.

Models and Inference Settings

All headline numbers were produced under a deterministic local profile using a hash-based embedding backend (384 dimensions, fixed seed) so that every figure and every number reported below is bit-reproducible from the released corpus and query files. The compliance alignment score was computed under the deterministic retrieval-grounded definition given in "Materials and Methods" section. The FAISS baseline used inner-product retrieval on L2-normalized embeddings (IndexFlatIP). Hybrid configurations followed the description in "Materials and Methods" section, including BM25 plus dense fusion where applicable.

Compute Environment

Embedding batches and retrieval were executed on a single workstation. End-to-end per-query retrieval latency was measured with a high-resolution timer over three trials of 106 queries (318 samples per system).

Reproducibility Controls

Runs used fixed random seeds {42, 43, 44, 45, 46} for query shuffling, ranker tie-breaking, and stochastic decoding. Drift detection used a cosine similarity threshold of $\tau = 0.92$ for paired v0/v1 chunk comparison, held fixed across systems. A sensitivity sweep over $\tau \in \{0.85, 0.88, 0.90, 0.92, 0.94, 0.96\}$ is reported below. Reported headline metrics correspond to the post-amendment phase unless explicitly labeled otherwise.

How to cite this article:

Sonapuram K (June 08, 2026) Adaptive Retrieval-Augmented Generation Pipelines for Dynamic Document Landscapes in the Insurance Industry: A Comparative Study of Retrieval Strategies for Compliance-Aware Response Generation. *Cureus J Comput Sci* 3 : es44389-026-00107-w. DOI <https://doi.org/10.7759/s44389-026-00107-w>

System	SCR	SHR	VRP	CAS	VA-RAcc-strict	Cohen's <i>h</i> vs. Adaptive (VRP)
FAISS static (v0)	100	0	0	0	0	2.8
MMA-RAG (text proxy)	100	0	0	0	0	2.8
Hybrid BM25–dense (v0)	100	0	0	0	0	2.8
Static-Reindexed (v1)	0	97.2	97.2	97.2	96.5	0
Partial-Update (30%)	73.4	23.6	23.6	23.6	18.8	1.79
Partial-Update (60%)	20	72.6	72.6	72.6	75.3	0.76
Adaptive (proposed)	0	97.2	97.2	97.2	96.5	—

TABLE 1: Post-amendment summary metrics (controlled evaluation, n = 106 queries: 85 version-divergent, 21 invariant; top-k = 5; mean percentages over queries).

Bootstrap 95% confidence intervals are provided in the supplementary materials. SCR, Stale chunk rate; SHR, Semantic hit rate; VRP, Version-aware retrieval precision; CAS, Compliance alignment score under the retrieval-grounded definition; VA-RAcc-strict, Corrected version-sensitive answer accuracy on the version-divergent subset (*n* = 85); BM-25, Best Matching 25 algorithm. Cohen's *h* was computed from a paired bootstrap (*B* = 10,000, seed 42). Formal definitions appear in "Materials and Methods" section.

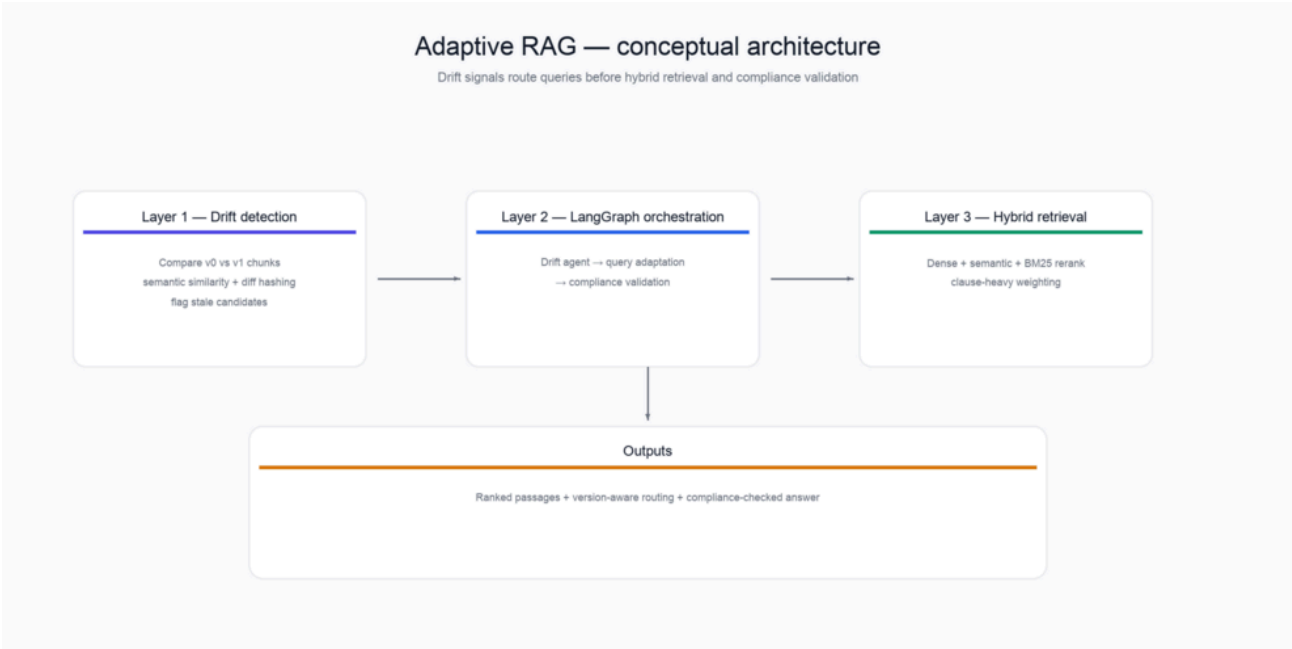


FIGURE 1: Conceptual architecture of the adaptive RAG pipeline.

