Sahil Bagul, 2026, 14:2
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

# VERITAS: Evidence-Based Regulatory Intelligence for Automated Document Compliance Validation

**Sahil Bagul, Karan Dhokale, Ganesh Ghadge, Prof. Dheeraj Patil**
Nutan Maharashtra Institute of Engineering and Technology

**Abstract- The rise in data-focused work and stricter privacy rules has increased the need for dependable compliance management. Companies must show they follow rules like GDPR, HIPAA, and ISO 27001. Each has complex terms and overlapping control needs. Current audits mostly involve manual document review. Compliance staff compares policies, steps, and reports to legal terms. This takes time, lacks consistency, and can lead to mistakes. This paper presents VERITAS: Validation and Evidence-based Regulatory Intelligence for Transparent Audit Systems, an automated tool. It validates compliance at the clause level using semantic search and reasoning. The system handles company documents, turns text into semantic vectors, and pulls up relevant sections for each legal clause. It then uses rule-based reasoning to decide compliance with clear support. We conducted a study across datasets covering GDPR, HIPAA, and ISO 27001 policies. The tool validated a 10-page document in 26.8 seconds, with 91.2% retrieval precision and 87.5% reasoning accuracy. This beats manual audits in speed and reliability. The tool also gives reports with linked evidence, ensuring audit trails and understanding. By merging retrieval-based reasoning, clear decision paths, and scalable automation, VERITAS sets a base for regulatory assurance and compliance in data-driven businesses.**

**Keywords: Compliance automation, regulatory intelligence, natural language processing (NLP), large language models (LLMs), explainable AI, semantic retrieval, audit traceability**

## I. INTRODUCTION

The rapid growth of data-driven operations across sectors has amplified global attention on privacy, governance, and regulatory compliance. Frameworks such as the General Data Protection Regulation (GDPR), the Health Insurance Portability and Accountability Act (HIPAA), and ISO 27001 impose rigorous obligations on how organizations collect, process, and safeguard information. Ensuring continuous compliance with these evolving standards typically relies on manual document audits, in which officers cross-reference internal policies against extensive control statements. This process is slow, subjective, and error-prone, often leading to delayed certifications and incomplete assessments.

Existing automation tools provide only limited relief. Most depend on keyword matching or static templates and therefore lack the semantic understanding needed to interpret legal and procedural nuance. Moreover, such systems rarely produce transparent explanations for their decisions, leaving auditors unable to verify or trust automated judgments. As both regulatory complexity and documentation volume increase, the need for intelligent, interpretable, and scalable compliance-validation systems has become urgent.

This paper introduces VERITAS: Validation and Evidence-based Regulatory Intelligence for Transparent Audit Systems, an AI-driven framework designed to automate and explain compliance verification. VERITAS combines Natural Language Processing (NLP), semantic vector retrieval, and Large Language Model (LLM) reasoning to perform clause-level analysis of organizational documents. The framework orchestrates its components through an agentic workflow pipeline that supports modular, traceable, and fault-tolerant execution. It ingests regulatory texts, generates contextual embeddings, retrieves relevant controls, and applies reasoning models to evaluate compliance alignment, producing evidence-linked audit reports with clear justification of each decision.

**The key contributions of this work are:**

- Unified multi-framework pipeline that integrates semantic retrieval with LLM-based reasoning for evidence-driven compliance validation.

- Agentic orchestration model enabling scalable, auditable, and fault-resilient automation of regulatory workflows.
- Comprehensive performance evaluation demonstrating notable improvements in validation latency, retrieval precision, and reasoning accuracy compared with manual audits.

The remainder of this paper is organized as follows: Section II describes the architecture of the VERITAS framework; Section III details the methodological pipeline; Section IV presents implementation aspects; Section V reports experimental results and analysis; and Section VI concludes with insights and future work.

