

ABSTRACT

Title of Thesis: A NEURO-SYMBOLIC FRAMEWORK FOR
ACCOUNTABILITY IN
PUBLIC-SECTOR AI

Allen Sunny
Master of Information Management, 2025

Thesis Directed by: Professor Ido Sivan-Sevilla
College of Information Studies

Automated eligibility systems increasingly determine access to essential public benefits, but the explanations they generate often fail to reflect the legal rules that authorize those decisions. This thesis develops a legally grounded explainability framework that links system generated decision justifications to the statutory constraints of CalFresh, California’s Supplemental Nutrition Assistance Program. The framework combines a structured ontology of eligibility requirements derived from the state’s Manual of Policies and Procedures (MPP), a rule extraction pipeline that expresses statutory logic in a verifiable formal representation, and a solver-based reasoning layer to evaluate whether the explanation aligns with governing law.

Case evaluations demonstrate the framework’s ability to detect legally inconsistent explanations, highlight violated eligibility rules, and support procedural accountability by making the basis of automated determinations traceable and contestable.

A NEURO-SYMBOLIC FRAMEWORK FOR ACCOUNTABILITY IN PUBLIC-SECTOR AI

by

Allen Sunny

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Master of Information Management
2025

Advisory Committee:

Professor Ido Sivan-Sevilla, Chair/Advisor
Professor Gabriel Kaptchuk
Professor Jessica Vitak

© Copyright by
Allen Sunny
2025

Acknowledgments

I am deeply grateful to my advisor, Professor Ido Sivan-Sevilla, for his guidance, trust, and commitment to my growth throughout this project. His mentorship has shaped both my work and my development as a researcher. I would also like to thank my committee members, Professor Jessica Vitak and Professor Gabriel Kaptechuk for their time, support, and thoughtful feedback during this process.

Special thanks to the faculty and staff of the University of Maryland's iSchool for creating a rigorous and supportive academic environment that has allowed me to pursue interdisciplinary research at the intersection of law, public policy, and artificial intelligence.

I am grateful to my friends and peers who shared this journey with me and offered encouragement on the days it mattered most. Finally, I owe the deepest thanks to my family, whose unwavering love and belief in me have carried me through every challenge. None of this would have been possible without them.

Table of Contents

Acknowledgments	ii
Table of Contents	iii
List of Tables	vi
List of Figures	vii
List of Abbreviations	viii
Chapter 1: Introduction	1
1.1 Motivation and Context	1
1.2 Problem Statement and Research Gap	2
1.3 Research Questions	2
1.4 Scope of Algorithmic Systems	3
1.5 Contributions	3
1.6 Thesis Structure	4
Chapter 2: Literature Review	5
2.1 SNAP and the Motivation for Welfare Automation	6
2.2 Current Failures and Controversies in deployed systems	8
2.3 Legal and Regulatory Foundations for Explainability	10
2.3.1 Data-Protection and Rights-Based Approaches	10
2.3.2 Administrative-Law and Due-Process Traditions	11
2.3.3 Sector-Specific and Emerging AI Regulation	11
2.3.4 The Erosion of Procedural Reasoning	13
2.3.5 Toward Legally Grounded Explainability	14
2.4 Limitations of Technical Explainability	15
2.4.1 Post-hoc Approaches	15
2.4.2 Intrinsic Approaches	16
2.4.3 Human-Centered XAI	17
2.4.4 Limitations for Legal Contexts	18
2.5 Legal Ontologies and Symbolic Reasoning	18
2.6 Neuro-Symbolic Methods	22
2.6.1 Neuro-Symbolic Verification in Practice	22
2.6.2 Legal Applications of Neuro-Symbolic Systems	23
2.7 Toward Legal Verification of Eligibility Explanations	23
2.8 Summary	24
Chapter 3: System Design and Methodology	26
3.1 Motivation for System Design	26

3.2	System Overview	27
3.3	Data Sources and Case Construction	29
3.3.1	Statutory Corpus: California MPP Division 63	30
3.3.2	Eligibility Case Dataset	31
3.4	Ontology Construction (TBox)	32
3.4.1	Ontology Design and Concept Hierarchy	33
3.4.2	Vocabulary Creation	35
3.4.3	Final TBox Knowledge Representation	42
3.5	Building the Assertion (ABox)	42
3.5.1	Assertion Vocabulary Extraction	43
3.5.2	Assertion Rule Creation	44
3.5.3	Final ABox Knowledge Representation	45
3.6	SMT Solver for Legal Consistency Checking	46
3.6.1	Formal Verification through SMT	46
3.6.2	Solving the Legal Reasoning Environment	48
3.7	Evaluation Design	49
3.7.1	Ontology Quality	49
3.7.2	Rule Robustness	49
3.7.3	Legal Consistency Evaluation	50
Chapter 4:	Results	51
4.1	Overview	51
4.2	Ontology Validation	52
4.3	Rule Robustness	53
4.3.1	Ground-Truth Rule Construction	54
4.3.2	Prompting Strategy Performance	54
4.4	Testing Dataset Creation	56
4.5	Legal Consistency Verification	57
4.6	Statutory Violation Trace Visualization	59
4.7	Individual Case Walkthrough	60
4.8	Conclusion	61
Chapter 5:	Discussion	62
5.1	Operationalizing HCXAI in the Public Sphere	62
5.2	Fairness Beyond Legal Compliance	63
5.3	Policy and Administrative Implications	65
Chapter 6:	Conclusion	67
6.1	Summary of Findings	67
6.2	Synthesis Across Research Questions	68
6.3	Limitations	69
6.3.1	Drift between output and system explanation	69
6.3.2	Reliance on statutory text without interpretive context	69
6.3.3	LLM sensitivity to legal complexity	70
6.3.4	Incomplete ontology coverage	70

6.3.5	Limited evaluation dataset	70
6.3.6	Fairness dimensions beyond legality	71
6.3.7	System interpretation rigidity	71
6.4	Future Work	71
6.4.1	Expanding normative coverage.	71
6.4.2	Broader administrative domains.	72
6.4.3	Integrating fairness and equity analysis.	72
6.4.4	User-centered contestation.	72
Appendix A:		73
A.1	Ontology Example	73
A.2	Rules Example	74
A.3	Case Files	75
Bibliography		76

List of Tables

3.1	TBox summary	42
3.2	Normalization of explanation variants into canonical ABox assertions.	45
4.1	Ground-truth statutory eligibility rules and SMT-executable logic.	54
4.2	Rule extraction success rates by model, prompting strategy, and statutory test case. Each cell reports successful formalizations out of 30 attempts; the final column shows the average success rate across the three cases.	55
4.3	Distribution of evaluation cases by eligibility category and legal outcome	57
4.4	Explanation verification performance across eligibility categories.	58

List of Figures

2.1	Example XAI Output: SHAP Explanation	16
3.1	System architecture of the statutory explainability framework.	29
3.2	A section of the statutory law as visualized in a knowledge graph	31
3.3	TBox Architecture diagram	33
3.4	Initial Ontology Structure	36
3.5	Ontology Expansion After Concept Integration	39
3.6	ABox creation pipeline.	43
3.7	SMT reasoning architecture for legal verification.	47
4.1	Ontology concept clusters by eligibility domain	52
4.2	Performance comparison across prompting strategies.	55
4.3	Solver trace visualization with red nodes identified as violated statutes	59
4.4	Example case walkthrough visualization	60
5.1	Visualization of satisfied and violated explanations in a sample eligibility case.	63
A.1	First page of example decision. Full document provided in project repository.	75

List of Abbreviations

ABox	Assertional Component (Case Facts)
AI Act	European Union Artificial Intelligence Act
GDPR	General Data Protection Regulation
HCXAI	Human-Centered Explainable Artificial Intelligence
JSON	JavaScript Object Notation
KG	Knowledge Graph
LLM	Large Language Model
MPP	Manual of Policies and Procedures
NOA	Notice of Action
NLP	Natural Language Processing
OECD	Organisation for Economic Co-operation and Development
OSTP	Office of Science and Technology Policy
SMT	Satisfiability Modulo Theories
SNAP	Supplemental Nutrition Assistance Program
TBox	Terminological Component (Ontology)
UMAP	Uniform Manifold Approximation and Projection
USDA	United States Department of Agriculture
XAI	Explainable Artificial Intelligence

Chapter 1: Introduction

1.1 Motivation and Context

Over the past few years, government service delivery has come under increasing strain. Economic instability, rising living costs, and global supply disruptions have driven more households to seek public assistance programs such as food benefits and childcare support [1]. At the same time, administrative budgets and staffing capacity have not kept pace with demand. Facing higher workloads with fewer resources, agencies have increasingly turned toward automation to maintain service delivery [2]. The rapid advancement of artificial intelligence and machine learning has accelerated this shift, encouraging the adoption of automated decision systems with the promise of efficiency and consistency [3].

However, integrating machine learning into public-benefit decision making introduces serious risks [4]. Advanced predictive systems are often opaque, making it difficult or impossible to understand how eligibility decisions are reached. Decisions that cannot be explained cannot be meaningfully challenged, restricting applicants' rights to contest errors that may deny them essential support. As automation expands without clear mechanisms for oversight, there is a growing concern that those most marginalized in society may be disproportionately harmed by incorrect or unreviewable determinations.

1.2 Problem Statement and Research Gap

Automated eligibility systems increasingly rely on predictive models whose internal logic does not correspond to the statutory rules that determine access to public benefits. Eligibility criteria are expressed in complex legal language, while model reasoning is framed in statistical terms. This disconnect means that explanations provided by AI systems may appear plausible yet fail to reference the required legal authority or may misstate the conditions under which benefits should be granted.

Without a shared legal structure, neither applicants nor administrators can determine whether an automated decision is justified under the law. Current explainability methods reveal correlations and model influence, but they cannot assess whether a decision reflects the correct statutory requirements or violates eligibility constraints. This gap highlights a fundamental accountability problem: explanations must be legally valid and not just interpretable, if automation is to preserve due-process rights in public-benefit administration.

1.3 Research Questions

Motivated by this gap, this thesis investigates the following research questions:

- **RQ1 — Representation:** How can statutory eligibility rules be structured into a computable form that preserves their legal semantics and hierarchical constraints?
- **RQ2 — Alignment:** How can model-generated explanations be translated into these legal structures so that the reasoning behind decisions is expressed in terms of the law?
- **RQ3 — Verification:** How can eligibility explanations be automatically tested for legal

compliance, detecting when they satisfy or violate statutory requirements?

1.4 Scope of Algorithmic Systems

This thesis focuses on automated decision systems used in public-benefit eligibility determination, where inputs, outputs, and governing rules are public by statute. These systems differ from commercial predictive models in that: (1) their operational logic is constrained by legally defined eligibility criteria [5], (2) agencies are obligated to provide explanations grounded in statutory authority [6], and (3) determinations are subject to administrative review and appeal [7]. The proposed framework therefore applies to contexts with formal legal mandates and observable decision rationales, rather than discretionary or proprietary systems.

1.5 Contributions

To address these questions, this thesis introduces a legally grounded explainability framework for automated eligibility decisions in public-benefit programs. This work provides the following contributions:

1. A **legal ontology** modeling statutory rules and constraints from the California Manual of Policies and Procedures (MPP), enabling a formal representation of eligibility law (RQ1).
2. A **semantic alignment method** that converts neural model explanations into structured rule assertions compatible with the ontology (RQ2).
3. A **formal reasoning workflow** using satisfiability-based verification to determine whether explanations are legally compliant or in violation (RQ3).

1.6 Thesis Structure

The remainder of this thesis details the design and evaluation of this approach. Chapter 2 reviews relevant work at the intersection of explainability, algorithmic governance, and computational law. Chapter 3 presents the system architecture, including legal knowledge representation, explanation alignment, and verification mechanisms. Chapter 4 evaluates system performance on CalFresh eligibility cases. Chapter 5 discusses broader implications for policy and human-centered oversight. Chapter 6 concludes by outlining limitations and future directions.

Chapter 2: Literature Review

The increasing automation of welfare and other public-benefit systems has generated parallel research trajectories in law, social science, and artificial intelligence, each grappling with the problem of how to make algorithmic decisions accountable. This chapter reviews those literatures to situate the problem of legally grounded explainability within broader debates on administrative automation and responsible AI. It traces how public agencies’ turn toward data-driven decision-making; reconfigures foundational legal obligations of due process and reason-giving, and how existing technical approaches to explainability only partially meet those obligations.

The review proceeds in eight sections. Section 2.1 traces the historical evolution of welfare automation and the administrative pressures that drove the adoption of eligibility technology. Section 2.2 examines the harms observed in deployed systems, highlighting how opacity and rigid rule execution undermine due-process protections. Section 2.3 outlines the legal foundations of explainability, emphasizing reason-giving as a core requirement of administrative legitimacy. Section 2.4 reviews explainability approaches in artificial intelligence and analyzes why they fall short in legally constrained decision-making. Section 2.5 introduces legal ontologies and symbolic reasoning as methods for representing statutory structures in machine-interpretable form. Section 2.6 surveys neuro-symbolic approaches that integrate language interpretation with constraint-based verification. Section 2.7 brings these strands together to identify the technical

and legal requirements for verifying eligibility explanations. Finally, Section 2.8 summarizes the research gap: although explainability tools and legal reasoning systems have advanced significantly, few frameworks ensure that automated justifications are not only interpretable but legally valid.