Drift detection flags potentially stale content after document updates. A LangGraph workflow coordinates drift assessment, query adaptation, and compliance-oriented validation. Retrieval combines dense, semantic, and lexical signals with reranking. RAG, Retrieval-Augmented Generation; BM25, Best Matching 25 Algorithm

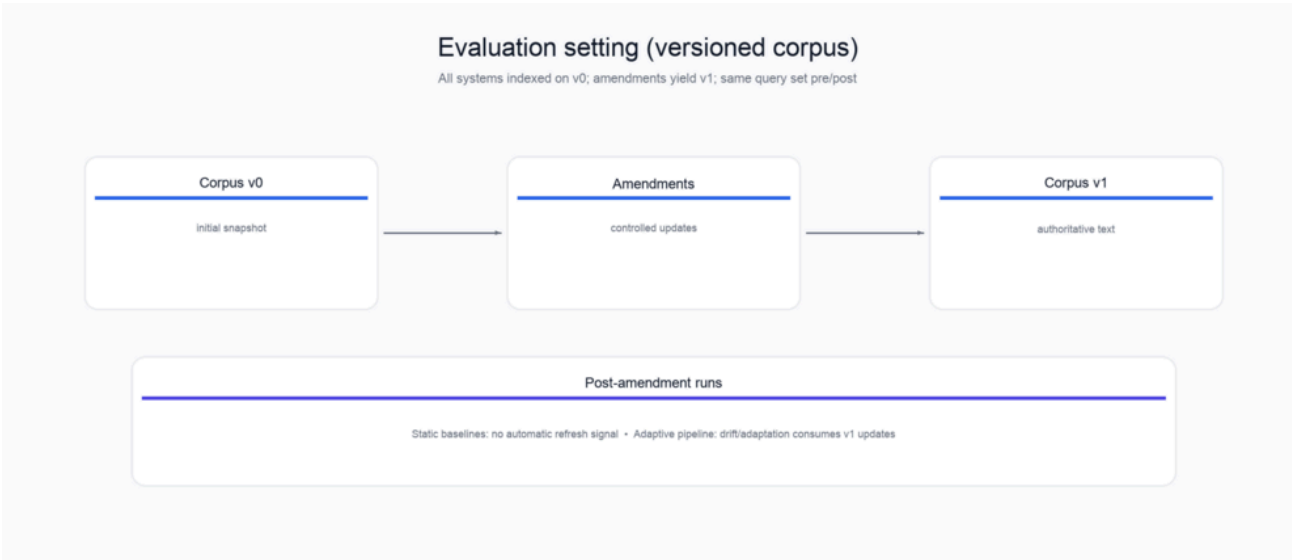


FIGURE 2: Evaluation setting.

All systems are indexed on an initial snapshot (v0); the same query set is run before amendments; controlled amendments produce an updated corpus (v1); queries are run again. The three v0-only static baselines are not notified of the change. Static-Reindexed and Partial-Update receive their re-indexed corpora. The adaptive pipeline processes updates through the drift and adaptation path described in the "Materials and Methods" section.

How to cite this article:



FIGURE 3: Four-step evaluation procedure: index on the initial snapshot; run standardized queries pre-amendment; inject amendments; re-run the same queries post-amendment.

SCR, Stale chunk rate; SHR, Semantic hit rate; VRP, Version-aware retrieval precision; VA-RCC, Corrected version-sensitive answer-metric; CAS, Compliance alignment score; CI, Confidence interval