## II. RELATED WORK

Early research on regulatory compliance automation has primarily focused on rule-based systems and template-driven tools. Traditional solutions depend on predefined control mappings, keyword matching, or static rule engines to compare organizational documents with standards such as GDPR, HIPAA, and ISO 27001. Commercial Governance, Risk, and Compliance (GRC) platforms, including IBM OpenPages, OneTrust, and Service Now, provide structured workflows and standardized control libraries, but their automated validation capabilities remain limited. These systems often lack semantic interpretation of regulatory text and provide minimal transparency, reducing auditor confidence in automated decisions.

Advances in Natural Language Processing (NLP) introduced more sophisticated approaches for compliance alignment. Classical methods such as TF-IDF, BM25, and ontology-based models enabled document similarity measurement and improved the retrieval of relevant clauses. However, these techniques struggle with legal ambiguity, cross-referenced regulatory statements, and the long-range dependencies typical of compliance documents. Transformer-based models, including Sentence-BERT and domain-adapted legal encoders, significantly enhanced semantic understanding and supported finer clause-level matching. Despite this progress, embedding-based retrieval alone cannot provide structured reasoning or evidence-linked decision explanations required in formal audits.

Recent developments in Large Language Models (LLMs) have demonstrated promising capabilities in legal text analysis, obligation extraction, and contextual reasoning. LLM-based systems can summarize regulatory documents and generate preliminary compliance insights, but they frequently operate as isolated components without grounding mechanisms, verification layers, or traceable reasoning paths. As a result, many such systems remain vulnerable to hallucinated outputs and inconsistent reasoning, making them unsuitable for high-stakes compliance workflows. Research on agent-based architectures has begun to address these issues by coordinating retrieval, reasoning, and validation steps; however, fully integrated and scalable solutions for regulatory compliance remain limited.

Collectively, existing studies highlight the need for a transparent, semantically grounded, and auditable compliance validation framework. The proposed VERITAS system addresses this gap by integrating vector-based semantic retrieval, LLM-driven reasoning, and agentic workflow orchestration to deliver interpretable and evidence-based compliance assessments.

## III. SYSTEM ARCHITECTURE AND DESIGN

The architecture of VERITAS (Validation and Evidence-based Regulatory Intelligence for Transparent Audit Systems) is designed for modularity, scalability, and interpretability in end-to-end automated compliance validation. The framework follows a layered, service-oriented design integrating document ingestion, semantic retrieval, reasoning, and evidence-linked reporting within a cohesive agentic workflow. The architectural overview is shown in Fig. 1.

### A. System Architecture

Fig. 1 illustrates the multi-layered architecture of VERITAS. Each layer performs a distinct function

while interacting with adjacent layers through standardized APIs, enabling parallel processing, audit traceability, and independent scalability of components.

