DDRP v0.1 — A Deterministic Protocol for Lexical Document Review Without Semantic Inference

Author: Bruce Tisler
Affiliation: Independent Researcher
DOI: 10.5281/zenodo.18427220
Keywords: deterministic systems, document review, lexical analysis, reproducibility, protocol design, auditability

Abstract

Contemporary document review systems increasingly rely on probabilistic language models that produce plausible interpretations while obscuring their internal reasoning and introducing non-reproducibility. This creates challenges for auditability, evidentiary use, and trust—particularly in regulatory, contractual, and organizational contexts where identical inputs must yield identical outputs.

This paper introduces the **Deterministic Document Review Protocol (DDRP) v0.1**, a non-interpretive, fully deterministic protocol for extracting and structuring lexical obligations from documents. DDRP explicitly prohibits semantic inference, probabilistic components, confidence scoring, and compliance judgments. Instead, it operates via transparent, rule-based pattern matching to detect linguistically explicit operators (e.g., requirements, definitions, scope constraints) and to instantiate obligations whose completeness is evaluated solely through the presence or absence of explicitly stated fields.

The protocol emphasizes falsification over validation and includes a hostile audit methodology designed to detect hidden interpretation, nondeterminism, or prohibited capabilities. DDRP does not determine correctness, intent, or compliance; rather, it preserves evidence about **where interpretation would be required**. This paper documents the protocol's design principles, operator taxonomy, obligation model, verification strategy, execution continuity guarantees, and limitations, establishing a reproducible reference suitable for independent implementation and organizational scaling.

1. Motivation and Problem Statement

Document review underpins contracts, policies, standards, and regulatory processes. Increasingly, automated tools are used to summarize, interpret, or assess such documents. However, systems that rely on semantic interpretation—particularly those based on stochastic language models—exhibit several structural weaknesses:

- **Non-reproducibility:** Identical inputs may yield different outputs across runs or environments.

- **Opaque inference:** Conclusions cannot be traced to verifiable textual evidence.
- **Overclaiming:** Systems implicitly judge compliance, correctness, or risk without explicit grounding.
- **Audit fragility:** Outputs cannot be independently verified or falsified.

In high-trust contexts, these weaknesses are unacceptable. The core insight motivating DDRP is that trust begins not with better interpretation, but with **disciplined restraint**. A system that makes fewer claims—but makes them reproducibly—can be more useful than one that claims understanding.

DDRP is designed to operate **prior to interpretation**. It surfaces where obligations are stated, where information is missing, and where scope or causality is asserted—without attempting to resolve meaning.

2. Design Principles

DDRP is governed by the following principles:

- **Determinism over expressiveness**
  Every execution must be reproducible given the same input and ruleset.
- **Lexical evidence over semantic inference**
  Only explicitly present textual operators are considered.
- **Falsification over validation**
  The protocol is designed to be challenged, not trusted by default.
- **Absence reporting over confidence scoring**
  Missing information is surfaced explicitly; no confidence metrics are produced.
- **Open inspection over hidden heuristics**
  All rules are inspectable, modifiable, and versioned.

These principles constrain capability by design and form the basis of DDRP's trust posture.

3. Protocol Overview

At a high level, DDRP v0.1 follows this pipeline:

```
Input Document
        ↓
Canonicalization (deterministic text normalization)
        ↓
Lexical Operator Detection
        ↓
Operator Classification
        ↓
Obligation Instantiation
        ↓
Field Presence / Absence Evaluation
        ↓
Status Assignment
```

The protocol operates entirely on canonicalized text and produces structured JSON outputs suitable for audit, storage, or downstream human review.

4. Operator Taxonomy

DDRP v0.1 defines six operator classes, detected via deterministic pattern matching:

- **REQ (Requirement / Deontic)**
  Obligations or prohibitions (e.g., *must*, *shall*, *may not*).
- **DEF (Definition / Binding)**
  Explicit term definitions (e.g., "X" means…).
- **CAUSE (Causality / Justification)**
  Explicit causal markers (e.g., *because*, *therefore*).
- **SCOPE (Applicability / Context Gating)**
  Jurisdictional, conditional, or contextual boundaries.
- **UNIV (Universals / Quantifiers)**
  Absolute or quantifying terms (e.g., *all*, *never*, *only*).
- **ANCHOR (External Reference / Authority)**
  References to external standards, policies, or documents.