By mapping this interdisciplinary terrain, the chapter establishes the conceptual foundation for the framework proposed in later chapters, a neuro-symbolic system that operationalizes the law’s reason-giving mandate within computational architectures.

2.1 SNAP and the Motivation for Welfare Automation

The Supplemental Nutrition Assistance Program (SNAP) [8] is the largest food-assistance program in the United States, providing monthly benefits that enable low-income households to purchase groceries. Participation is based on meeting statutory eligibility criteria that reflect both financial need and household circumstances, including income, residency, citizenship, resources student status etc. As an entitlement program, benefits must be issued to every individual who qualifies; errors in eligibility determination therefore put applicants at risk of losing access to basic subsistence needs.

In California, SNAP operates under the name *CalFresh* [9]. Eligibility determinations are governed by the California Manual of Policies and Procedures (MPP) Division 63 [10], which translates federal and state statutory requirements into detailed administrative rules. When a decision is made, counties must issue a Notice of Action (NOA) [11] that explains the legal basis for an approval, denial, reduction, or termination of benefits. Because internal decision processes may be automated or otherwise opaque, the NOA is often the only publicly visible

justification for how eligibility rules were applied to a particular household. As a result, the NOA serves as the central accountability mechanism through which applicants can understand, challenge, or correct decisions that affect their access to food.

SNAP and state-administered programs like CalFresh operate under increasing strain [12]. Rising living costs, economic volatility, and staffing shortages have expanded caseloads while shrinking the resources available to process them efficiently [13]. To maintain service delivery at scale, welfare agencies have increasingly turned to automation, first through business rules and case-management software, and more recently through machine learning systems [14]. These tools promise efficiency and uniformity, but they also intensify longstanding challenges in welfare administration: when eligibility reasoning is encoded in technical infrastructure, the rules guiding determinations can become opaque to those directly affected [15, 16].

This shift does not represent a sudden rupture but an acceleration of a decades-long trajectory. Performance-management pressures and “audit culture” reframed social rights as data-management problems, prioritizing throughput and fraud detection over deliberative case-work [17]. Even before digitization, standardized paperwork and regulatory checklists had already created a proto-automated bureaucracy. Modern AI systems extend this history by further obscuring how rules, exceptions, and case facts are applied [18]. As automation becomes central to CalFresh delivery, ensuring that decision systems remain transparent and legally accountable is essential for protecting due- process rights.

These concerns are not isolated to CalFresh: across the welfare state, automated eligibility systems have produced improper denials, opaque decision-making, and substantial harm to beneficiaries when the legal basis for decisions is inaccurate or unclear [19].

2.2 Current Failures and Controversies in deployed systems

Despite promises of efficiency and accuracy, automated eligibility systems in welfare administration have repeatedly produced large-scale harms [20]. These are not isolated technical errors but structural failures that arise when bureaucratic discretion is replaced by rigid code. Once eligibility rules are converted into executable logic, they lose the interpretive flexibility that caseworkers once exercised, allowing minor data inconsistencies or procedural lapses to cascade into automatic denials.

The Michigan Integrated Data Automated System (MiDAS), for example, used rule-based pattern matching to detect unemployment fraud between 2013 and 2015. Operating with no human oversight, it falsely accused roughly 40,000 people of fraud and automatically imposed severe penalties, later ruled to violate due-process rights [21]. A similar pattern occurred in Indiana’s 2007 Welfare Modernization Project, where privatized eligibility software interpreted missing paperwork as “failure to cooperate,” triggering mass benefit terminations before the contract was cancelled [22]. Comparable dynamics have appeared in California’s CalWIN and CalSAWS systems, where county auditors and advocates have reported data mismatches, opaque eligibility rules, and automatically generated notices that omit the governing legal citation [23].

Across these cases, several mechanisms of failure recur. Opacity prevents both applicants and agencies from inspecting the logic of decision rules embedded in proprietary code. [24] Procedural drift occurs when software updates modify eligibility conditions without parallel legal review. [25] Administrative burden shifting transfers work to claimants, who must verify and correct data through digital portals. [26] Data fragility allows trivial inconsistencies to trigger denials because systems interpret uncertainty as non-compliance. [27] And vendor capture leaves

public agencies dependent on private contractors, weakening accountability and transparency. [28]

Beyond initial determinations, automation has profoundly altered the landscape of appeal and redress. [18] In traditional welfare administration, applicants could challenge an adverse decision by presenting new evidence, clarifying documentation, or engaging directly with caseworkers who understood the program’s logic. [29] Automated eligibility engines, however, often produce determinations without a transparent reasoning trace, leaving both claimants and front-line staff unable to identify the source of error. When an applicant appeals, administrators must reconstruct decisions from incomplete data logs or vendor-controlled audit trails, a process that converts legal review into forensic debugging. The opacity of proprietary code and the rigidity of rule-based engines mean that even successful appeals tend to address symptoms rather than systemic flaws [30] . In some jurisdictions, automated denials trigger appeal backlogs so large that relief arrives only after benefits have lapsed, effectively nullifying the right to timely redress. The very procedures intended to guarantee fairness thus become absorbed into the same computational architecture that produced the harm, illustrating how algorithmic administration compresses not only discretion but also contestability within bureaucratic systems.

Collectively, these controversies reveal how automation transforms the epistemic structure of welfare governance. Algorithmic systems optimize for throughput and standardization but erode the reason-giving practices that administrative law demands. The result is a widening gap between computational and legal rationality, between decisions that are efficiently produced and those that are procedurally justified.

2.3 Legal and Regulatory Foundations for Explainability

Explainability in automated decision-making is no longer only a design preference or ethical aspiration; it is increasingly codified as a legal obligation. Across jurisdictions, legislators and courts are beginning to treat algorithmic explanation as a procedural right that ensures accountability, transparency, and the possibility of redress. [31] Although these frameworks differ in scope and philosophy, they share a common premise: when state or corporate actors rely on automated reasoning, they must be able to justify those decisions in a manner intelligible to affected individuals and reviewable by oversight bodies.

2.3.1 Data-Protection and Rights-Based Approaches

The most influential articulation of a legal “right to explanation” originates in European data-protection law. Article 22 of the General Data Protection Regulation (GDPR) [31] establishes that individuals shall not be subject to decisions based solely on automated processing that significantly affect them, while Recital 71 [32] mandates that such processing be accompanied by “meaningful information about the logic involved.” Scholars have debated the strength of this provision whether it creates a substantive right to explanation or merely procedural transparency [33], [34] yet it nonetheless set the global benchmark for algorithmic accountability. National implementations, such as France’s Digital Republic Act (2016) [35], go further by requiring public authorities to disclose the data sources and parameters used in automated decisions. Outside Europe, Canada’s Directive on Automated Decision-Making (2020) [36] and Brazil’s LGPD (2020) [37] incorporate similar duties of transparency and contestability. Rights-based regimes thus frame explainability as an instrument of informational self-determination:

individuals must be able to understand and challenge the reasoning that affects their legal or economic standing.

2.3.2 Administrative-Law and Due-Process Traditions

In the United States, obligations of explanation emerge not from data-protection statutes but from administrative and constitutional law. The Administrative Procedure Act (APA § 706) [38] requires agencies to provide a “reasoned explanation” for their actions, a doctrine that courts enforce through the “arbitrary and capricious” standard of review. Landmark welfare cases such as [39] and [40] established that recipients of public benefits are entitled to notice and an opportunity to contest before deprivation. Scholars like [3] and [41] have extended these principles into the digital realm under the banner of technological due process, arguing that when algorithms replace human discretion, they must still furnish the procedural safeguards , explanations, hearings, and review that due process demands. Within this tradition, explainability is not a matter of user comprehension but a constitutional mechanism that preserves the rule of law within automated administration.

2.3.3 Sector-Specific and Emerging AI Regulation

Beyond these foundational doctrines, a new generation of AI-specific regulations seeks to institutionalize explainability through design and documentation mandates. The EU AI Act (2024) [42] classifies welfare, credit, and law-enforcement algorithms as “high-risk,” obliging providers to maintain technical documentation, traceability logs, and human-oversight procedures that make system logic auditable. The OECD AI Principles (2019) [43] and the UNESCO

Recommendation on the Ethics of AI (2021) [44] reinforce similar expectations of transparency, accountability, and contestability. In the United States, the proposed Algorithmic Accountability Act (2023) [45] and the White House Blueprint for an AI Bill of Rights (2022) [46] advance parallel requirements: notice to individuals, explanations of automated decisions, and the ability to opt for human review. These instruments shift the regulatory emphasis from voluntary corporate disclosure to mandatory procedural justification, making explainability a condition of lawful deployment.

The principle that public authorities must provide reasons for their decisions lies at the core of modern administrative governance and the rule of law. Legal theorists from Lon Fuller to Jerry Mashaw have long argued that procedural fairness, not merely the correctness of outcomes legitimizes bureaucratic power [47].

[48] described this as part of the “inner morality of law”: a lawful system must generate intelligible and contestable reasons if it is to remain non-arbitrary. [39] later extended this within the U.S. administrative state, emphasizing that legitimacy arises from reason-giving procedures that enable affected individuals to understand and challenge state action.

In administrative law, this obligation is operationalized through due process; The right to notice, an explanation, and an opportunity to contest. [39] established that welfare benefits cannot be terminated without prior notice and a fair hearing, confirming that government determinations are acts of legal judgment that must be articulated and reviewable.

Applied to contemporary welfare automation, these doctrines make explainability a procedural right rather than a design preference. When algorithmic systems determine eligibility, the duty to “give reasons” transfers from the human caseworker to the system’s architecture. [49] The state’s accountability thus depends on computational traceability: affected individuals must

be able to see which statutes and factual predicates governed their case and how those were applied.

2.3.4 The Erosion of Procedural Reasoning

As agencies replace human adjudication with digital infrastructure, the practical capacity to give reasons has eroded. What was once a dialog exchange between citizen and caseworker has become an impersonal data transaction [29]. Applicants now receive automated notices of approval or denial, but rarely the reasoning or legal provisions that produced those outcomes. The obligation to explain persists, yet its execution is displaced into software architectures never designed to fulfill it [50].

Scholars describe this transformation as a technological due-process failure. When automated systems replicate administrative functions without embedding the notice-and-hearing safeguards required by law, state action becomes procedurally illegible. [3] identifies the paradox: algorithms can implement formal logic precisely while remaining unaccountable because their inner operations are inaccessible to those they govern.

In welfare contexts, proprietary vendor contracts and fragmented data integrations deepen this opacity. Administrators themselves often cannot reconstruct why a decision was made, let alone communicate that reasoning to claimants [24]. Oversight mechanisms, appeals, judicial review, legislative inquiry presume an interpretable record of reasoning. Automated systems frequently lack such records, producing outcomes without a traceable path from facts to legal norms. As a result, the meaningful opportunity to be heard promised by [39] collapses into a purely procedural formality.

These failures reflect not only gaps in usability but a deeper structural issue: automated eligibility systems lack a computable representation of legal rules that would enable them to produce justified and reviewable determinations.

2.3.5 Toward Legally Grounded Explainability

Legal accountability requires more than transparency; it requires reasoned justification that maps outcomes to their normative sources. In administrative contexts, an explanation is adequate only if it demonstrates consistency with the statutes, regulations, or policy provisions that authorize it. Explainability must therefore evolve from a cognitive aid into a form of juridical traceability, the ability to show, in formal terms, that a system’s reasoning aligns with the law [51] [52].

This insight points toward a new class of technical solutions that integrate computational reasoning with legal logic. Rather than treating explainability as a human-factors problem, these methods conceive of it as a compliance mechanism. By aligning algorithmic outputs with explicit rule representations, a decision system can serve as a procedural interface that upholds due-process obligations [53].

Bridging this gap requires hybrid architectures that combine semantic interpretation with formal verification i.e. linking the flexibility of natural-language to the rigidity of logical constraint solving. Such approaches move explainability beyond narrative description toward computational justification.

In automated eligibility systems, explainability is not simply a matter of model transparency but a legal duty of justification. Determinations must articulate how specific statutory

requirements were applied to the facts of a case. Meeting this obligation in computational settings requires technical mechanisms capable of representing legal rules and verifying that decisions conform to them. The following sections examine technical methods of explainability and their various limitations.

2.4 Limitations of Technical Explainability

Over the past decade, the field of explainable artificial intelligence (XAI) has developed a wide range of methods designed to render machine learning models interpretable to humans. These techniques fall broadly into two categories: post-hoc explainability, which attempts to approximate the reasoning of an already trained model, and intrinsic explainability, which designs interpretability directly into the model architecture. Both paradigms share an epistemic goal that is to make black-box systems more intelligible. But they differ in their mechanisms and, more importantly, in their suitability for legally accountable decision-making.

2.4.1 Post-hoc Approaches

Post-hoc methods generate explanations after a model has made its prediction. Tools such as LIME [54] and SHAP [55] exemplify this approach. They construct local approximations around an instance to show which input features most influenced the outcome. Other variants, like counterfactual explanations [56], describe how small input changes could yield a different result, while influence functions and feature attribution scores trace contributions back to training data. These approaches are widely used because they are model-agnostic and computationally efficient. However, they explain how a model reached a result, not whether that

result accords with law or policy [57].