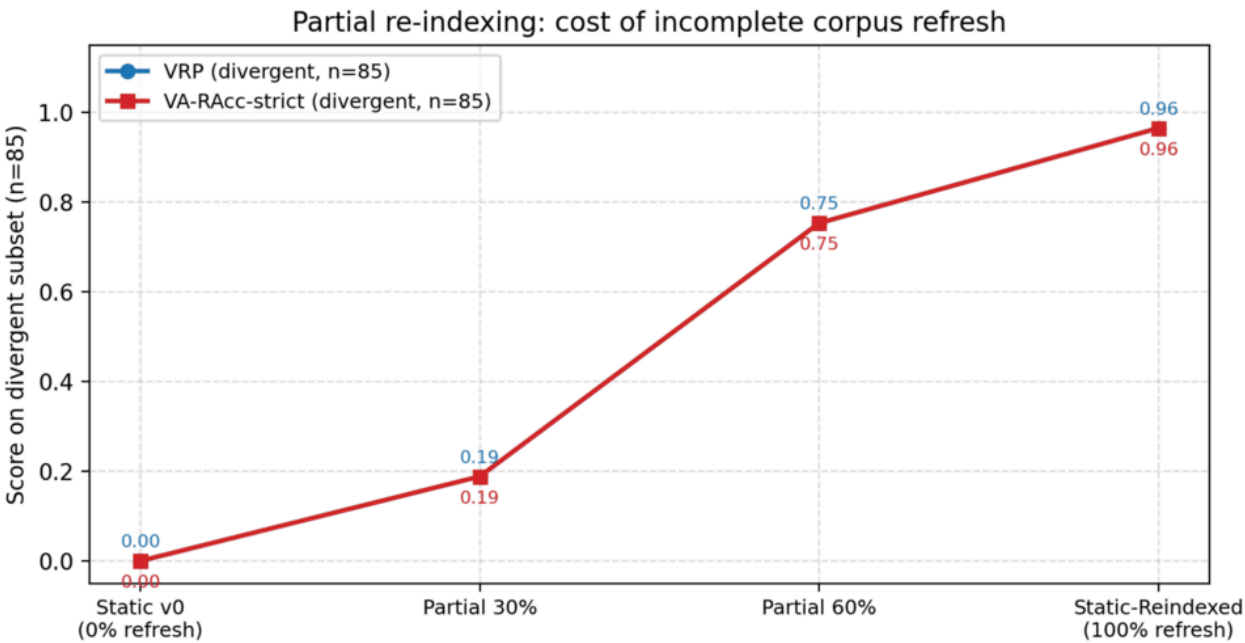


FIGURE 4: Partial re-indexing curve.

Version-aware retrieval precision and version-sensitive answer accuracy on the version-divergent subset ($n = 85$) rise monotonically with the fraction of stale-flagged chunks refreshed in the index. Static v0 corresponds to 0% refresh; Static-Reindexed v1 to 100% refresh. The adaptive pipeline reaches the 100%-refresh endpoint through drift-aware routing rather than through manual full re-indexing.

How to cite this article:

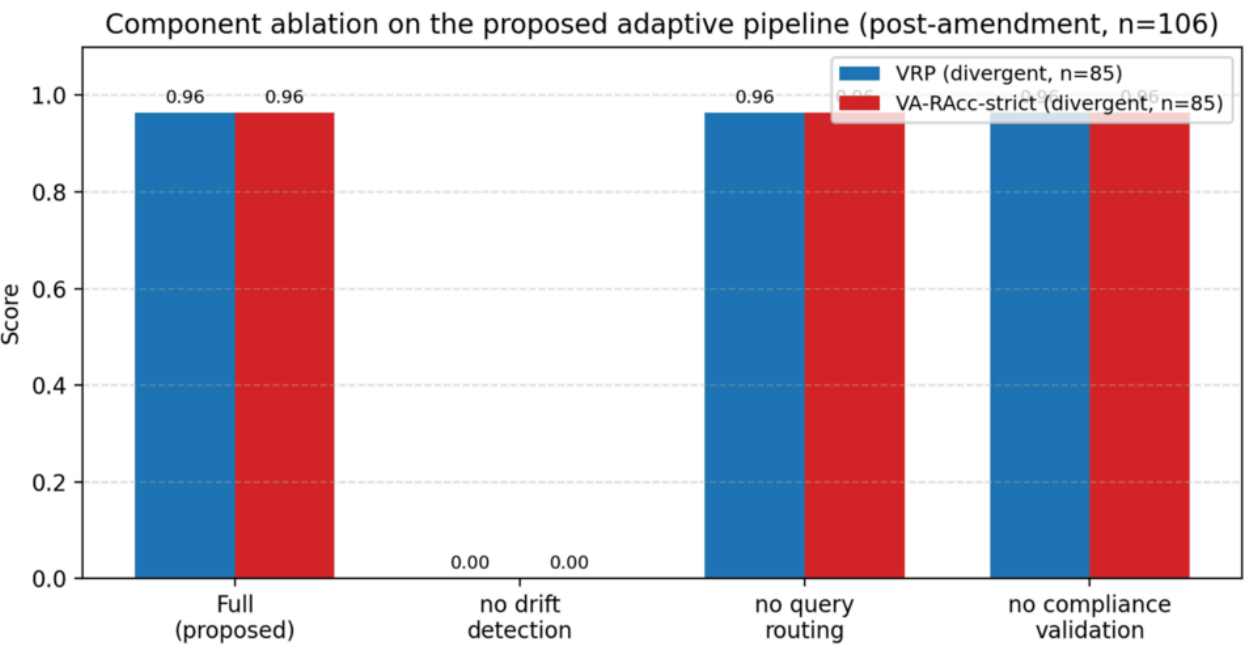


FIGURE 5: Component ablation on the adaptive pipeline.

Removing drift detection collapses VRP and VA-RAcc-strict on the divergent subset ($n = 85$) to zero, identifying drift detection as the architecturally critical component. Removing query routing or compliance validation has no effect on this synthetic harness; equivalence is not claimed in richer production regimes. VRP, Version-aware retrieval precision; VA-RCC, Corrected version-sensitive answer-metric