Each operator is detected solely based on lexical patterns. No attempt is made to normalize meaning or infer unstated relationships.

5. Obligation Model

Detected operators are grouped into **obligations**, which represent structured claims made by the document.

Each obligation tracks the explicit presence or absence of the following fields:

- Who
- What
- When
- Where
- Why
- How

Crucially, DDRP does not infer missing fields. **Absence is treated as evidence.**

Obligation Status Values

- **SATISFIED:** All required fields are present.
- **OPEN:** One or more required fields are missing.
- **AMBIGUOUS:** Conflicting or underspecified operators are present.
- **CONTRADICTED:** Reserved for future versions; not resolved in v0.1.

An OPEN status does not indicate system uncertainty; it indicates **document incompleteness**.

6. Determinism, Execution Continuity, and Verification

DDRP v0.1 enforces determinism through:

- Fixed operator pattern registries
- Stable operator ordering (character offset, then pattern ID)
- Stateless, pure functions
- Deterministic canonicalization and hashing

Each run produces a **transaction record** that captures:

- Input hash and length
- Detection and obligation hashes
- Execution environment
- Run identifier
- Hash chaining for continuity

The transaction record explicitly documents execution continuity and **does not assert compliance, correctness, or legal validity**.

Verification includes repeated execution tests producing byte-identical outputs across runs.

7. Hostile Audit Methodology

Unlike validation-oriented test suites, DDRP includes a hostile audit designed to falsify claims about system behavior.

Tests explicitly verify the **absence** of prohibited capabilities, including:

- Risk scoring
- Compliance determination
- Recommendations
- Intent interpretation
- Confidence estimation
- Semantic normalization

Failure to detect these absences constitutes a critical protocol violation.

8. Structural Integrity Characterization

DDRP v0.1 may optionally classify the **structural integrity** of a processed document based on observable, non-semantic criteria, such as:

- Operator density
- Obligation completeness ratios
- Scope stacking
- Canonicalization and hashing integrity

Indicative structural states include:

- **Normal**
- **Abnormal**
- **Flawed**
- **Corrupted**

These indicators describe **document structure only**.
They do not assess legality, correctness, compliance, or intent.

9. Scope Limitations

DDRP v0.1 intentionally does not:

- Interpret meaning
- Assess compliance
- Judge correctness
- Resolve contradictions
- Normalize semantic equivalents

DDRP may ingest PDFs and other non-plain-text formats **only via deterministic text extraction and canonicalization**. No layout, formatting, or visual inference is performed.

These limitations are documented to prevent misuse and overextension.

10. Intended Use and Organizational Scaling

DDRP is designed for:

- Individual document inspection
- Internal organizational review
- Independent third-party implementation

Scaling is achieved through:

- Independent pattern registry governance
- Versioned protocol specifications
- Decentralized implementation by separate teams

This avoids centralized authority over interpretation.

11. Relationship to Other Systems

DDRP does not replace interpretive or semantic systems. It precedes them.

Where other systems attempt to answer *what a document means*, DDRP answers *what the document explicitly commits to*.

This separation enables downstream interpretation to be challenged against a stable structural baseline.

12. Conclusion

The Deterministic Document Review Protocol v0.1 demonstrates that meaningful document analysis can occur without semantic inference or probabilistic reasoning. By making fewer claims—and making them reproducibly—DDRP preserves evidence about where interpretation becomes necessary rather than attempting to perform that interpretation itself.

This restraint is not a limitation; it is the protocol's defining feature.

Data and Software Availability

A reference implementation of DDRP v0.1, including hostile audit artifacts and deterministic transaction records, is publicly available at:

https://github.com/btisler-DS/ddrp