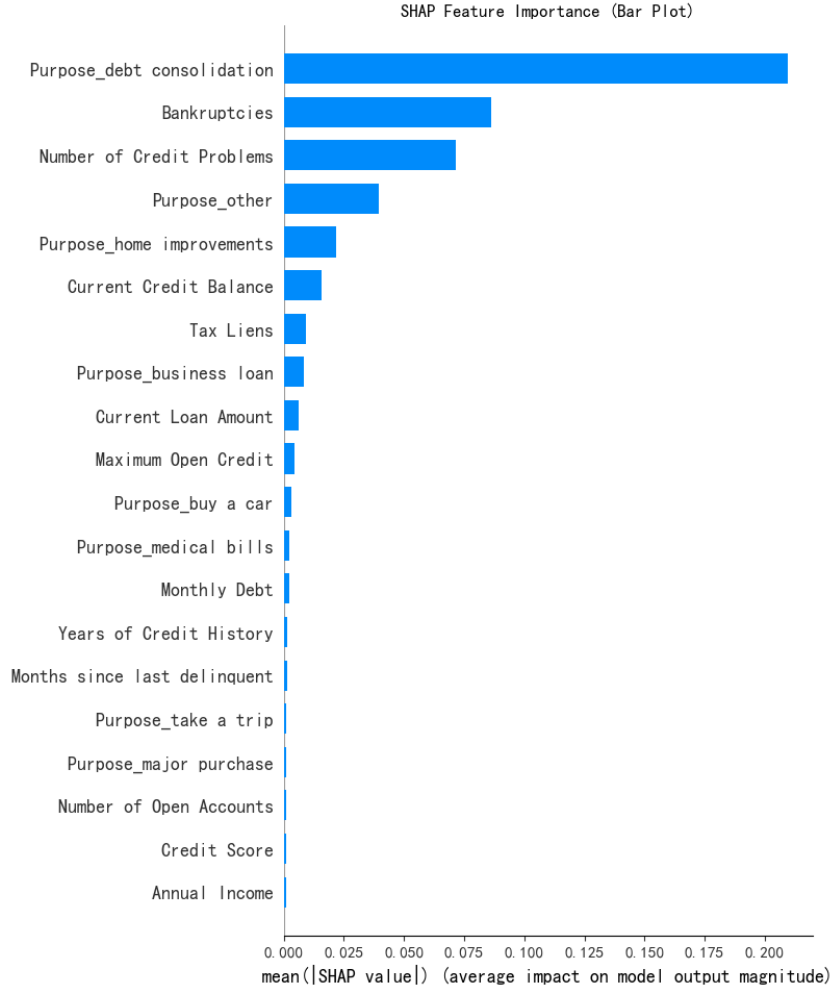


Figure 2.1: Example of a SHAP (SHapley Additive Explanations) plot illustrating how model features contribute to a specific eligibility decision. Each bar represents a feature’s marginal contribution to the model’s prediction relative to a baseline expectation. While such visualizations provide insight into which inputs the model deemed influential, they do not indicate whether the reasoning aligns with statutory requirements or references the correct legal conditions.

2.4.2 Intrinsic Approaches

Intrinsic or interpretable-by-design models, such as decision trees, rule lists, or monotonic generalized additive models, embed interpretability directly into their structure. Because their internal logic can be directly inspected, these models offer greater transparency and auditability

than deep neural networks. Yet, in high-stakes administrative contexts, they remain insufficient. Even a perfectly transparent model may still apply criteria that lack a legal basis. Interpretability, in this sense, is not equivalent to accountability. As [58] notes, a decision system can be “transparent but illegitimate” if it offers reasons unmoored from the normative frameworks that authorize them.

2.4.3 Human-Centered XAI

Recent work in human-centered explainable AI (HCXAI) [59] reframes explainability as a relational and contextual process rather than a purely technical property of models. Instead of treating explanations as outputs to be visualized, HCXAI examines how explanations function within human decision-making ecosystems, what users need to understand, contest, or trust a system. [60] emphasized that good explanations are social acts, shaped by conversational norms and cognitive expectations. Subsequent frameworks [61] extend this perspective to emphasize user goals, institutional settings, and power asymmetries. In public administration, this lens is crucial: the “user” is not merely a data scientist interpreting a model but an applicant, case-worker, or judge navigating rights and responsibilities. HCXAI thus shifts the design focus from explainability for insight to explainability for agency enabling affected parties to understand, challenge, and remedy algorithmic decisions. This human-centered orientation complements but does not replace legal grounding; rather, it provides the procedural scaffolding through which legally meaningful explanations can be communicated and acted upon.

While HCXAI emphasizes the communicative and social dimensions of explanation, it still assumes that the underlying decision logic is available to be interpreted. In legally regulated

domains, however, explainability must also be formally grounded i.e., anchored to statutes, regulations, and eligibility criteria that define what counts as a valid decision. To ground human-centered explainability in statutory authority, we require representational structures that can encode legal norms in machine-interpretable form.

2.4.4 Limitations for Legal Contexts

Across all paradigms, the prevailing assumption is that explainability is a cognitive aid—a means for users or auditors to understand model behavior. In contrast, public-benefit administration requires normative traceability: the ability to show that an outcome is consistent with governing law. Existing XAI methods cannot guarantee such alignment because they operate in a space of statistical correlation rather than formal obligation. They may approximate reasoning, but they do not encode the rules that confer legal validity. In effect, XAI techniques provide interpretive clarity, whereas administrative law demands procedural justification.

To move from interpretability to legality, decision systems must represent the concepts, thresholds, and dependencies that define eligibility in formal terms. This shift turns attention from post-hoc interpretive tools to knowledge representations capable of expressing the structure of law itself.

2.5 Legal Ontologies and Symbolic Reasoning

To satisfy the legal duty of reason-giving, automated decision systems must do more than interpret data; they must represent the law itself. Legal ontologies and symbolic reasoning frameworks respond directly to this requirement by encoding the concepts, thresholds, and hier-

archical dependencies that govern eligibility. Rather than explaining how a model behaves, they formalize why a decision is authorized, grounding computational outcomes in explicit normative structures.

In artificial intelligence and knowledge representation, an ontology is a formal specification of how entities within a domain are categorized and related to one another [62]. It articulates a shared conceptualization of reality, defining the classes (types of things), relations (how those things connect), and axioms (constraints that must hold true) that together describe a domain’s structure.

Gruber defines an ontology as “an explicit specification of a conceptualization” [63], positioning ontologies as a bridge between human semantic categories and the formal structures required for computational reasoning in legally regulated domains. Over time, several representational standards have emerged to encode such knowledge structures, ranging from the Resource Description Framework (RDF) [64], which expresses knowledge as subject–predicate–object triples, to the Web Ontology Language (OWL) [65], which builds on RDF to support richer logical semantics based on Description Logic. Prominent examples include the Gene Ontology [66] for biological processes, SUMO (Suggested Upper Merged Ontology) [67] for general reasoning across domains, and PROV-O [68] for provenance modeling on the semantic web. Within the legal domain, such frameworks have inspired specialized ontologies that formalize normative relations and procedural hierarchies, making the implicit structure of legal reasoning explicit. This formalization enables consistent interpretation across systems and supports reasoning tasks that depend on understanding both the semantics of legal terms and their logical interdependencies.

Research in AI and Law has long explored how legal reasoning might be expressed in

computational form. Early expert systems such as TAXMAN [69] and SHYSTER [70] demonstrated that statutory interpretation could be modeled through rule-based logic, yet they also exposed law’s resistance to full formalization. Legal norms are context-sensitive, exception-laden, and hierarchically organized; symbolic representations could capture syntactic rules but not the pragmatic reasoning that human adjudication requires.

From these limitations emerged the field of legal knowledge representation, which sought to separate the structure of legal concepts from the logic of their application. The development of ontologies which are formal, machine-readable vocabularies of legal entities and relations, was central to this shift. Frameworks such as LKIF-Core (Legal Knowledge Interchange Format) [71], LegalRuleML [72], and other OWL-based models introduced standardized methods for describing legal domains, defining hierarchical relationships among concepts, and encoding dependencies between rules. These ontological frameworks allowed legal knowledge to be represented with greater precision and interoperability, enabling reasoning engines to perform tasks such as compliance checking, conflict detection, and inference of legal consequences. In doing so, they transformed legal texts from static documents into structured knowledge bases that could be computationally queried and analyzed. This formalization provided a foundation for a new class of reasoning systems: once legal rules were expressed as explicit logical statements, they could be subjected to automated verification and consistency checking.

Satisfiability Modulo Theories (SMT) [73] frameworks extend this line of work by providing the computational mechanism to evaluate whether those formalized legal constraints can be jointly satisfied under given factual conditions. Whereas ontologies specify what entities and relationships exist in a legal domain, SMT reasoning tests whether the instantiated facts comply with those logical structures. By combining propositional logic with domain-specific theories,

such as arithmetic, temporal relations etc. SMT solvers can evaluate both qualitative and quantitative conditions (for example, determining whether a household’s income and expenses satisfy statutory thresholds, or whether overlapping obligations produce contradictions). In this sense, SMT reasoning operationalizes the normative content encoded in legal ontologies: it converts the declarative structure of law into a testable system of constraints, enabling computational models to verify rule compliance and detect violations.

Ontological and constraint-based approaches together have become foundational for tasks such as normative reasoning, legal information retrieval, and automated compliance checking. Process auditing systems like Regorous [74] integrate both layers, representing legal norms through formal ontologies and enforcing them through SMT-based constraint solving. Others employ logical reasoners to infer legal consequences or detect rule conflicts [75, 76]. Collectively, this body of work demonstrates that symbolic models can render the structure of law explicit and computationally tractable. Yet scholars continue to note their fragility [77]: they rely on exhaustive rule enumeration, require expert maintenance, and often remain disconnected from the natural-language texts and socio-legal contexts that give those rules meaning.

The extent to which legal rules can be exhaustively formalized remains contested. Statutory eligibility criteria often contain discretionary clauses, temporal dependencies, and exception handling that resist complete specification in symbolic form. As a result, current systems typically focus on the operational core of legal rules, leaving substantial interpretive work to human administrators or adjudicators [78].

Recent studies have therefore begun to explore hybrid approaches, linking ontological representations with natural-language processing or statistical learning to bridge the gap between textual norms and formal reasoning [79] and [80]. These efforts mark a broader turn toward

executable semantics in law, an attempt to make legal norms both interpretable and machine-actionable. The convergence of symbolic logic and semantic representation provides the conceptual foundation for emerging neuro-symbolic models that aim to integrate interpretability with normative verification.

2.6 Neuro-Symbolic Methods

As automated decision-making expands into public services, there is a growing need for systems that can both interpret natural-language reasoning and ensure consistency with formal constraints. Neuro-symbolic AI offers a pathway to bridge this gap by combining neural learning with symbolic reasoning [80]. Neural architectures extract structured meaning from free-text explanations, while symbolic solvers provide precise mechanisms to evaluate whether those interpretations align with rule-governed requirements.

2.6.1 Neuro-Symbolic Verification in Practice

Neuro-symbolic systems have demonstrated value in settings where behavioral errors carry significant risk and correctness must be formally established. Frameworks such as Logic Tensor Networks (LTNs) [81] and DeepProbLog [82] incorporate logical constraints directly into neural inference, ensuring that predictions obey domain-specific structure rather than unconstrained statistical associations.

More recent pipelines connect large language models to verification systems. For example, LLM-plus-solver architectures [83] where neural models extract candidate rules or explanations from natural language, and symbolic solvers check the satisfiability of those claims. These

workflows have been applied to visual reasoning tasks [84], scientific discovery [85], and automated theorem proving [86], demonstrating that neural interpretations can be systematically tested against external logical constraints rather than accepted at face value. Collectively, these approaches show that learned systems can be made subject to explicit rule auditing, offering stronger assurances than post-hoc interpretability alone.

2.6.2 Legal Applications of Neuro-Symbolic Systems

Legal domains have begun to explore neuro-symbolic models that align language interpretation with normative rule structures. Neural systems can identify legally operative concepts, clause boundaries, and exceptions from regulatory or adjudicatory text [87]. Symbolic solvers then verify whether extracted assertions satisfy formally expressed requirements [88]. Related approaches combine rule extraction with constraint validation—e.g., LLM + Z3 or LLM + Prolog pipelines [88, 89]—verifying whether the logic behind a textual explanation is complete and consistent. Across these efforts, neural networks interpret legal language while symbolic reasoning safeguards legal fidelity.

2.7 Toward Legal Verification of Eligibility Explanations

In government benefit programs, an explanation is not merely a transparency artifact: it is a legally operative justification determining whether a person receives essential support [3, 17]. Eligibility decisions must reference statutory authority and provide a basis for appeal. Ensuring that automated explanations meet these obligations requires more than interpretability—it requires verification that a justification conforms to governing law. Neuro-symbolic approaches

provide the core technical foundation for such oversight by enabling both the extraction of decision rationales and the formal evaluation of their legal correctness [89].

2.8 Summary

Across the bodies of scholarship reviewed in this chapter, a common trajectory emerges: automation has re-engineered the procedural foundations of welfare administration, while law and computer science have struggled to keep pace. Sociotechnical studies reveal how algorithmic systems optimize efficiency at the expense of transparency; legal theorists frame explanation as a right essential to due process; and technical research offers increasingly sophisticated methods for interpretability and verification. Yet these literatures remain fragmented. Legal frameworks articulate why explanations are required, while XAI and symbolic-reasoning frameworks explore how they might be produced, but few attempts reconcile the two within a single architecture.