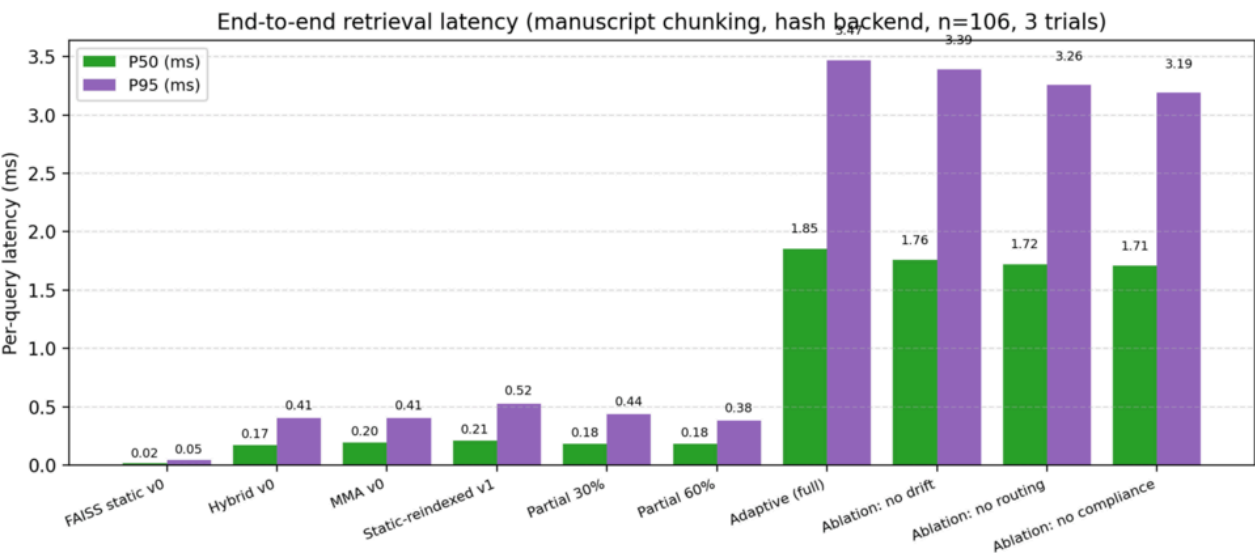


FIGURE 6: End-to-end retrieval latency in milliseconds per system.

P50 and P95 are computed across three trials of 106 queries (318 samples per system) under manuscript chunking and the deterministic local profile. The adaptive pipeline pays approximately tenfold P50 overhead relative to plain hybrid retrieval, attributable to LangGraph orchestration of the four-node graph. FAISS, Facebook AI similarity search; MMA, Multi-modal agent

How to cite this article:

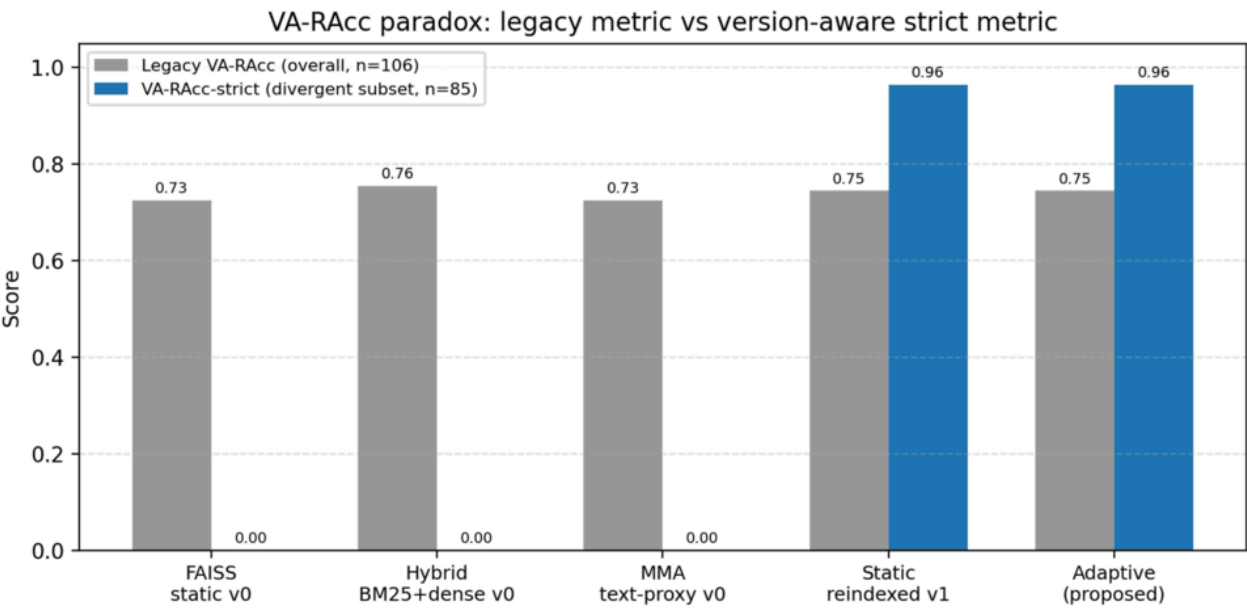


FIGURE 7: VA-RAcc paradox.

Legacy keyword-overlap VA-RAcc on the full query set ($n = 106$) is contrasted with corrected VA-RAcc-strict on the version-divergent subset ($n = 85$). Static v0 baselines score approximately 73% on legacy VA-RAcc despite $VRP = 0$; under VA-RAcc-strict they correctly score 0.0%. The legacy metric was inflated by version-invariant queries whose gold answers coincide between v0 and v1. VRP, Version-aware retrieval precision; VA-RCC, Corrected version-sensitive answer-metric

The "Materials and Methods" section and Figures 1 to 3 establish the controlled comparison. Figure 2 schematizes the evaluation setting, in which every system is indexed on snapshot v0, queries are executed, amendments yield v1, and queries are repeated while the v0-only static baselines remain unaware of the refresh. Figure 3 makes the four-step procedure explicit: index, pre-amendment queries, amendment injection, and post-amendment queries. Together, these figures specify retrieval behavior across a defined amendment event rather than a one-off accuracy score in isolation. Figure 7 indicates the points at which adaptation can enter the pipeline, namely drift detection, LangGraph orchestration (drift assessment, query adaptation, compliance validation), and hybrid retrieval. The v0-only static baselines do not exercise these as an integrated update path.