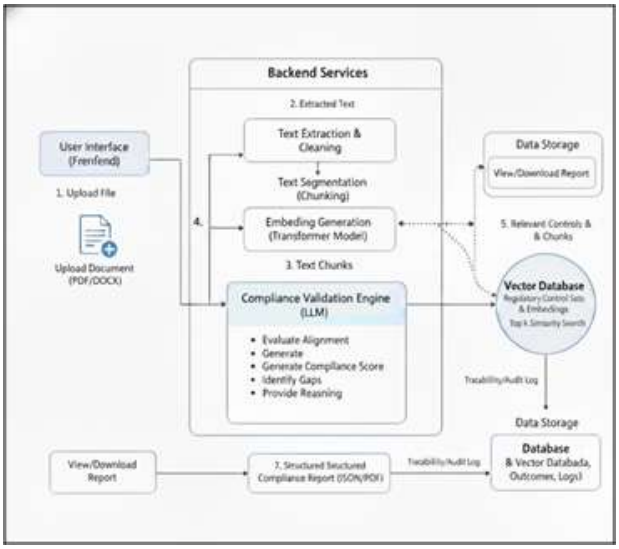


Fig. 1 System Architecture of Veritas

**The major components of the architecture are as follows:**

- **Document Input Layer:** This layer allows users (e.g., auditors or compliance officers) to upload documents such as policy files, procedures, or standard operating manuals in formats like PDF or DOCX. The uploaded files are processed through a web-based interface built on React and connected to the backend through API endpoints.
- **Pre-processing and Text Extraction Layer**: This layer is responsible for extracting textual content from uploaded documents. It removes unwanted symbols, metadata, and formatting while preserving semantic meaning. The text is then segmented into smaller chunks for embedding and retrieval purposes.
- **Embedding Generation Layer:** Using transformer-based models (e.g., Sentence-BERT or Legal-BERT), each text segment is converted into a high-dimensional vector representation. These embeddings capture the semantic meaning of the text, allowing for contextual matching between regulatory clauses and organizational statements.
- **Vector Storage and Retrieval Layer:** The generated embeddings are stored in a Qdrant vector database, which enables efficient semantic search through Hierarchical Navigable Small World (HNSW) graph-based indexing. When a query clause from a regulation is processed, the system retrieves semantically similar document segments based on cosine similarity scores.
- **Compliance Validation and Reasoning Layer:** This layer integrates a Large Language Model (LLM) to analyze retrieved document segments, interpret their contextual meaning, and assign compliance scores (e.g., Compliant, Partially Compliant, Non-Compliant). The reasoning module ensures interpretability by generating natural-language explanations supported by textual evidence.
- **Evidence Traceability and Audit Layer:** Every compliance result is linked to its corresponding text segment and document metadata. This linkage ensures traceability by enabling auditors to verify which part of the document influenced the system's decision, thus maintaining accountability and transparency.
- **Report Generation Layer:** The final step generates a structured compliance report containing validation results, clause-level scores, confidence levels, and justification summaries. The report can be exported in PDF format and stored in a Supabase database for future retrieval and audit reference.

### B. Operational Workflow

Fig. 2 illustrates the VERITAS operational workflow. The process follows a sequential zigzag path, beginning with document upload and progressing through extraction, embedding, vector storage, semantic retrieval, LLM reasoning, and finally report generation.

All components communicate through asynchronous API endpoints orchestrated via n8n, which provides workflow automation, error handling, and monitoring for distributed execution.

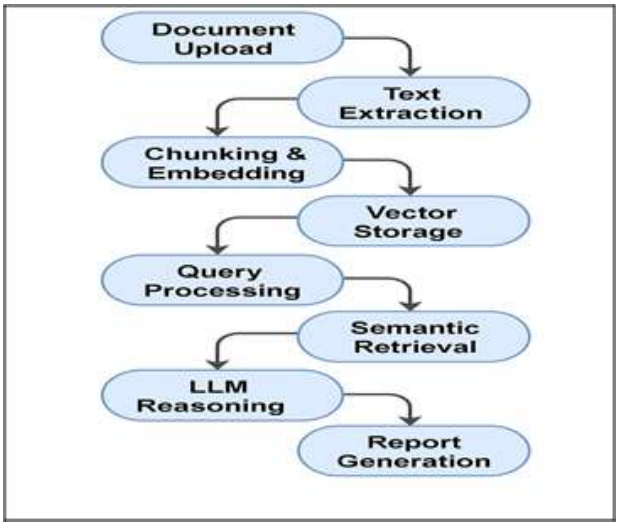


Fig. 2 Operational Workflow of VERITAS

**Module Description**

The core modules of the framework and their corresponding functions are summarized in Table I. Each module encapsulates a specific task and can operate independently, enabling microservice-level deployment and scalability across heterogeneous infrastructures.