Existing explainability tools clarify statistical correlations within model behavior but cannot demonstrate that a decision is legally grounded. Conversely, formal verification ensures logical consistency but often neglects the semantic commitments and contextual constraints encoded in statutory rules. As a result, current approaches cannot determine whether an automated eligibility decision is legally justified or identify when a violation of statutory requirements has occurred.

This gap between interpretability and legality defines the frontier of responsible automation. Bridging this divide requires a unified approach that represents statutory obligations in computable form and ensures that the reasoning behind automated decisions remains accountable to those obligations. The next chapter introduces a system architecture designed

to operationalize this principle by linking model explanations to the law they are intended to enforce.

Chapter 3: System Design and Methodology

3.1 Motivation for System Design

The architecture in this chapter directly operationalizes the research questions presented in Chapter 1. To determine whether eligibility explanations are legally grounded, the system must satisfy three requirements:

1. **Represent statutory authority in computable form (RQ1).** The system must capture eligibility law in a way that preserves legal semantics, constraints, and exceptions. This motivates the construction of a formal legal representation (TBox).
2. **Express case reasoning using the same legal structure (RQ2).** Agency explanations must be translated into rule assertions that reference the correct statutory conditions. This requires instantiating case facts and explanation-derived predicates within an assertional layer (ABox).
3. **Verify whether reasoning complies with the law (RQ3).** To support due-process review, the system must automatically identify when the asserted reasoning satisfies or violates eligibility requirements. This motivates the integration of a solver-based verification process.

These requirements define the layered design of the system: legal representation, explanation alignment, and automated verification. The following section provides a high-level overview of how these components interact to support legally accountable decision automation.

3.2 System Overview

At the core of the framework is a structured representation of legal reasoning based on two layers of knowledge. The TBox or “Terminological Box”, which encodes the statutory rules governing CalFresh eligibility, and the ABox or “Assertional Box”, which represents the factual and justificatory content of an individual case.

The TBox contains the ontology and formal rules derived from MPP Division 63, defining the conditions under which applicants are eligible for benefits. This can be viewed as a shared translation layer that maps the vocabulary and grammar of the legal system into representations that are operable for both human and machine reasoning. The TBox is grown dynamically as legal statutes are fed into the system. A comprehensive list of ontology and rules can be found in the [Appendix A](#).

The ABox instantiates these rules with concrete case-level data. It contains two categories of assertions: (1) factual attributes describing the applicant’s circumstances, such as household size, income values, or whether verification was provided; and (2) explanation-based assertions, representing the conditions the decision rationale claims are relevant to the eligibility outcome. These explanation-derived assertions are expressed in the same legal vocabulary defined in the TBox, allowing them to be evaluated against statutory requirements. In this way, the ABox captures both the state of the world and the reasoning offered to justify the decision. The

critical function of the ABox is therefore not only to represent case facts, but to make explicit the normative logic the explanation invokes, providing a structured basis for testing whether the stated reasoning aligns with the legal framework encoded in the TBox.

The TBox and ABox are jointly evaluated using a constraint-solving procedure. If the ABox assertions can satisfy all relevant TBox rules, the explanation is deemed legally coherent; if not, the solver identifies the specific statutory provisions that are violated. This architecture ensures that explanations are not only interpretable, but demonstrably aligned with the legal requirements that govern eligibility determinations. All of the data is modeled in a graphical representation commonly referred to as a knowledge graph [90]. In this context, a knowledge graph is simply a structured network of legally relevant concepts (nodes) and the relationships between them (edges). Representing rules and case facts in this form ensures that the same legal vocabulary is used throughout the pipeline maintaining semantic consistency in how eligibility conditions are invoked. The graph structure also enables intuitive visualization of which statutory conditions are satisfied or violated in a given case, supporting both human interpretability and procedural accountability.

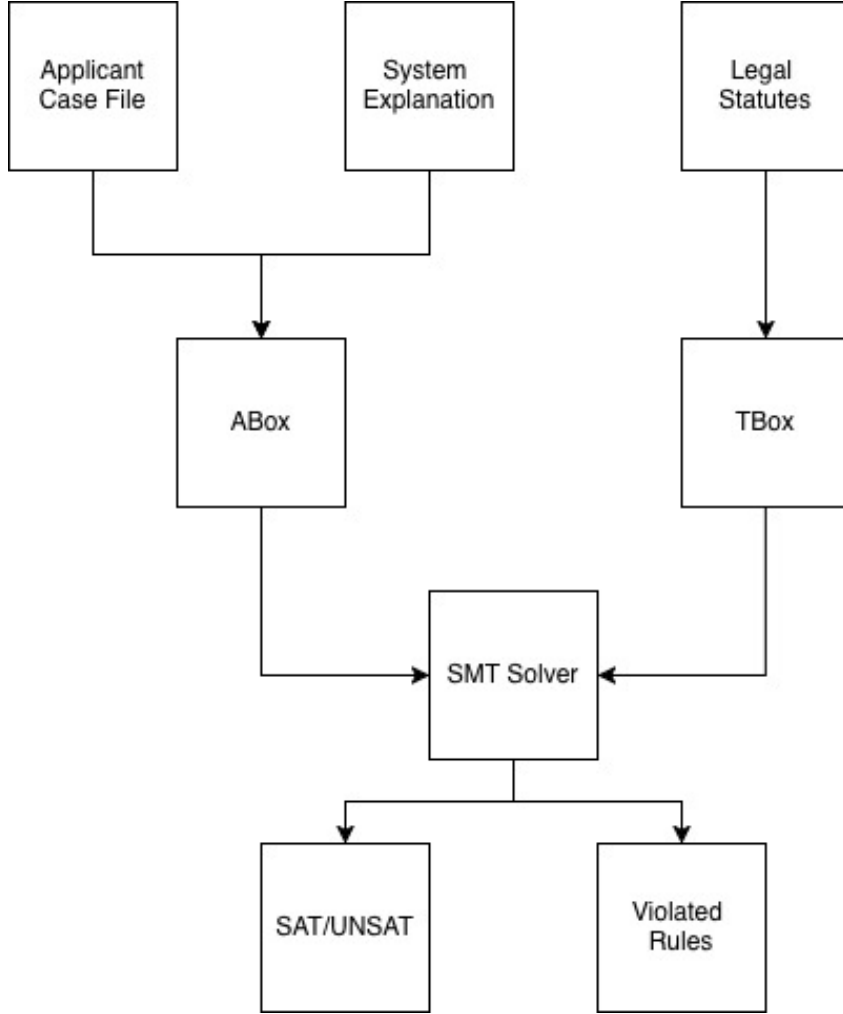


Figure 3.1: System architecture of the statutory explainability framework.

3.3 Data Sources and Case Construction

The neuro-symbolic framework developed in this thesis relies on two foundational sources of knowledge: the California Department of Social Services Manual of Policies and Procedures (MPP), Division 63, and a structured dataset of CalFresh eligibility cases. The MPP serves as the authoritative legal foundation from which statutory rules and ontology elements are derived, while the case dataset provides the empirical grounding required to evaluate whether the system can accurately assess the legal sufficiency of decision explanations. Together, these sources

establish a complete environment for testing legally grounded AI reasoning under realistic conditions, with the cases reflecting realistic sources of legal ambiguity and program administration, enabling evaluation of the system on decision types where accurate statutory justification is critical to due process.

3.3.1 Statutory Corpus: California MPP Division 63

Division 63 of the MPP codifies the administrative rules governing Supplemental Nutrition Assistance Program (CalFresh) eligibility across California. It defines the conditions under which households may qualify for food assistance, including requirements relating to household composition, income thresholds, allowable resources, residency in the administering county, citizenship or immigration status, reporting obligations, and eligibility for expedited services. These regulations are expressed in a hierarchical structure spanning sections, sub-sections, and clause-level normative obligations.

To enable machine reasoning, the official MPP publication was extracted in HTML format and segmented into JSON documents. Each record preserves the regulatory citation, raw and tokenized text for embedding, metadata such as the effective date of the provision, and explicit statutory references pointing to related sections. These cross-references are an important part of the legal logic: they establish dependencies between eligibility factors that must hold jointly for a decision to be lawful. Examples of the corpus structure can be found in the [Appendix A](#).

The structured statutory records were then imported into a legal knowledge graph using the graph database Neo4j, in which clauses function as nodes organized by the eligibility domain they regulate. Referenced provisions are connected as directed edges in the graph, preserving the

relational architecture of the law as written. This representation enables retrieval of conceptually related rules during reasoning, supports traceability between logical rules and their authoritative statutory sources, and maintains legal structural integrity when evaluating consistency. This statutory knowledge graph forms the foundation for constructing the terminology and rule set of the TBox, grounding every ontology concept and solver rule in a verifiable location within the MPP.

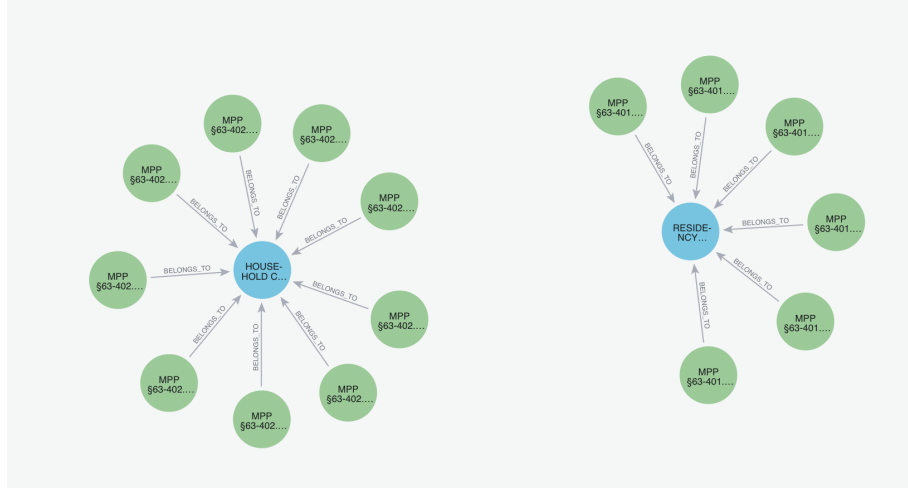


Figure 3.2: A section of the statutory law as visualized in a knowledge graph

3.3.2 Eligibility Case Dataset

To evaluate the legal-reasoning capabilities of the framework, a dataset of forty-three CalFresh eligibility cases was assembled directly from publicly accessible administrative hearing documents and agency-provided examples of eligibility decision explanations. These materials already omit personal identifiers, and no sensitive household information is included; thus, no additional anonymization was required. Each case was re-encoded into a structured JSON format that captures only factual variables relevant to eligibility, for example, residence county, income values, verification status, and categorical eligibility flags as well as the explanation provided for

the decision outcome. This dataset therefore provides a realistic testbed for assessing whether the framework can identify the correct areas of law implicated by a decision explanation and determine whether that explanation is logically consistent with the governing requirements of CalFresh eligibility. An example of the case file PDF can be found in the Appendix.

3.4 Ontology Construction (TBox)

The TBox, or terminological component, of the system captures the abstract schema of legal categories, relationships, and constraints that define eligibility under the California Manual of Policies and Procedures (MPP). Its purpose is to translate unstructured statutory language into a structured, machine-readable vocabulary that can later be instantiated with factual data in the ABox. To bridge system-level computation with human legal interpretation, the TBox is implemented as a legal ontology that preserves the meaning and hierarchy of statutory concepts while enabling their formal use in automated reasoning. This section describes how the ontology was designed, expanded, and embedded for downstream evaluation and verification.

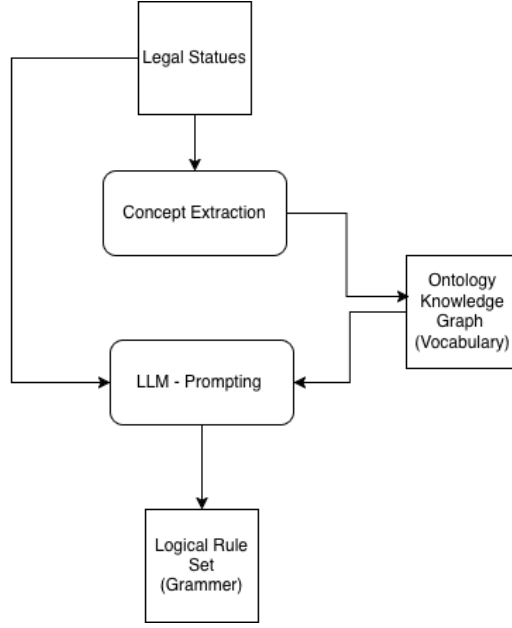


Figure 3.3: TBox Architecture diagram

3.4.1 Ontology Design and Concept Hierarchy

The hierarchy was derived directly from the legal organization of MPP Division 63. The MPP structures eligibility rules into major regulatory domains (e.g., income, residency, resources), each of which governs a legally distinct basis for eligibility. These top-level divisions were adopted as parent classes in the ontology to preserve the statutory separation of authority. This design decision follows the modeling conventions of LKIF-Core [71], where legally operative concepts are encoded as classes reflecting distinct normative roles in decision-making.