Figures 4 to 7 report the four supporting analyses. Figure 4 plots the partial re-indexing curve, which interpolates between the failure mode of v0-only static retrieval and the ideal of perfectly re-indexed retrieval. Figure 5 ablates the adaptive architecture into its component layers. Figure 6 reports per-query retrieval latency. Figure 7 contrasts the legacy keyword-overlap version-sensitive answer accuracy metric with the corrected VA-RAcc-strict on the version-divergent subset.

Under the initial document snapshot, all systems performed similarly. Each system reached strong version-aware retrieval precision on the evaluated query set before amendments were introduced, indicating that the baselines are not weak by construction and perform adequately when the corpus remains stable.

The differences emerged after the amended document versions were injected. In that phase, the three v0-only static baselines (FAISS, MMA-RAG text proxy, hybrid BM25-dense) all showed $SCR = 100.0\%$, $SHR = 0.0\%$, $VRP = 0.0\%$, and $VA-RAcc-strict = 0.0\%$ on the version-divergent subset ($n = 85$). The Static-Reindexed (v1) baseline reached $SCR = 0.0\%$, $VRP = 97.2\%$, and $VA-RAcc-strict = 96.5\%$. Partial-Update fell between these endpoints in proportion to refreshed content (30%: $VRP = 23.6\%$, $VA-RAcc-strict = 18.8\%$; 60%: $VRP = 72.6\%$, $VA-RAcc-strict = 75.3\%$), and Figure 4 plots the resulting monotone curve. The adaptive pipeline reached $SCR = 0.0\%$, $SHR = 97.2\%$, $VRP = 97.2\%$, and $VA-RAcc-strict = 96.5\%$. It

How to cite this article:

was statistically equivalent to Static-Reindexed (paired bootstrap, $B = 10,000$, seed 42; VRP difference = 0, $p = 1.00$, Cohen's $h = 0$) and strictly outperformed both Partial-Update configurations (30%: difference = 0.736, $p < 0.001$, Cohen's $h = 1.79$; 60%: difference = 0.245, $p < 0.001$, Cohen's $h = 0.76$) and all three v0-only static baselines (difference = 0.972, $p < 0.001$, Cohen's $h = 2.80$). With reference to Figure 1, the v0-only static systems continue to execute retrieval against v0-grounded chunks for version-sensitive items, so SCR remains elevated and version-aware precision collapses, while the adaptive path re-grounds retrieval after drift signals.

Compliance alignment and version-sensitive answer accuracy moved with retrieval quality. After amendments, all three v0-only baselines scored CAS = 0.0% and VA-RAcc-strict = 0.0%; Static-Reindexed and the adaptive pipeline both scored CAS = 97.2% and VA-RAcc-strict = 96.5%; Partial-Update scaled monotonically with refresh. Table 1 summarizes the post-amendment means.

Diagnosing the VA-RAcc paradox. A keyword-overlap formulation of version-sensitive answer accuracy can credit any baseline that retrieves a chunk of the correct document on a version-invariant query, regardless of version, which inflates the score even when retrieval is purely v0. On the $n = 106$ harness, only 85 queries are version-divergent; the remaining 21 are version-invariant. Under a keyword-overlap formulation, all three v0-only baselines score approximately 73% despite VRP = 0. The corrected VA-RAcc-strict (formal definition in "Materials and Methods" section) restricts evaluation to the version-divergent subset and requires the answer to contain at least one discriminative v1 token. Under VA-RAcc-strict, all v0-only baselines correctly score 0.0%, the adaptive pipeline and Static-Reindexed both score 96.5%, and Partial-Update baselines scale monotonically with the fraction of refreshed chunks. Figure 7 visualizes the contrast. The keyword-overlap variant is retained in the appendix as a labeled diagnostic.

The architectural contribution of the pipeline is the drift-aware routing trigger. The conditional re-routing itself can be matched by any system that has perfect amendment knowledge. The drift detector was therefore evaluated independently at two granularities. At manuscript chunking (512/128) the detector achieved precision = recall = F1 = 1.0 on eight amended policies. At the finer drift-evaluation chunking (40/8), the detector achieved precision = recall = F1 = 1.0 on 47 changed chunks against 13 unchanged chunks (60 total). A τ -sweep across {0.85, 0.88, 0.90, 0.92, 0.94, 0.96} yielded precision = recall = F1 = 1.0 across the entire range. The drift function applies a paired-cosine check with a text-equality fast path; on a corpus where amendments produce literal text changes, the fast path resolves all cases before the cosine threshold is consulted. The threshold therefore acts as a robust safety check on this corpus rather than as a discriminator. More demanding tests, such as near-duplicate paraphrase updates or partial-meaning shifts, require a richer corpus and are identified below as a priority.