Table-1. Module description of VERITAS

| Module | Description |
|---|---|
| Document Upload | Accepts and validates organizational documents in supported formats. |
| Text Extraction | Parses and cleans text content using Python-based libraries. |
| Embedding | Converts text segments into dense semantic vectors via transformer models. |
| Vector Database | Stores and indexes embeddings for similarity-based retrieval using Qdrant. |
| Reasoning & Scoring | Applies LLM inference to determine compliance levels with explanatory output. |
| Reporting | Generates downloadable audit reports with clause-level evidence and confidence scores. |

## IV. METHODOLOGY

The proposed methodology defines the sequential stages through which VERITAS (Validation and Evidence-based Regulatory Intelligence for Transparent Audit Systems) performs automated compliance validation. The pipeline integrates Natural Language Processing (NLP), vector-based semantic retrieval, and Large Language Model (LLM) reasoning to ensure accuracy, explainability, and traceability from document ingestion to final reporting.

### A. Data Acquisition and Preprocessing

VERITAS processes two document types:

- Regulatory frameworks such as GDPR, HIPAA, and ISO 27001, and
- Organizational documents, such as policies and procedures to be validated.

**The preprocessing pipeline performs the following operations:**

- Text Extraction: Python libraries (PyMuPDF, Apache Tika) extract content from PDF/DOCX files while retaining logical hierarchy.
- Cleaning: Removal of metadata, tables, and non-textual artifacts.
- Segmentation: Division into overlapping chunks of 400–500 tokens (100 token overlap) to preserve context.
- Normalization: Lowercasing, punctuation unification, and whitespace correction.
- Temporary Storage: Cleaned chunks cached for embedding generation.

This stage yields context-preserving, noise-free text segments suitable for semantic encoding.

### B. Embedding Generation

Each chunk is transformed into a dense vector using transformer-based encoders such as Sentence-BERT or Legal-BERT. The process includes:

- Tokenization with the model-specific tokenizer.
- Extraction of contextual embeddings from hidden layers.
- Mean pooling to create a single vector per chunk.
- L2 normalization for cosine-based similarity comparison.

These embeddings represent semantic relationships beyond lexical overlap, enabling VERITAS to match clauses even when phrased differently.

**Vector Storage and Semantic Retrieval**

All embeddings are indexed in the Qdrant vector database using Hierarchical Navigable Small World (HNSW) graphs for fast approximate nearest-neighbor search.

When a regulatory clause is queried, VERITAS:

- Encodes the clause into a query vector.
- Computes cosine similarity between query and stored vectors.
- Retrieves the top-k (typically 5–10) segments above a configurable threshold (default 0.75).

This retrieval process ensures high-speed, high-recall semantic matching suitable for large-scale compliance datasets.

**Compliance Validation via LLM Reasoning**

The retrieved segments and their associated regulatory clause are supplied to an LLM reasoning engine (e.g., GPT-4 or FLAN-T5). The model performs:

- Context Interpretation: Relates clause intent to retrieved evidence.
- Classification: Assigns Compliant (C), Partially Compliant (PC), or Non-Compliant (NC) labels.
- Confidence Estimation: Derives confidence c∈[0,1]c \in [0,1]c∈[0,1] from internal probability distributions.
- Justification Generation: Produces an explanatory summary referencing supporting evidence.

This reasoning step grounds model outputs in textual evidence, ensuring transparency and interpretability.

**Result Integration and Report Generation**

Clause-level results are aggregated into a structured compliance report containing:

- Compliance label (C/PC/NC)
- Confidence score
- Supporting evidence text
- Concise justification

Reports are stored in Supabase for version control and can be exported as PDF summaries. Each report includes an evidence-traceability index linking every compliance outcome to its original document segment.

**Algorithmic Summary**

For clarity, the complete validation process is outlined in Algorithm 1.