Within each domain, subordinate concepts were identified by extracting defined eligibility conditions, named variables, and verification requirements from statutory clauses. Concepts were added only when they represented a legally operative element i.e a factor that can independently determine an eligibility outcome under CalFresh regulations. This operational emphasis is consistent with LKIF’s distinction between normative concepts (conditions that affect legal

status) and descriptive facts (attributes without normative force).

Beyond organizing eligibility categories, the ontology also models the legally meaningful relationships between them. Each eligibility domain imposes conditions that reference or constrain attributes defined in other domains. For example, `IncomeEligibility` depends on `HouseholdComposition` to determine the household-specific income threshold, while `ResidencyRequirement` and `CitizenshipStatus` jointly define whether participation is permitted within a California county. These cross-domain dependencies are represented as object properties that capture how administrative decisions draw from multiple sources of statutory authority, again reflecting LKIF’s approach to encoding normative interactions rather than treating rules as isolated checks.

The ontology uses a two-level hierarchy consistent with LKIF’s intent to model only normatively relevant distinctions. The top level encodes legally independent eligibility domains; the second represents the concrete conditions required within each domain. No deeper subclassing was introduced, as the statutory structure does not assign normative weight to additional internal subdivisions (e.g., distinguishing between income verification and income limit provisions). This keeps the ontology aligned with how county eligibility workers reason: identifying the governing eligibility basis, then determining whether its required conditions are satisfied. The resulting structure provides a legally faithful conceptual foundation for the automated reasoning performed in later stages of the system.

3.4.2 Vocabulary Creation

3.4.2.1 Define Seed Concepts

The process is started by manually defining some of the key concepts for the ontology, concepts like IncomeEligibility, ResidencyRequirements , MedicalExpenses NetIncome etc. These are derived from the top subsections of the manual just to give a starting point for the vocabulary building process. The initial relationships between the sections was also derived.

The starting ontology JSON format is:

```
{
  "Root": {
    "subclasses": [
      "IncomeEligibility",
      "ResidencyRequirement",
      "CitizenshipStatus",
      "HouseholdComposition",
      "ResourceEligibility",
      "WorkRequirement"
    ]
  },
  "IncomeEligibility": {
    "attributes": [
      "GrossIncome",
      "NetIncome",
      "IncomeThreshold",
      "AllowableDeductions"
    ]
  }
}
```

This relationship is fed into the neo4j graph database and can also be viewed as a knowledge graph:

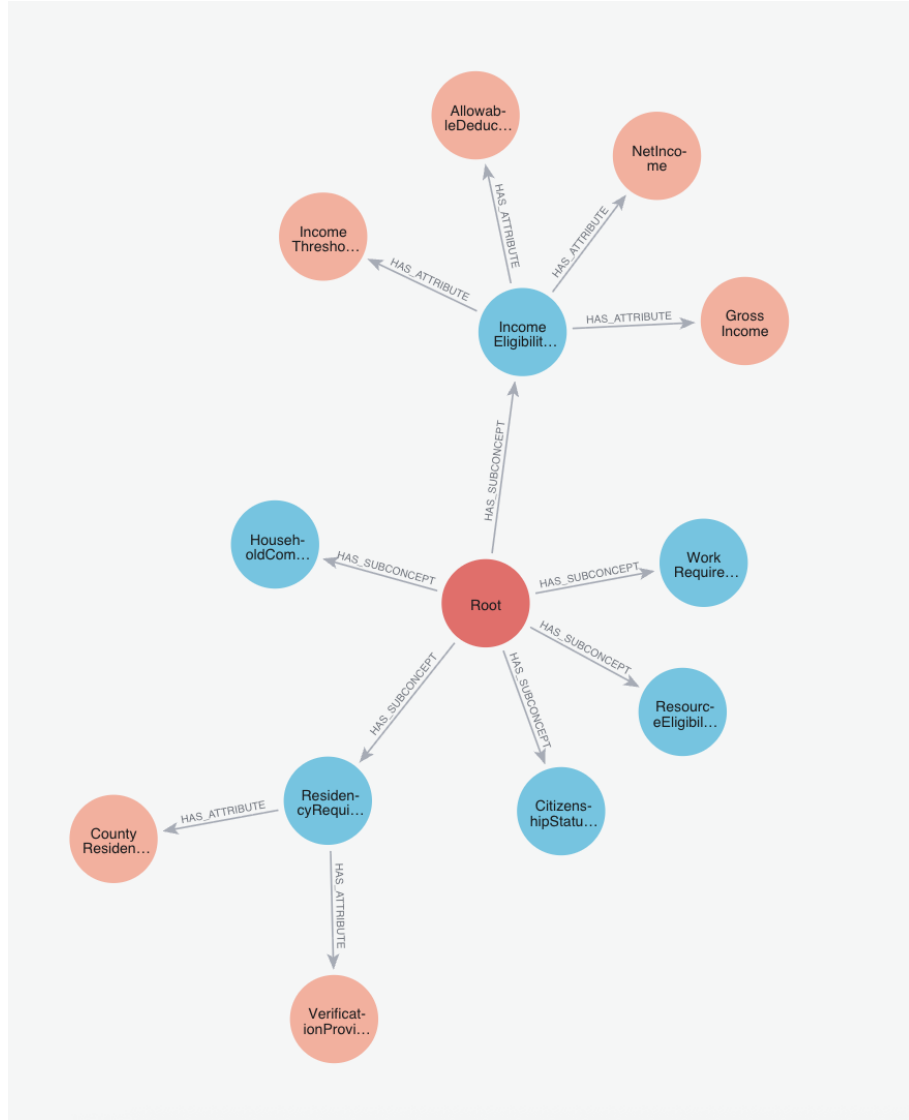


Figure 3.4: Initial Ontology Structure

3.4.2.2 Vocabulary Extraction from Statutory Law

Each statutory subsection in MPP Division 63 is processed sequentially to derive the ontology vocabulary used in the TBox. The goal is to identify the core legal concepts that will later be referenced by ABox assertions and solver rules.

This can be done in a two step process:

1. **Clause division:** Statutory text is segmented into minimal eligibility-relevant clauses.
2. **Concept extraction:** Each clause is processed using noun-phrase extraction to isolate legally operative terms. Entities, conditions, and eligibility predicates referenced in the statute. Extracted spans are normalized into ontology compatible labels through lemmatization, stopword filtering, and conversion to canonical predicate formats. Citation references to the originating statutory clause are preserved to ensure traceability during later verification.

Example:

MPP §63-401.1: *“A household shall be considered a resident of a county when it is living there and applies for benefits in that county.”*

Segmented clauses and extracted concepts result in:

```
{  
  "MPP 63-401.1": [  
    "Residency_HouseholdLocation",  
    "Residency_ApplicationCounty",  
    "Residency_County"  
  ]  
}
```

3.4.2.3 Integrating Extracted Concepts into the Ontology

Once candidate concepts are extracted, they are embedded using the **e5-large-v2** embedding model and compared against existing ontology terms using cosine similarity. A similarity threshold of > 0.85 is applied to determine whether a newly extracted term represents the same underlying legal concept already captured in the ontology. This prevents the creation of duplicate or semantically redundant nodes (e.g., “household residency” vs. “resident in the county”),

ensuring that each eligibility factor appears only once and maintains a consistent representation across rules and cases. Concepts below the similarity threshold are added as distinct nodes, allowing the ontology to expand.

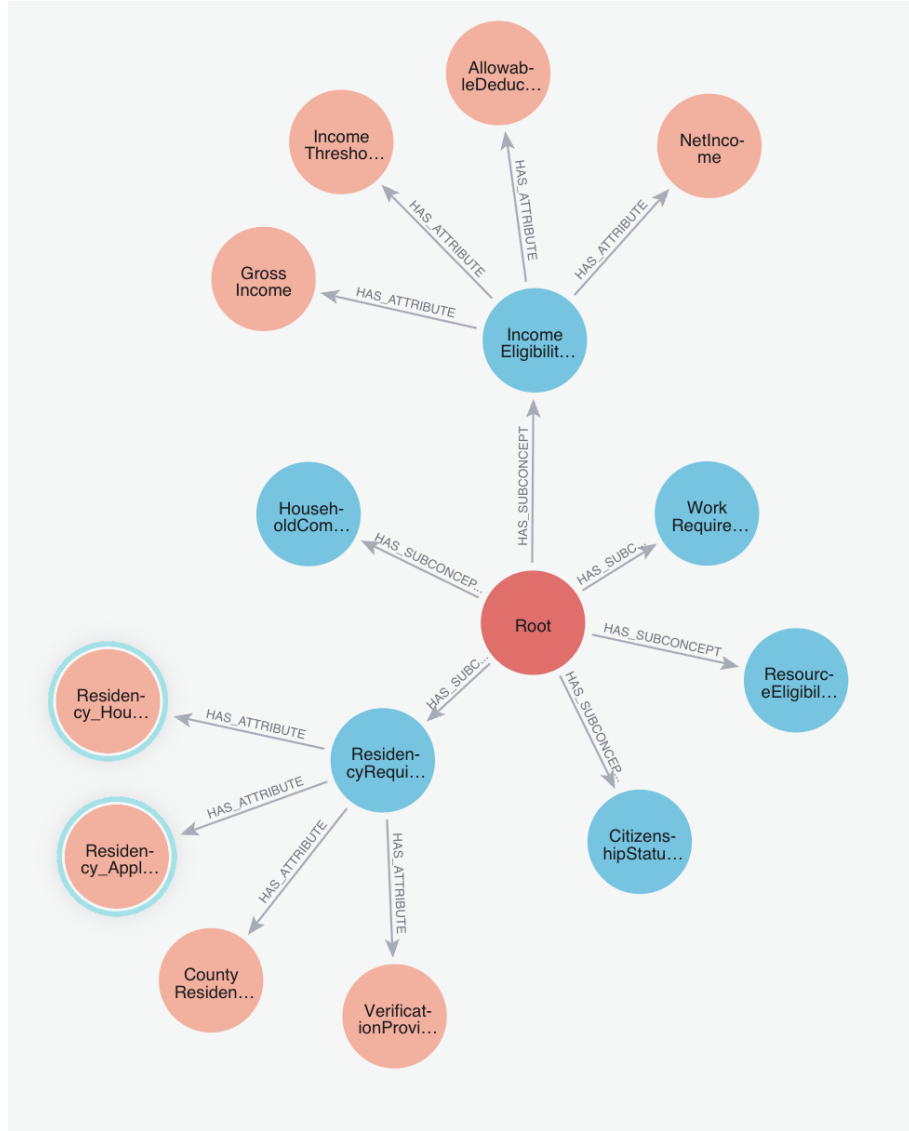


Figure 3.5: Ontology Expansion After Concept Integration

3.4.2.4 Rule Formalization (Creating the Grammar)

After the ontology vocabulary is established, statutory clauses are converted into formal rules that express legally operative logic in a machine-interpretable form. This process defines the “grammar” of eligibility verification. Rule creation consists of two sequential steps:

1. **Ontology Retrieval:** For each statutory clause, the system retrieves relevant ontology

concepts from the graph representation:

```
{
  "MPP 63-401.1": [
    "Residency_HouseholdLocation",
    "Residency_ApplicationCounty"
  ]
}
```

2. **Logical Translation:** Retrieved ontology terms form the vocabulary for a first-order logic rule. A large language model (GPT-o1) encodes the statutory semantics into a Z3-compatible state.

Example statutory clause: *“A household shall be considered a resident of a county when it is living there and applies for benefits in that county.”* (MPP §63-401.1)

The resulting solver rule reflects residency as a conjunctive requirement:

```
{
  "hasLogic": "Implies(And(Residency_HouseholdLocation,
                           Residency_ApplicationCounty),
               Applicant_Eligible)"
}
```

This representation preserves statutory intent while making residency determinations formally verifiable. Each rule maintains citation metadata linking the constraint to its authoritative legal source, enabling traceability during explanation evaluation.

The following Python prompt template constrains the model output:

```
prompt_template = """
You are a legal reasoning assistant that converts explanatory
clauses into formal logic rules compatible with the Z3 SMT solver.

Use only the ontology variables provided below.
Represent the eligibility conclusion as Applicant_Eligible.

Syntax Requirements:
- Express each rule as a single logical implication
- Use first-order logic operators: Implies, And, Or, Not, Equals
- Use only ontology variable names exactly as listed
- Output only JSON with the field: "hasLogic"
- Do not include natural language explanations

Ontology Concepts:
{ontology_concepts}

Clause:
"{explanation_clause}"

Output format:
{
  "hasLogic": "<first-order logical implication>"
}
"""
```

Model-generated rule:

```
{
  "hasLogic":
    "Implies(
      And(
        Residency_HouseholdLocation,
        Residency_ApplicationCounty
      ),
      Applicant_Eligible
    )"
}
```

The resulting rules are stored as a collection of JSONs.