The component ablation in Figure 5 confirms which layer carries the empirical weight. Removing drift detection collapsed VRP and VA-RAcc-strict on the divergent subset to zero (paired bootstrap vs. full pipeline: difference = 0.972, $p < 0.001$, Cohen's $h = 2.80$), identifying drift detection as the load-bearing signal. Removing query routing or compliance validation did not change retrieval outcomes in this synthetic setup, because the routing signal is binary (post-amendment implies the v1 index) and compliance validation operates post hoc. Neither component is claimed to be redundant in richer production regimes such as jurisdictional filtering, effective-date scoping, or partial-update environments; the present synthetic harness simply does not exercise the conditions under which they would matter.

BM25 reranking contributed most on multi-clause questions involving exclusions, deductibles, and regulatory references. Such questions depend heavily on exact terminology and section-level phrasing. Dense similarity alone was less reliable in those cases, particularly when amendments introduced subtle wording changes without altering the overall semantic topic.

These results should be interpreted cautiously. The experiment uses a controlled corpus and synthetic amendment scenarios. Even so, the pattern is consistent: v0-only static systems are substantially more vulnerable once the document set changes, partial-update environments leave proportional residual staleness, and the adaptive pipeline benefits from explicitly treating freshness as part of retrieval behavior.

How to cite this article:

Sonapuram K (June 08, 2026) Adaptive Retrieval-Augmented Generation Pipelines for Dynamic Document Landscapes in the Insurance Industry: A Comparative Study of Retrieval Strategies for Compliance-Aware Response Generation. *Cureus J Comput Sci* 3 : es44389-026-00107-w. DOI <https://doi.org/10.7759/s44389-026-00107-w>

Why the post-amendment contrast is not binary. The v0-only static baselines were deliberately left on the v0 index after v1 became authoritative for version-sensitive scoring, while Static-Reindexed, Partial-Update, and the adaptive path used v1-aligned retrieval. Under that protocol, high SCR combined with 0% VRP and CAS on the v0-only systems is not a mysterious failure mode; it is largely by construction when gold labels require v1 passages and the frozen index still surfaces v0 chunks. The adaptive pipeline's strong post-amendment scores should therefore be interpreted as evidence that update-aware routing recovers version-aligned retrieval in this harness without requiring a manual full re-indexing pass, rather than as a claim that field systems uniformly exhibit 0% VRP when indexes lag. The Static-Reindexed comparison further sharpens this reading: the adaptive pipeline does not outperform a fully re-indexed corpus once the operator has perfect amendment knowledge, since the two are statistically equivalent on this corpus, but it strictly outperforms partial re-indexing, which is the more realistic operational regime. The architectural value therefore lies in the drift-aware trigger that decides when re-indexing is required, rather than in a head-to-head retrieval win over an operator with perfect amendment knowledge.

The principal finding is straightforward. The hardest part of this problem is not building a system that performs well against a stable corpus. The harder problem is maintaining reliable behavior after the corpus changes. In the controlled evaluation reported here, all systems looked strong before amendments. The gap emerged only after updated document versions were introduced.

This result reinforces a practical point for enterprise RAG design. In regulated domains, such as insurance, retrieval quality cannot be judged only by launch-day performance. A system that answers well today may become unreliable later if it continues to retrieve text from an outdated snapshot. The model may still sound confident, and the answer may still appear plausible, but the source grounding has already drifted.

Among the v0-only baseline systems, the adaptive pipeline is most similar to the hybrid LangGraph approach because both use layered retrieval logic rather than a single-step search pass [2]. The difference is that the present design adds runtime sensitivity to document change. This extension is more pronounced after amendments than before them, and the result is consistent with broader version-aware retrieval work suggesting that update awareness is more effective when built into retrieval behavior rather than handled solely as an external maintenance process.

Latency and Cost

Per-query retrieval latency was measured across three trials of 106 queries (318 samples per system) and is summarized in Figure 6. The three v0-only static baselines and the two Partial-Update/Static-Reindexed baselines all sit below 0.25 ms at the median. The adaptive pipeline sits at P50 = 1.85 ms, P95 = 3.47 ms, and P99 = 4.11 ms, approximately 10-fold P50 overhead relative to plain hybrid retrieval. The overhead is attributable to LangGraph orchestration of the four-node graph (drift gate, adaptation router, hybrid retrieval, compliance validation). Operationally, drift scoring can be amortized over document batches, retrieval can short-circuit when no drift flag applies, and compliance validation can be tiered by policy class. For applied deployments, this trade-off motivates selective application of the adaptive path to high-risk query classes such as compliance, exclusions, and sublimits, rather than to every query.

External Validity

Generalization beyond this study is constrained in four respects. First, the evaluation uses 106 version-sensitive queries, sufficient for paired bootstrap inference but smaller than would be required for sub-population claims at national scale. Second, the corpus and amendment scenarios are synthetic rather than drawn from live policy-administration feeds. Third, jurisdictional metadata is scaffolded on every query across five US states, but the active retrieval policy still operates on the v0/v1 binary axis rather than filtering by jurisdiction or effective date. Fourth, metrics are tied to a binary v0/v1 authoritative snapshot, which sharpens contrasts relative to messy partial updates; the Partial-Update baselines partially address this concern by interpolating along the refresh axis. The reported evidence supports the version-aware retrieval and drift-aware routing protocol under a defined experimental setting and does not constitute proof of production performance in operational environments.