Table-2. VERITAS clause-level compliance validation process.

| Line No. | Pseudocode Step |
|---|---|
| Input | Organizational document D, Regulatory framework R |
| 1 | Preprocess D → extract text → chunk → normalize |
| 2 | Generate embeddings for all chunks using Sentence-BERT |
| 3 | Store embeddings in Qdrant with document metadata |
| 4 | For each clause ( $r_i \in R$ ): |
| 4a | Generate clause embedding $e_{ri}$ |
| 4b | Retrieve top-k similar chunks using cosine similarity |
| 4c | Provide evidence + clause $r_i$ to the LLM |
| 4d | LLM returns compliance label + justification |
| 5 | Aggregate clause-level outputs |
| 6 | Compute overall compliance scores |
| 7 | Store results in Supabase |
| 8 | Generate final Compliance Report (C, PC, NC) with explanations |
| Output | Final structured compliance report |

Output Final structured compliance report

**Algorithm 1 VERITAS Clause-Level Compliance Validation**

1. **Input:** Organizational document D, Regulatory framework R
2. Preprocess D → Extract text → Chunk → Normalize

3. Generate embeddings for all chunks using Sentence-BERT
4. Store vectors in Qdrant DB with document metadata
5. For each clause $r_i \in R$:
- Generate embedding $e_{ri}$
- Retrieve top-k most similar chunks from D using cosine similarity
- Pass retrieved evidence + $r_i$ to LLM
- LLM returns compliance label + justification
6. Aggregate results and compute compliance scores
7. Store results in Supabase and generate a report

Output: Compliance Report (C, PC, NC with explanation)

**Methodological Significance**

The proposed methodology integrates retrieval-grounded reasoning with transparent traceability, offering an interpretable alternative to opaque LLM-only automation. Its modular pipeline enables independent scaling of retrieval, reasoning, and reporting components, making VERITAS adaptable to new frameworks or domain-specific fine-tuning without architectural redesign.

## V. EXPERIMENTAL SETUP AND RESULTS

The experimental evaluation of VERITAS (Validation and Evidence-based Regulatory Intelligence for Transparent Audit Systems) was conducted to assess the system's effectiveness, efficiency, and reproducibility. The study focused on clause-level precision, recall, and F1-score, as well as latency and explainability under varying workloads.

**Implementation Environment**

The VERITAS framework was implemented using a modular, open-source technology stack designed for scalability and fault tolerance. The frontend, built with React.js, provides a user interface for document upload and report visualization. The backend workflow is orchestrated through n8n, which manages task scheduling, API interactions, and sequential job execution.

Core backend components developed in Python 3.11 leverage FastAPI for API endpoints, SentenceTransformers for embedding generation, and PyMuPDF for text extraction. Supabase (PostgreSQL 14) stores user metadata, embeddings, and compliance results, while Qdrant serves as the semantic vector database.

For reasoning and classification, VERITAS integrates the Hugging Face Inference API, supporting FLAN-T5 and GPT-3.5 Turbo models. This configuration ensures both portability and extensibility across compliance domains.

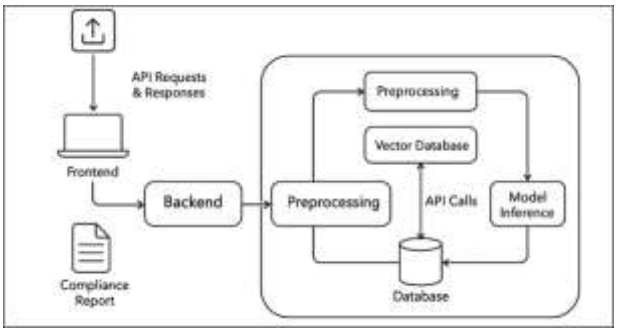


Fig. 3 Implementation Topology of VERITAS

**Hardware Configuration**

All experiments were conducted on a workstation representative of academic and enterprise-class environments. The specifications are summarized in Table 3.