3.4.3 Final TBox Knowledge Representation

The resulting ontology and rule set constitute the TBox. Examples shown below:

MPP Clause	Ontology Concepts	Formal Logical Rule
§63-401.1	Residency_HouseholdLocation, Residency_ApplicationCounty	Implies(And(Residency_HouseholdLocation, Residency_ApplicationCounty), Applicant_Eligible)
§63-502.36	GrossIncome, IncomeThreshold	Implies(GrossIncome \geq IncomeThreshold, Not(Applicant_Eligible))
§63-405.1	CitizenStatus, VerificationProvided	Implies(And(Not(CitizenStatus), Not(VerificationProvided)), Not(Applicant_Eligible))
§63-406	StudentFlag, MeetsStudentExemption	Implies(And(StudentFlag, Not(MeetsStudentExemption)), Not(Applicant_Eligible))
§63-501.3	HouseholdResources, ResourceThreshold	Implies(HouseholdResources \geq ResourceThreshold, Not(Applicant_Eligible))

Table 3.1: TBox summary

3.5 Building the Assertion (ABox)

The ABox (assertion box) represents the factual layer of the framework, the level at which concrete case data and textual explanations are instantiated and tested against the legal norms encoded in the TBox. Whereas the TBox specifies the formal structure of law, the ABox captures particular states of the world: applicant information, decision rationales, and derived factual assertions. Together, these layers allow the system to evaluate whether a model-generated or human-authored explanation is logically consistent with the statutory conditions that govern

benefit eligibility.

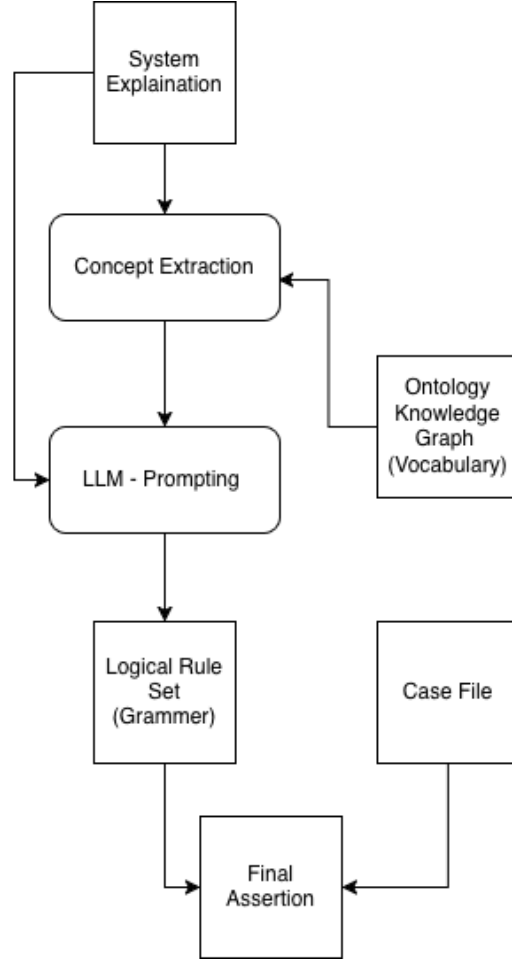


Figure 3.6: ABox creation pipeline.

3.5.1 Assertion Vocabulary Extraction

Assertion construction begins by grounding the free-text explanation in the same vocabulary that underlies the TBox. In the same way that statutory clauses were decomposed and mapped to ontology concepts, each explanation is first segmented into clauses and then aligned with ontology predicates.

1. **Clause segmentation.** The explanation is divided into minimal assertive units.

Example: “Your income was too high and you did not provide proof of residency.”

Distinct Clauses:

- “Your income was too high.”
- “You did not provide proof of residency.”

2. **Concept matching.** Each clause is mapped to ontology predicates using embedding-based similarity.

Mappings:

- Clause 1 \rightarrow GrossIncome, IncomeThreshold
- Clause 2 \rightarrow Residency_HouseholdLocation, VerificationProvided

These normalized assertions become explanation-linked ABox predicates, enabling the solver to test whether the stated rationale is legally sufficient for the outcome reached.

3.5.2 Assertion Rule Creation

Once the assertion vocabulary has been identified, each clause of the explanation is translated into a case-specific logical statement. This step mirrors the statutory rule-formalization process described in the TBox section, but focuses only on the concepts actually referenced in the explanation.

Prompt Abox Template:

```
prompt_template = """
You are a legal reasoning assistant that converts explanatory
clauses into formal logic rules compatible with the Z3 SMT solver.

Use only the ontology variables provided below.
```

Represent the eligibility conclusion as `Applicant_Eligible`.

Syntax Requirements:

- Express each rule as a single logical implication
 - Use first-order logic: `Implies`, `And`, `Or`, `Not`, `Equals`
 - Output only JSON with `"hasLogic"`
 - No natural language explanations
- """

Generated output example:

```
{
  "hasLogic": [
    "Implies(GrossIncome > IncomeThreshold, Not(Applicant_Eligible))",
    "Implies(Not(ResidencyVerificationProvided), Not(Applicant_Eligible))"
  ]
}
```

These statements constitute the ABox assertion set for the case.

3.5.3 Final ABox Knowledge Representation

The conversion process normalizes linguistic variability into a canonical solver-compatible format, ensuring that equivalent explanations map to identical rules.

Table 3.2: Normalization of explanation variants into canonical ABox assertions.

Explanation Variant	Canonical Logical Rule
"You applied in a different county than where you live."	<code>ResidenceCounty(Applicant) = ApplicationCounty(Applicant)</code>
"Eligibility denied — jurisdiction mismatch."	<code>ResidenceCounty(Applicant) = ApplicationCounty(Applicant)</code>
"You must live in the county you file in."	<code>ResidenceCounty(Applicant) = ApplicationCounty(Applicant)</code>
"Address on file belongs to another county."	<code>ResidenceCounty(Applicant) = ApplicationCounty(Applicant)</code>
"Applicant's household is located outside this county."	<code>ResidenceCounty(Applicant) = ApplicationCounty(Applicant)</code>

3.6 SMT Solver for Legal Consistency Checking

3.6.1 Formal Verification through SMT

The final stage evaluates whether the explanation-derived assertions remain legally consistent when combined with case facts and statutory constraints. A Z3 SMT solver determines whether there exists a logically coherent assignment of values under which all statements can be true simultaneously.

When the solver returns **UNSAT**, it also provides a minimal unsatisfiable core identifying the specific statutory constraints that the assertions violate. These violated rules are surfaced directly to the user or reviewer, enabling precise legal contestation of the decision’s justification rather than a generic error signal. In this way, the solver functions as a procedural accountability mechanism.

The system retrieves only those statutory rules whose vocabulary overlaps with the explanation-derived assertions, forming a targeted legal reasoning environment. Rather than evaluating the full regulatory corpus, the solver focuses exclusively on the legal provisions that the explanation claims are relevant to the decision. This ensures that the verification step tests the legal sufficiency of the stated rationale, not its consistency with unrelated eligibility criteria.

Example: If an explanation cites excessive income and missing residency verification, the system activates only the rules governing:

- Gross income thresholds (e.g., MPP §63–301.1)
- County residency and verification requirements (e.g., MPP §63–401.1)

This selective retrieval prevents unnecessary constraint expansion and directs formal ver-

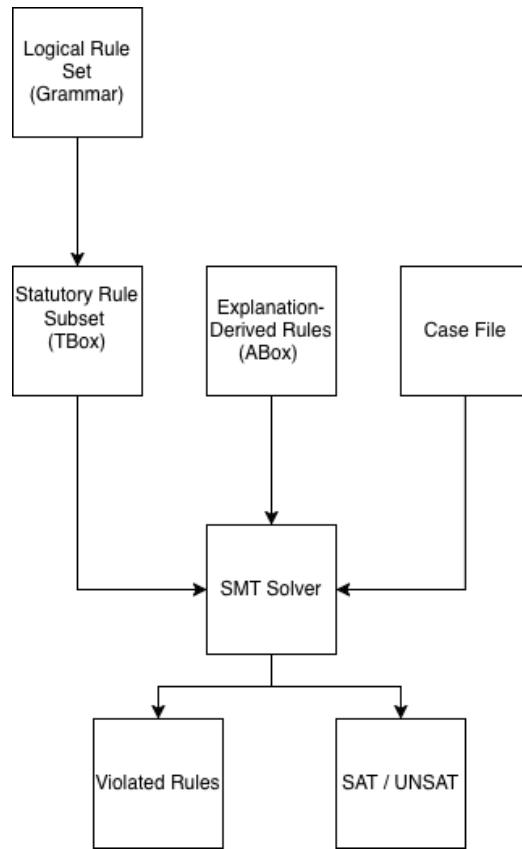


Figure 3.7: SMT reasoning architecture for legal verification.

ification to the precise statutory grounds invoked in the explanation.

3.6.2 Solving the Legal Reasoning Environment

To demonstrate the solver workflow, consider a case where the agency explanation states:

“Your income is too high and you did not provide proof of residency.”

From this explanation, the system extracts:

Ontology concepts: GrossIncome, IncomeThreshold, ResidencyVerificationProvided, Applicant_Eligible

Retrieved statutory constraints:

- MPP §63–301.1 — Gross income must fall below the household-specific limit
- MPP §63–401.1 — Residency must be verified in the administering county

These rules form the constrained legal environment for reasoning.

Scenario A: Facts support the explanation

If case data confirm both excessive income and missing residency verification, the solver returns:

SAT — explanation is legally valid

The reasoning trace identifies the specific satisfied rules, supporting due-process notice obligations.

Scenario B: Explanation contradicts the law

If income is below the threshold but residency verification is missing, the solver returns:

UNSAT — violated constraint: MPP §63–301.1

The solver surfaces the exact statutory rule that fails, enabling targeted correction or contestation of the decision rationale.

3.7 Evaluation Design

The performance of the proposed framework is evaluated along three dimensions that correspond directly to the major components of the verification pipeline. Each dimension isolates a distinct source of potential failure: semantic structure, rule formalization, and legal consistency checking. This design ensures that the evaluation assesses not only whether the system produces correct outcomes, but whether it does so for reasons that are legally grounded.

3.7.1 Ontology Quality

The first evaluation examines whether the semantic structure of the ontology accurately reflects the legal organization of MPP Division 63. Concepts belonging to different eligibility domains should form meaningfully distinct groups, enabling accurate rule retrieval during explanation verification. This is assessed through embedding-based cluster separation, inter- and intra-cluster distance measures, and visual inspection of concept clusters.

3.7.2 Rule Robustness

The second evaluation tests whether explanation-derived rules can be consistently translated into canonical solver-compatible logic, even when explanation texts vary. Multiple prompting configurations and language models are used to assess whether logical statements converge

to legally equivalent forms across cases and decision conditions. Failures in convergence indicate ambiguity or brittleness in the translation process.

3.7.3 Legal Consistency Evaluation

The final evaluation assesses whether the solver correctly determines whether an explanation is legally coherent with governing statutory constraints. The solver’s SAT/UNSAT outcomes and violated-rule identifications are compared against ground-truth expectations for each case to determine correctness across eligibility categories.

The results of these evaluations are presented in Chapter [4](#), organized according to the same three dimensions.

Chapter 4: Results

4.1 Overview

This chapter presents the empirical findings from evaluating the proposed framework. Because the internal mechanics of CalFresh eligibility determination are not publicly observable, we can equate the current black box administrative state to a hypothetical black box AI system. In the current process, if a claimant is denied benefits, the agency will send out a Notice of Action (NOA) explaining the agency’s decision and the statutory authority under which it has taken that action. We evaluate only the legal validity of the explanations provided in the Notice of Action (NOA).

Results are organized around three components of the system’s verification pipeline:

- **Ontology quality:** whether the semantic organization of encoded legal rules reflects statutory structure and supports relevant rule retrieval.
- **Rule robustness:** whether explanation-derived rules can be consistently expressed in formal, verifiable logic.
- **Legal consistency evaluation:** whether the SMT solver correctly detects alignment or conflict between the decision asserted in a case and the governing statutory constraints.

These results collectively assess the system’s ability to identify legally inconsistent expla-

nations, surface violated rules, and support procedural accountability.

4.2 Ontology Validation

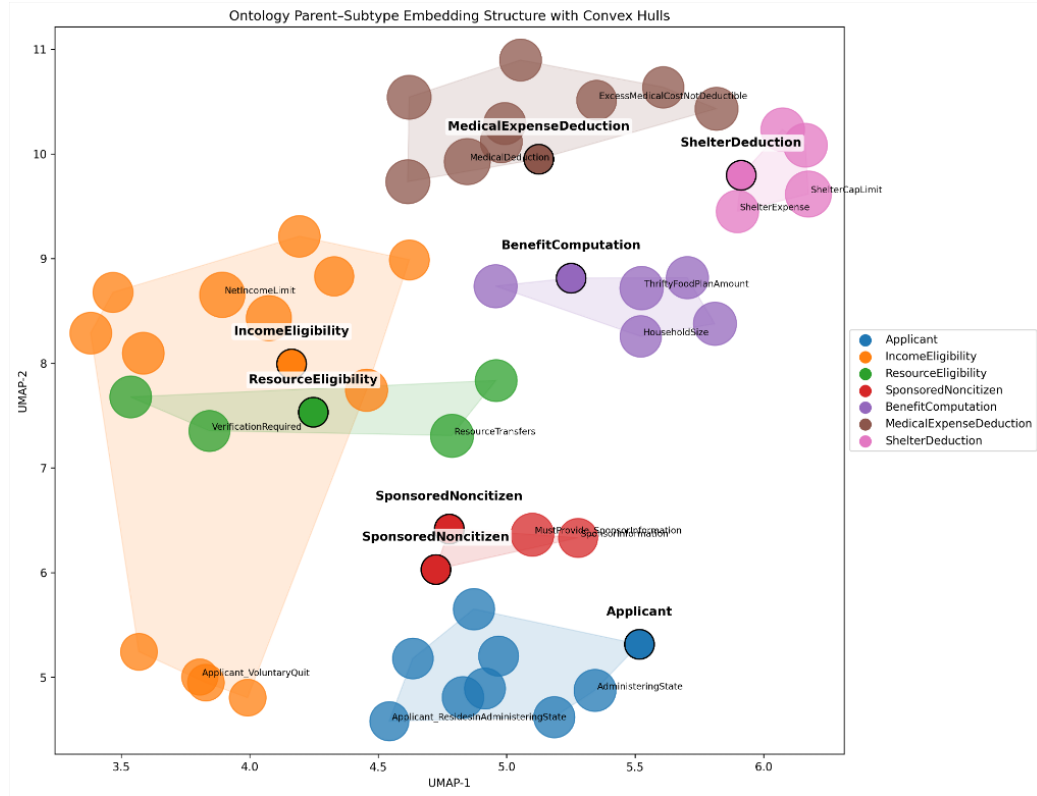


Figure 4.1: Ontology concept clusters by eligibility domain

A key design requirement of the ontology is that legally distinct eligibility concepts remain divergent. CalFresh law contains separate criteria for income, residency, resources, citizenship, and student status. These categories should not overlap semantically: if the representation space blends residency concepts into income concepts, the system could retrieve the wrong statutory rules during explanation verification.