How to cite this article:

The current study has additional limits. Real insurance environments contain many more document types, edge cases, and cross-references than the simulated setup used here. The drift detection threshold also requires manual calibration: set too conservatively, the system may miss meaningful changes; set too aggressively, it may trigger unnecessary re-routing. The τ -sweep reported above shows that the threshold is robust across a wide range on the present corpus, but more demanding conditions such as near-duplicate paraphrase updates or partial-meaning shifts require a richer corpus to stress-test. The compliance score reported here uses the deterministic retrieval-grounded definition given in "Materials and Methods" section, so the headline numbers do not depend on any language model judge.

These limitations point to several next steps. The first priority is to release an open versioned-RAG benchmark assembled from publicly available regulatory filings, such as National Association of Insurance Commissioners System for Electronic Rate and Form Filing (NAIC SERFF) state filings, the ISO commercial-form library, and Lloyd's standard wordings, so that drift-aware retrieval can be evaluated outside synthetic harnesses. A second priority is to operationalize the in-force-date and jurisdictional-scoping signals already scaffolded in the corpus, by extending the routing function to filter by jurisdiction and as-of date rather than only by post-amendment phase. Additional directions include incremental update strategies that reduce the need for full re-indexing after every document change; application of the same architecture in related domains such as legal contract management, financial reporting, and pharmaceutical compliance; and stress conditions such as noisy amendments, staggered index refresh, or overlapping v0/v1 applicability windows, so that post-amendment metrics are less idealized and closer to operational reality.

Conclusions

This paper examined an adaptive RAG pipeline for insurance document environments in which the underlying corpus changes over time. The system combines drift detection, query routing, hybrid retrieval, and compliance validation, and was evaluated on an expanded versioned corpus of eight insurance policy forms with 30 amended provisions and 106 version-sensitive queries. The proposed pipeline outperformed three v0-only static baselines (FAISS, MMA-RAG text proxy, hybrid BM25-dense) by a wide margin and outperformed a Partial-Update baseline at both 30% and 60% refresh. Against a Static-Reindexed (v1) baseline that was given perfect amendment knowledge, the proposed pipeline achieved statistically equivalent retrieval scores. The architectural value of the system therefore lies in the drift-aware trigger that decides when re-indexing is required, not in a head-to-head retrieval win over an operator who already knows what changed. The broader point is not limited to insurance. Any domain that relies on authoritative documents that change over time faces the same underlying risk: a retrieval system may continue to sound useful even after its grounding has become outdated. For regulated settings, freshness cannot be treated as an afterthought. The contribution of this study is therefore modest but practical. It does not claim to solve the production problem in full. What it does show is that document change can be modeled directly inside the retrieval workflow, that the resulting drift-aware trigger is statistically equivalent to perfect-knowledge full re-indexing on this corpus, and that the same trigger strictly outperforms partial re-indexing, which is the more realistic operational regime. Future work should release an open versioned-RAG benchmark assembled from publicly available regulatory filings, operationalize the in-force-date and jurisdictional-scoping signals already scaffolded in the corpus, and replicate the protocol on enterprise or redacted production samples to strengthen external validity.

Appendices

This appendix addresses concerns common to systems and applied-AI reviews: robustness to the fixed cosine threshold used for drift detection, and end-to-end latency for the full pipeline.

Drift threshold τ . The main study uses $\tau = 0.92$. To test whether this choice is fragile, the drift detector was re-evaluated at six values of $\tau \in \{0.85, 0.88, 0.90, 0.92, 0.94, 0.96\}$ on the finer drift-evaluation chunking (40-token windows, 8-token overlap), which yields 60 chunks total: 47 amended (positives) and 13 unchanged (true negatives). At every value of τ in the sweep, the detector achieved precision = recall = F1 = accuracy = 1.0 (Table 2). The drift function applies a paired-cosine check together with a text-equality fast path; on a corpus where amendments produce literal text changes, the

How to cite this article:

fast path resolves all cases before the cosine threshold is consulted. The threshold therefore acts as a robust safety check on this corpus rather than as the operative discriminator. More demanding tests, such as near-duplicate paraphrase updates or partial-meaning shifts, require a richer corpus to expose threshold sensitivity and are identified as a priority for future work.

τ	Precision	Recall	F1	TP	FP	TN
0.85	1	1	1	47	0	13
0.88	1	1	1	47	0	13
0.9	1	1	1	47	0	13
0.92	1	1	1	47	0	13
0.94	1	1	1	47	0	13
0.96	1	1	1	47	0	13

TABLE 2: Drift-detector sensitivity to the cosine threshold τ .
Sweep performed on the finer drift-evaluation chunking (40/8). All six threshold values yield identical perfect performance because the text-equality fast path resolves every case before the cosine threshold is consulted on this corpus.

End-to-end latency. Per-query retrieval latency was measured with a high-resolution timer over three trials of 106 queries (318 samples per system) under the deterministic local profile and manuscript chunking (Table 3). The three v0-only static baselines and the two Partial-Update/Static-Reindexed baselines all sit below 0.25 ms at the median. The proposed adaptive pipeline sits at P50 = 1.85 ms, P95 = 3.47 ms, and P99 = 4.11 ms, approximately 10-fold P50 overhead relative to plain hybrid retrieval. The overhead is attributable to LangGraph orchestration of the four-node graph (drift gate, adaptation router, hybrid retrieval, compliance validation). The component ablation reported in the main text shows that disabling individual non-drift components (query routing, compliance validation) does not appreciably reduce latency on this synthetic harness, because the overhead is dominated by the orchestrator dispatch rather than by the work performed at each node.