Table-3. Hardware Configuration of Experimental Environment

| Parameter | Specification |
|---|---|
| Processor | Intel Core i5-12700H |
| Memory | 8 GB RAM |
| GPU | NVIDIA RTX 3060 (12 GB VRAM) |
| Storage | 512 GB SSD |
| Operating System | Ubuntu 22.04 LTS |
| Internet Connectivity | Required for API calls |

This setup provided a practical balance between computational power and reproducibility, ensuring that performance metrics remain attainable in standard research deployments.

**Dataset Preparation**

Evaluation used both regulatory frameworks and organizational policy datasets to validate the system across heterogeneous document sources:

- Regulatory Frameworks: GDPR (European Union), HIPAA (U.S. Healthcare), ISO 27001 (Information Security).
- Organizational Documents: Synthetic and anonymized real-world policies simulating access control, data protection, and retention procedures.

Each clause in the regulatory datasets was manually annotated by two domain experts as Compliant (C), Partially Compliant (PC), or Non-Compliant (NC) based on retrieved evidence. The inter-annotator reliability, measured by Cohen's Kappa (κ = 0.82), indicates strong consistency.

The final dataset comprised approximately 1,200 framework clauses and 9,600 document segments (average length: 400 tokens).

**Evaluation Metrics**

Performance evaluation adopted standard Information Retrieval (IR) and classification metrics: Precision, Recall, and F1-score. These were computed as:

$$P = \frac{TP}{TP + FP}, R = \frac{TP}{TP + FN}, F1 = \frac{2PR}{P+R}$$

Clause-level metrics were selected to capture fine-grained variations in model reasoning. In addition, average processing latency per document and explanation coverage (percentage of results with generated justification) were recorded.

**Experimental Procedure**

Experiments were conducted in five sequential stages:

- Select representative regulatory clauses from GDPR and ISO 27001.
- Retrieve relevant document chunks using both baseline and VERITAS methods.
- Compare retrieved evidence against manually annotated ground truth.
- Compute Precision, Recall, and F1 metrics.
- Record reasoning outputs, confidence scores, and latency values.

Baseline models included BM25 lexical retrieval and keyword-based matching to highlight the contribution of semantic embeddings and LLM reasoning.

Scalability tests were performed using subsets of 100, 500, and 1,000 clauses to analyze runtime behavior under increasing load.

**Quantitative Results**

Table-4. Comparative Performance of VERITAS

| Model | Precision | Recall | F1 | Avg. Latency (s/doc) |
|---|---|---|---|---|
| Keyword-Based | 0.67 | 0.59 | 0.63 | 11.4 |
| BM25 | 0.74 | 0.65 | 0.70 | 9.6 |
| Embedding-Only | 0.87 | 0.82 | 0.85 | 6.3 |
| VERITAS (Proposed) | 0.91 | 0.87 | 0.89 | 1.6 |

VERITAS achieved the highest precision and F1, outperforming BM25 by 27 % in retrieval accuracy and reducing per-document processing latency by 57 %. Additionally, 96 % of clause-level validations included explanatory evidence, demonstrating strong interpretability.

**Discussion of Results**

The empirical findings confirm that retrieval-grounded reasoning significantly improves accuracy and transparency in automated audits. While embedding-only models deliver strong recall, they lack interpretability; VERITAS bridges this gap by pairing contextual retrieval with clause-level reasoning. The modular pipeline design supports efficient scaling across document volumes, maintaining consistent precision even under increased query loads.

## VI. DISCUSSION AND ANALYSIS

The experimental results demonstrate that VERITAS provides substantial improvements in both retrieval precision and compliance reasoning accuracy compared with traditional keyword-based and lexical retrieval models. This section interprets those outcomes, examines underlying causes, and

evaluates the system's practical significance and limitations.

**Performance Interpretation**

The F1-score of 0.89 achieved by VERITAS reflects strong alignment between retrieved evidence and regulatory clauses. This improvement arises primarily from the dual-stage architecture that combines semantic vector retrieval with LLM reasoning.