Concepts were embedded and projected into a two-dimensional UMAP semantic space. Concepts were colored by legal domain assignment, and convex hulls were drawn around each

group for boundary clarity.

The resulting projection shows distinct and visually separable clusters. Distance analysis further supports this observation:

- Average intra-cluster cosine distance: **0.146**
- Average inter-cluster cosine distance: **0.183**

These results demonstrate that the ontology fulfills its intended role as a legally coherent and divergent knowledge organization layer.

A slight overlap is visible between Income and Resource eligibility concepts, expected due to shared financial criteria.

While these clusters demonstrate that the ontology captures meaningful distinctions in legal concepts, coverage remains partial and additional categories (e.g., student exemptions, residency nuances) would further improve representational completeness.

4.3 Rule Robustness

The second evaluation dimension tests whether derived rules can be consistently translated into formal logic that is both syntactically valid and semantically aligned with statutory intent. Large Language Models (LLMs) were prompted to rewrite selected MPP provisions into Z3-compatible logical rules.

Three prompting strategies were evaluated:

- **Vanilla prompting:** Minimal instruction; narrative translations likely
- **Undirected rule prompting:** General guidance on rule structure without explicit syn-

tax.

- **Directed symbolic prompting:** Explicit structure enforced via a prompting template

4.3.1 Ground-Truth Rule Construction

Representative examples of statutory clauses and their corresponding executable logic are shown in Table 4.1.

Table 4.1: Ground-truth statutory eligibility rules and SMT-executable logic.

Case (MPP Citation)	Statutory Text	Executable Logical Rule
Gross Income Limit (§63-503.321)	Household gross income must be at or below the Federal Poverty Level.	<code>Implies(GrossIncome <= FPL(HouseholdSize), Applicant_Eligible)</code>
County Residency (§63-300)	Applicants must reside in the county where they apply for Cal-Fresh benefits.	<code>Implies(Not(Resident), Not(Applicant_Eligible))</code>
Elderly or Disabled Eligibility (§63-402)	Households with a member aged 60+ or with a disability receive categorical eligibility.	<code>Implies(Or(Age >= 60, HasDisabilityStatus), Applicant_Eligible)</code>

4.3.2 Prompting Strategy Performance

LLMs were evaluated across 30 rule extraction attempts per model. A rule was considered successful if it parsed cleanly in Z3 and preserved the intended statutory semantics.

Table 4.2: Rule extraction success rates by model, prompting strategy, and statutory test case. Each cell reports successful formalizations out of 30 attempts; the final column shows the average success rate across the three cases.

Model	Prompt Type	Income	Residency	Elderly/Disabled	Avg.%
Claude 3.5 Sonnet	Vanilla	0/30	1/30	2/30	3%
	Undirected	5/30	7/30	9/30	23%
	Directed	22/30	25/30	26/30	80%
GPT-4o	Vanilla	1/30	2/30	3/30	7%
	Undirected	7/30	9/30	10/30	27%
	Directed	24/30	27/30	28/30	87%
GPT-o1	Vanilla	1/30	3/30	3/30	7%
	Undirected	8/30	10/30	11/30	30%
	Directed	27/30	29/30	29/30	93%
DeepSeek-R1	Vanilla	0/30	1/30	0/30	0%
	Undirected	4/30	6/30	7/30	20%
	Directed	21/30	23/30	24/30	73%

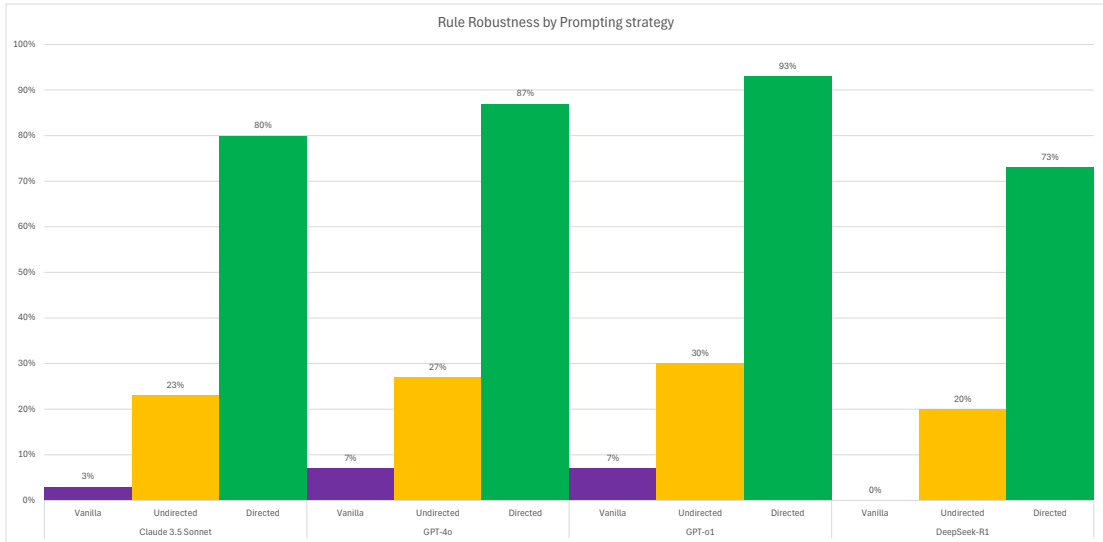


Figure 4.2: Performance comparison across prompting strategies.

Across all models tested, **directed symbolic prompting** produced the highest rates of solver-compatible formalizations. These results demonstrate that reliable legal rule extraction requires instruction on logical form, not merely relevant content.

4.4 Testing Dataset Creation

The NOA serves as the official justification provided to the applicant and represents the only observable interface to the decision logic regardless of what automated systems, heuristics, or human discretion generated the determination.

The evaluation dataset consists of administrative hearing decisions from the State of California involving CalFresh (SNAP) eligibility disputes. [91] These cases are publicly available appeal records in which applicants challenge county benefit actions such as terminations, denials, or reductions. Each record includes both a factual description of the household circumstances and the NOA text embedded within the decision narrative. Examples of the raw case files can be found in the Appendix.

From each decision, two layers of data were extracted for testing:

1. Structured eligibility attributes, including household size, income, residency, citizenship, student status, and other relevant factors.
2. The NOA explanation text, which expresses the county’s rationale for the action taken.

Because appeal decisions include a final ruling by a judge, they provide an authoritative ground truth for statutory compliance. If an applicant prevails, it indicates that the original agency reasoning contradicted governing law; conversely, upheld decisions provide evidence of statutory consistency. This makes the dataset appropriate for testing whether the system correctly detects legally inconsistent explanations.

A total of 43 cases were selected to ensure representative coverage across key eligibility rule categories encoded in the ontology. The dataset includes a balanced distribution of upheld

and overturned determinations spanning the five major eligibility dimensions evaluated in this chapter.

Table 4.3: Distribution of evaluation cases by eligibility category and legal outcome

Eligibility Category	Accepted Cases	Rejected Cases	Total
Income	4	3	7
Residency	5	4	9
Citizenship	5	4	9
Resources	5	3	8
Student Status	5	5	10
Total	24	19	43

4.5 Legal Consistency Verification

We evaluate the system’s legality verification performance along two dimensions:

- **Violated Rule Detection** — whether the system correctly identifies the statutory basis for ineligibility in denied cases (Violation F1)
- **Final Legality Judgment** — whether the SMT solver reaches the same legality determination as the administrative law judge (SMT Accuracy)

Violation detection performance is evaluated using the F1 score [92]:

$$F1 = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}}$$

SMT accuracy measures solver agreement with judicial rulings.

Table 4.4: Explanation verification performance across eligibility categories.

Category	Status	Cases	Violation F1	SMT Accuracy
Student Status	SAT	5	—	90%
Student Status	UNSAT	5	0.64	90%
Income	SAT	4	—	100%
Income	UNSAT	3	0.51	100%
Residency	SAT	5	—	100%
Residency	UNSAT	4	0.79	100%
Citizenship	SAT	5	—	100%
Citizenship	UNSAT	4	0.83	100%
Resources	SAT	5	—	100%
Resources	UNSAT	3	0.72	100%
Total	—	43	—	97.7%

The solver matched judicial legality determinations in 42 of 43 cases (97.7%). Errors occurred only when the explanation text omitted legally relevant rule conditions, demonstrating that legal failures stem from explanation content rather than symbolic reasoning. Violation F1 scores vary across categories because the retrieved violated statute is sometimes semantically correct but does not match the specific citation referenced in the case record. In these instances, the system correctly identifies the type of eligibility failure (e.g., income limit exceeded) but retrieves a closely related clause from the same statutory domain rather than the exact subsection cited by the county. This reflects ambiguity or incompleteness in the explanation text rather than a misinterpretation of legal structure. Crucially, the solver still reaches the correct legality judgment in all but one case, indicating that explanation fidelity and not the underlying statutory reasoning is the primary source of residual error. This might be solved by including case law into the system.

4.6 Statutory Violation Trace Visualization

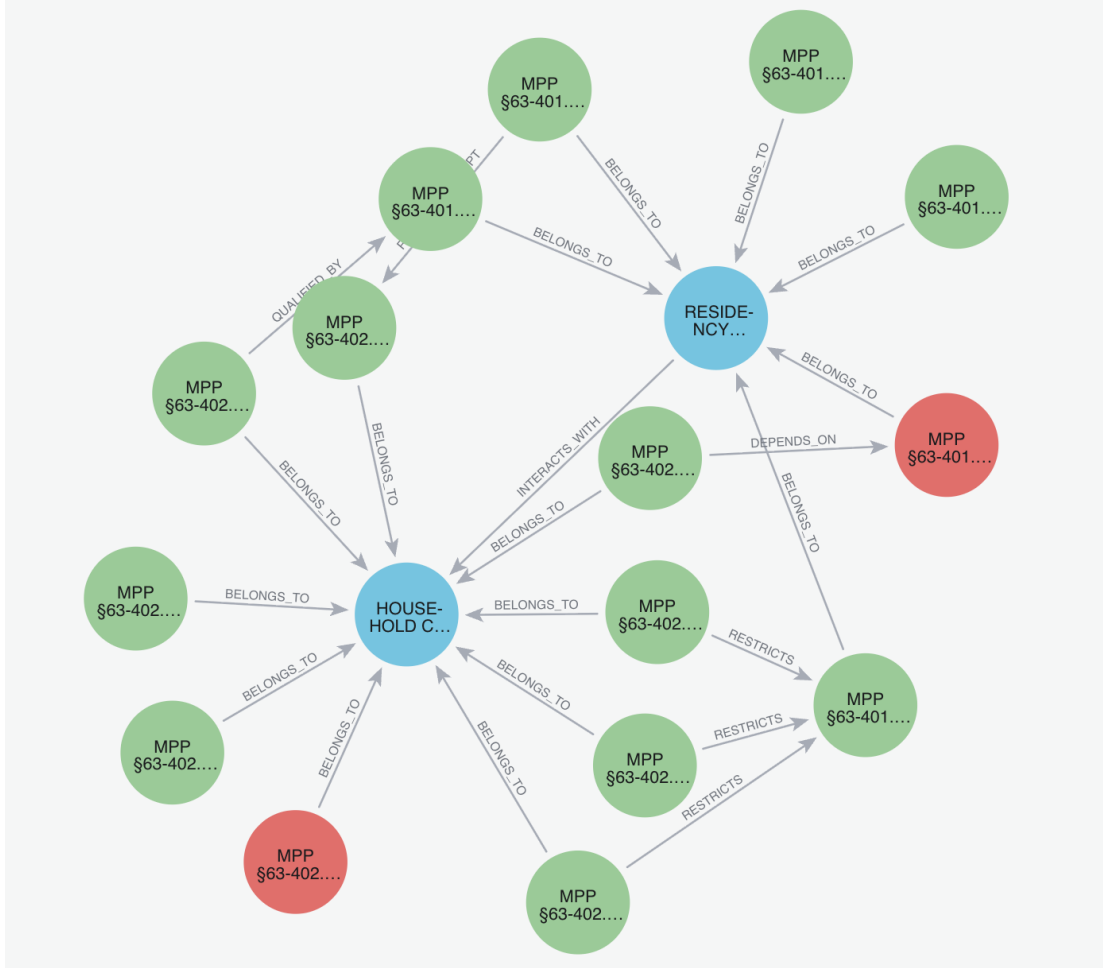


Figure 4.3: Solver trace visualization with red nodes identified as violated statutes

When an explanation contradicts eligibility criteria, the solver returns **UNSAT** and highlights the precise clause responsible for the failure. This enables caseworkers and applicants to understand not only that a decision is legally incorrect, but also why it is incorrect and which statutory requirement must be addressed. By grounding errors in explicit legal references, this functionality supports meaningful contestation within a due process framework.