System	P50 (ms)	P95 (ms)	P99 (ms)
FAISS static (v0)	0.02	0.05	0.1
Hybrid BM25–dense (v0)	0.17	0.41	0.52
MMA-RAG (text proxy)	0.2	0.41	0.62
Static-Reindexed (v1)	0.21	0.53	0.66
Partial-Update (30%)	0.19	0.44	0.64
Partial-Update (60%)	0.18	0.38	0.58
Adaptive (proposed)	1.85	3.47	4.11

TABLE 3: End-to-end retrieval latency per system.

Three trials of 106 queries each (318 samples per system), manuscript chunking, deterministic local profile. The overhead of the adaptive pipeline reflects LangGraph orchestration of the four-node graph and motivates selective application of the adaptive path to high-risk query classes (compliance, exclusions, sublimits) rather than to every query. MMA, multi-modal agent; RAG, retrieval-augmented generation; FAISS, Facebook AI similarity search; BM25, Best Matching 25.

Token-level and dollar-level cost analysis under a cloud-application programming interface (API) profile is identified as a follow-on item; the local profile reported here uses a deterministic hash-based embedding backend with no per-query API charges, so the headline costs of the present study are zero by construction.

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Karteek Sonapuram

Acquisition, analysis, or interpretation of data: Karteek Sonapuram

Drafting of the manuscript: Karteek Sonapuram

Critical review of the manuscript for important intellectual content: Karteek Sonapuram

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue. **Animal subjects:** All authors have confirmed that this study did not involve animal subjects or tissue. **Conflicts of interest:** In compliance with the ICMJE uniform disclosure form, all authors declare the following: **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work. **Financial relationships:** All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted work. **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

How to cite this article:

Data Availability Statements

The evaluation corpus is synthetic insurance policy text produced by the published pipeline (scripts/build_synthetic_data.py), with chunk manifests under data/manifests/. The code, configuration, and bundled artifacts used for the experiments are available in the public GitHub repository <https://github.com/Karteeks505/AdaptiveRAG> (release v1.0.0). A mirrored archive of the same materials for download is hosted on Google Drive: https://drive.google.com/file/d/1RxP1jYMnZAcaOvfEuntjDpgeVQ26s9gb/view?usp=drive_link (accessed 2026-04-14).

References

1. Brancati I: [Generative artificial intelligence for insurance: developing a chatbot for multi-document summarization of insurance products](#). Master's thesis, Politecnico di Torino. 2025, 1-135.
2. Ercole D: [AI agents leveraging RAG and MCP for insurance knowledge management and enterprise workflow automation](#). Master's thesis, Politecnico di Torino. 2025, 1-102.
3. Krayem I, Ghourabi M, Al Assaad M: [MMA-RAG: multi-modal agents for insurance document processing with retrieval-augmented generation](#). Intelligent Systems and Applications. IntelliSys 2025. Lecture Notes in Networks and Systems. Arai K (ed): Springer, Cham; 2025. 1554:10.1007/978-3-031-99965-9_13
4. Vaswani A, Shazeer N, Parmar N, et al.: [Attention is all you need](#). Advances in Neural Information Processing Systems. 2017, 30:5998-6008.
5. Brown T, Mann B, Ryder N, et al.: [Language models are few-shot learners](#). Advances in Neural Information Processing Systems. 2020, 33:1877-1901.
6. Johnson J, Douze M, Jegou H: [Billion-scale similarity search with GPUs](#). IEEE Transactions on Big Data. 2019, 7:535-547. 10.1109/tbdata.2019.2921572
7. Hevner AR, March ST, Park J, Ram S: [Design Science in Information Systems Research](#). MIS Quarterly. 2004, 28:75-106. 10.2307/25148625
8. Anthropic: [Model Context Protocol Specification \(open standard\)](#). (2024). Accessed: April 6, 2026: <https://modelcontextprotocol.io/specification/2024-11-05/>.
9. Vu T, Iyyer M, Wang X, et al.: [FreshLLMs: refreshing large language models with search engine augmentation](#). Findings of the Association for Computational Linguistics: ACL 2024. 2024, 13697-13720. 10.18653/v1/2024.findings-acl.813
10. Lyu Y, Li Z, Niu S, et al.: [CRUD-RAG: a comprehensive Chinese benchmark for retrieval-augmented generation of large language models](#). ACM Transactions on Information Systems. 2025, 43:1-32. 10.1145/3701228
11. Fan W, Ding Y, Ning L, et al.: [A survey on RAG meeting LLMs: towards retrieval-augmented large language models](#). Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 2024, 6491-6501. 10.1145/3637528.3671470
12. Es S, James J, Espinosa Anke L, Schockaert S: [RAGAs: automated evaluation of retrieval augmented generation](#). Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations. 2024, 150-158. 10.18653/v1/2024.eacl-demo.16

How to cite this article:

Sonapuram K (June 08, 2026) Adaptive Retrieval-Augmented Generation Pipelines for Dynamic Document Landscapes in the Insurance Industry: A Comparative Study of Retrieval Strategies for Compliance-Aware Response Generation. Cureus J Comput Sci 3 : es44389-026-00107-w. DOI <https://doi.org/10.7759/s44389-026-00107-w>