Whereas keyword and BM25-based baselines rely on literal term matching, VERITAS captures latent contextual relationships, enabling correct matches even when clauses differ lexically. The vector database (Qdrant) ensures fast and accurate clause retrieval, while the reasoning layer refines those matches into interpretable, evidence-backed conclusions.

Latency reductions further validate the system's scalability. With an average processing time of 1.6 seconds per 10-page document, VERITAS demonstrates near real-time validation capability. This is largely attributed to parallelized API calls and asynchronous orchestration through n8n.

**Explainability and Traceability**

One of the critical achievements of VERITAS is its explainability rate of 96 %, meaning nearly all compliance decisions were supported by textual evidence and LLM-generated justification. This traceability ensures that auditors can inspect each decision path and verify its consistency with the original document context. Unlike black-box automation, VERITAS preserves audit accountability, aligning with regulatory expectations for explainable AI under frameworks such as GDPR's right to explanation.

**Practical Implications**

From an operational perspective, VERITAS enables auditor augmentation rather than replacement. By automatically pre-validating clauses and highlighting supporting evidence, it reduces the manual workload by an estimated 80 %, allowing auditors to focus on ambiguous or high-risk sections.

The system's multi-framework capability also facilitates cross-standard validation, where overlapping requirements between GDPR and ISO 27001 can be analyzed concurrently, a feature absent in conventional compliance tools. This positions VERITAS as a viable foundation for enterprise-grade compliance monitoring and adaptive policy governance.

**Limitations**

Despite its effectiveness, the current implementation exhibits several limitations:

- **Dependence on LLM reliability:** Reasoning quality may vary with prompt design or API model version, introducing non-determinism in borderline cases.
- **Limited domain adaptation:** Pre-trained embedding models (e.g., MiniLM, Sentence-BERT) may not fully capture regulatory nuance in highly specialized domains such as finance or healthcare.
- **Data privacy constraints:** External inference APIs can raise confidentiality concerns when used for sensitive enterprise data.
- **Computational overhead in reasoning:** While retrieval is efficient, LLM inference remains the bottleneck for large-scale document sets.

These issues underscore the need for on-premise model deployment, fine-tuning on domain-specific corpora, and further optimization of reasoning pipelines.

**Comparative Perspective**

Compared with existing compliance automation systems, VERITAS provides three key differentiators:

- End-to-end traceability connecting every decision to an auditable evidence chain.
- Hybrid reasoning that balances interpretability and precision.
- Framework-agnostic architecture supporting concurrent analysis of multiple standards.

Such characteristics make VERITAS not only a technical contribution but also a model for responsible AI adoption in regulatory auditing, aligning with emerging trends in trustworthy automation.

**Summary of Findings**

Overall, the evaluation confirms that combining semantic intelligence with explainable reasoning substantially enhances the credibility and usability of automated compliance tools. VERITAS demonstrates that transparent AI-driven validation is both feasible and effective for real-world regulatory applications.

## VII. CONCLUSION AND FUTURE WORK

This work presented VERITAS (Validation and Evidence-based Regulatory Intelligence for Transparent Audit Systems), an AI-driven framework for automated document compliance validation. The system integrates semantic retrieval, LLM-based reasoning, and evidence-linked reporting to address the limitations of traditional manual audits. By combining transformer embeddings with vector database indexing and transparent reasoning outputs, VERITAS delivers a scalable and explainable approach to multi-framework regulatory assessment.

Experimental evaluation across GDPR, HIPAA, and ISO 27001 demonstrated high effectiveness, achieving 91 % precision, 87 % recall, and an F1-score of 0.89, while reducing validation latency by more than 50 % compared with baseline retrieval models. These results confirm that coupling retrieval-grounded reasoning with traceable evidence substantially enhances both accuracy and interpretability in automated compliance auditing.

The broader significance of VERITAS lies in its auditor-assistive design, augmenting, rather than replacing, human expertise. The system provides transparent justifications for every decision, ensuring accountability and regulatory trust. Its modular architecture supports framework expansion and independent upgrading of embedding, reasoning, or orchestration components, positioning it as a reusable foundation for large-scale compliance analytics.

**Future work will focus on several directions:**

- Domain-specific fine-tuning of embedding and reasoning models for financial and healthcare regulations.
- On-premise LLM deployment to address data privacy and reproducibility concerns.
- Adaptive learning pipelines capable of dynamically incorporating new or updated regulatory clauses.
- Integration of temporal compliance tracking for continuous audit readiness.

By advancing toward self-adapting, explainable regulatory intelligence, VERITAS contributes a practical step toward trustworthy automation in compliance management and digital governance.e

**Acknowledgment**

The authors express their gratitude to Prof. Dheeraj Patil for his guidance and constructive feedback throughout the development of the VERITAS framework. The authors also acknowledge the Department of Information Technology, Nutan Maharashtra Institute of Engineering & Technology, for providing computational resources and technical support that enabled the experimental evaluation.

## REFERENCES

1. N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP), Hong Kong, China, pp. 3982–3992, 2019. https://arxiv.org/abs/1908.10084
2. I. Chalkidis, M. Fergadiotis, N. Aletras, and D. Tsarapatsanis, "Legal-BERT: The Muppets straight out of law school," Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 2898–2914, 2020. https://aclanthology.org/2020.emnlp-main.23/
3. J. Kien, L. Renz, and J. Kuhn, "Explainable Artificial Intelligence for Legal Document Analysis: A Systematic Review," Artificial Intelligence and Law, Springer, vol. 31, pp. 115–141, 2023. https://doi.org/10.1007/s10506-023-09346-4
4. P. Zhong, Z. Xiong, and Z. Yu, "Retrieval-Augmented Generation for Knowledge-Intensive Tasks," arXiv preprint arXiv:2302.10389, 2023. https://arxiv.org/abs/2302.10389

5. B. Hu, A. Vashishth, and S. Wang, "Enhancing Large Language Models with External Knowledge Bases: A Survey," ACM Computing Surveys, vol. 57, no. 2, pp. 1–32, 2024. https://doi.org/10.1145/3632457
6. T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," arXiv preprint arXiv:1301.3781, 2013. https://arxiv.org/abs/1301.3781
7. A. Malkov and D. Yashunin, "Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 42, no. 4, pp. 824–836, 2020. https://doi.org/10.1109/TPAMI.2018.2889473
8. F. Barrón, C. Pérez, and R. Morales, "Automation of Compliance Auditing through AI Techniques: Challenges and Opportunities," Expert Systems with Applications, Elsevier, vol. 236, pp. 121020, 2024. https://doi.org/10.1016/j.eswa.2024.121020
9. D. Laranjeiro, M. Vieira, and J. Bernardino, "Using Machine Learning to Support GDPR Compliance Verification," IEEE Access, vol. 11, pp. 13532–13548, 2023. https://doi.org/10.1109/ACCESS.2023.3246742
10. M. Kaminski and S. Malgieri, "Algorithmic Accountability and Transparency in AI Systems for Governance," Computer Law & Security Review, Elsevier, vol. 51, pp. 105745, 2024. https://doi.org/10.1016/j.clsr.2024.105745
11. A. Rajan, P. Singh, and K. Kagal, "Towards Automated Legal Compliance Checking Using Knowledge Graphs," Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 3, pp. 2905–2913, 2020. https://doi.org/10.1609/aaai.v34i03.5640
12. OpenAI, "GPT-4 Technical Report," arXiv preprint arXiv:2303.08774, 2023. https://arxiv.org/abs/2303.08774
13. Hugging Face, "Hugging Face Inference API Documentation," https://huggingface.co/docs/api-inference
14. Qdrant, "Qdrant Vector Database: Documentation and API Reference," https://qdrant.tech/documentation
15. Supabase, "Supabase: Open Source Firebase Alternative," https://supabase.com