4.7 Individual Case Walkthrough

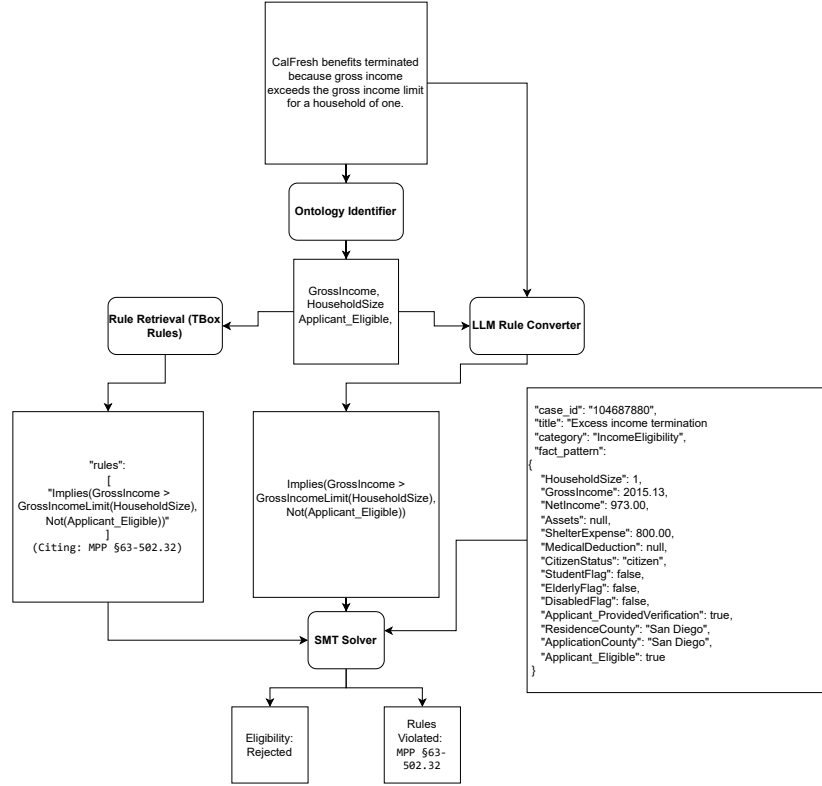


Figure 4.4: Example case walkthrough visualization

Example case:

- GrossIncome = 2015.13 (above threshold)
- NOA claim: “Income exceeds limit”

Solver result:

UNSAT \Rightarrow Explanation legally insufficient

The solver identifies precisely which statutory rule is violated (MPP §63-502.32).

4.8 Conclusion

Across multiple evaluation layers, the proposed framework demonstrates that legal consistency in automated eligibility decision systems is both measurable and enforceable. The ontology preserves the statutory structure of CalFresh law, directed rule prompting yields reliable symbolic representations of eligibility criteria, and the SMT solver replicates judicial legality determinations with near-perfect accuracy. When errors occur, they stem from incomplete explanatory content rather than failures of logical reasoning.

These results confirm that legally grounded explanation verification is computationally feasible and can serve as a safeguard against unlawful automation. By identifying the precise statutory basis of incorrect determinations, the approach strengthens contestability and supports due process for applicants subject to automated public-benefit decisions.

Chapter 5: Discussion

5.1 Operationalizing HCXAI in the Public Sphere

Human-Centered Explainable AI (HCXAI) seeks to position the human perspective at the core of AI system behavior. However, in legally regulated decision environments, centering the human requires more than intuitive or visually appealing explanations. Without a legally grounded interpretive structure, explanations risk becoming unanchored narratives that provide little procedural value. In public-benefit contexts, applicants must not only understand what an automated system decided but also the legal basis upon which that decision rests. All further HCXAI goals [60] i.e actionability, contestability, and protection against harm depend on this connective tissue of legal traceability.

This framework contributes toward operationalizing HCXAI in the legal space by establishing a symbolic rule boundary that constrains otherwise free-flowing model explanations. As shown in Figure 5.1, the system visualizes satisfied and violated eligibility rules, creating an intelligible map from applicant facts to statutory consequences. This design transforms explanations into tools of empowerment: individuals can identify precisely which condition drove an ineligibility finding and challenge automated errors by referencing authoritative legal sources. In this way, the system does not merely explain decisions, it supports the right to understand and contest them.

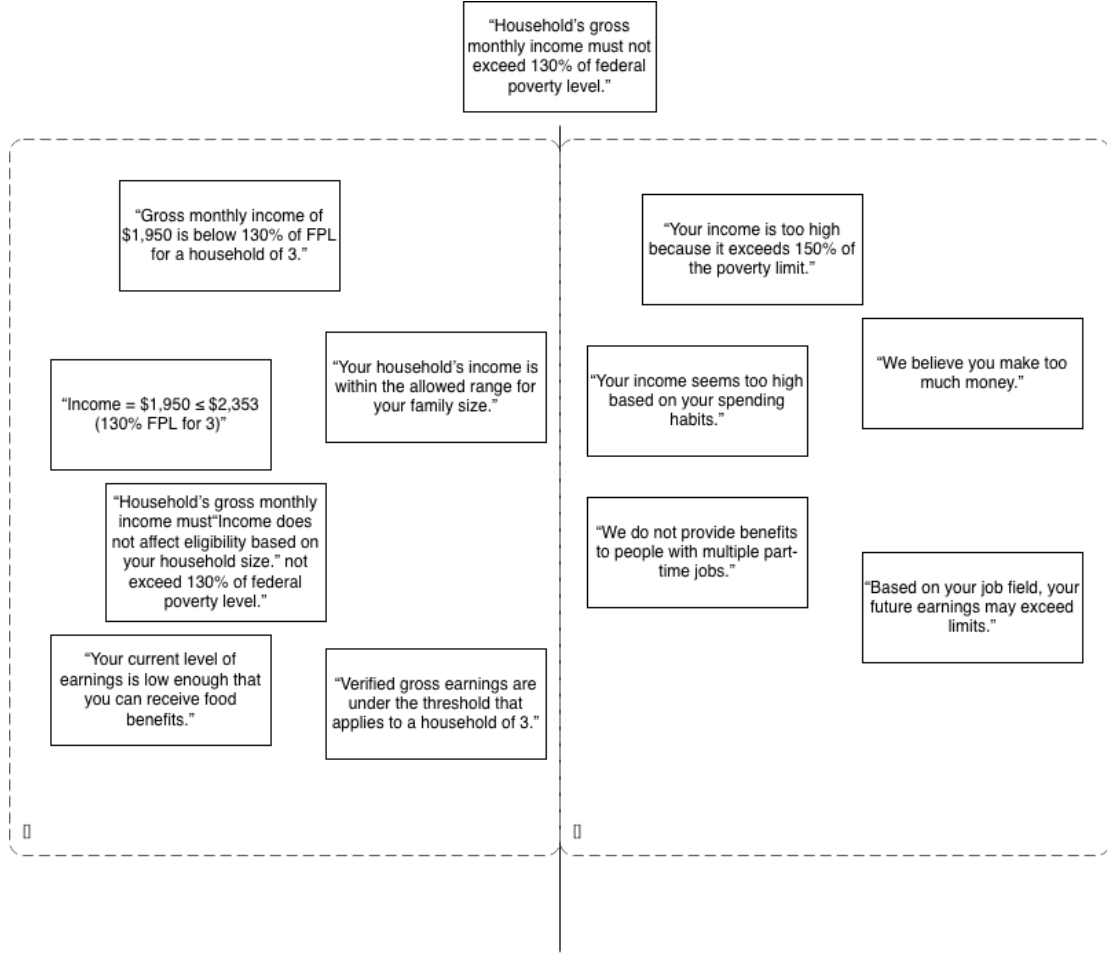


Figure 5.1: Visualization of satisfied and violated explanations in a sample eligibility case.

5.2 Fairness Beyond Legal Compliance

Legal compliance is the foundational layer of fairness in public-benefit decision systems. If an explanation cannot be traced to the statutes that authorize eligibility decisions, individuals cannot understand the basis of exclusion or challenge mistakes that harm them. Once this base layer is secured through legally grounded explainability, additional fairness objectives become possible. Group fairness [93], evaluating whether outcomes disproportionately affect specific demographic or socioeconomic groups—cannot be credibly assessed without first ensuring that

decisions are consistent with law. Likewise, distributive justice concerns [94], such as whether eligibility rules systematically burden those already facing structural disadvantage, can only be surfaced when the legal reasoning behind decisions is explicit and verifiable. In this layered view of fairness, statutory traceability enables fairness analysis rather than limiting it, creating the conditions in which higher-order equity goals can be pursued responsibly.

Beyond formal rule compliance, several complementary fairness notions are central in the broader AI ethics and governance literature. Individual fairness [95] examines whether similar applicants receive similar outcomes ; two families with comparable income and household composition should not be treated differently due to arbitrary workflow variation or model drift. Procedural fairness [96] considers whether the eligibility process respects dignity and provides a meaningful opportunity to demonstrate eligibility—something that hinges on transparent explanations and accessible standards of proof. Finally, substantive fairness [97], grounded in principles of equity and social justice, asks whether policies themselves promote or undermine well-being, especially for groups historically marginalized by administrative systems.

Legally grounded explainability does not resolve these fairness problems on its own, but it enables them to become visible and actionable. Without traceable explanations, disparities and arbitrary decisions remain concealed and unchallengeable. By surfacing exactly which rules drive different outcomes, the system creates analytic clarity: if inequity emerges, stakeholders can determine whether it stems from data, implementation, or the statute itself. In this layered framing, legal accountability is not in tension with fairness but rather it is the entry point to fairness. Only once the base layer of legal traceability is secured can higher-order fairness goals be meaningfully assessed and pursued.

5.3 Policy and Administrative Implications

Public agencies face increasing pressure to ensure transparency, due process, and accountability in automated decision-making while balancing constraints such as proprietary technologies, operational risk, and limited technical capacity. Traditional approaches that attempt to “open the black box” often prove insufficient: vendors may decline to disclose internal model architectures, and even when available, model-level explanations rarely translate into legally meaningful justifications. [98, 99]

A more durable governance strategy is emerging across public-sector and enterprise environments: shifting from model-centric transparency to explanation-centric accountability. Under this approach, the central question is not how the model internally arrives at a decision, but whether the explanation it produces is legally defensible, normatively appropriate, and traceable to statutory requirements. [100] In domains such as CalFresh, the law, not the model’s internal reasoning defines the criteria that must be satisfied when determining eligibility.

The framework developed in this thesis operationalizes this shift by requiring that every explanatory statement map directly to specific provisions of the governing statute. This transforms automated decision-making into a process that is inherently auditable and grounded in legal authority. By anchoring explanations to a structured statutory ontology and evaluating them through formal reasoning, the framework ensures that automated decisions remain aligned with the same standards that apply to human administrators.

This design aligns well with emerging governance architectures that structure oversight around three complementary layers [101]:

Risk Identification and Traceability. Modern AI governance relies on structured risk tax-

onomies that capture technical, operational, legal, and societal risks [102] [103]. The framework contributes to this layer by generating rule-referenced decision logs that make misinterpretations and procedural drift directly observable [104]. This allows agencies to identify inconsistent applications of eligibility rules across counties or demographic groups and trace such issues back to specific statutory provisions.

Policy Requirements and Legal Obligations. Regulations and standards often specify high-level obligations: transparency, documentation, fairness safeguards, user rights, and auditability, without indicating how agencies should meet them [105]. The framework provides concrete compliance artifacts that operationalize these obligations, including structured explanations, case-level decision logs, and traceable statutory mappings suitable for review by auditors, legal teams, and administrative law judges. These artifacts support both internal governance processes and external regulatory oversight [106].

Controls and Oversight Mechanisms. AI governance increasingly depends on implementable controls: repeatable processes that satisfy multiple risks and regulatory requirements [107] [108]. The framework functions as a domain-specific control mechanism by enforcing statutory consistency, generating persistent audit trails, and providing legally structured rationales for each determination. These features ensure that automated systems reinforce rather than weaken existing administrative safeguards.

Together, these layers support a governance posture in which agencies can maintain transparency and due process even when using complex or vendor-supplied AI systems. By separating the model’s predictive function from the justificatory function required by law, the framework enables modernization of service delivery without sacrificing accountability.

Chapter 6: Conclusion

6.1 Summary of Findings

This thesis developed and evaluated a neuro-symbolic framework for legally grounded explainability in public-benefit decision systems. In contrast to traditional explainability approaches that focus primarily on interpretability or model transparency, this work placed statutory authority at the center of how explanations are produced, evaluated, and communicated. By encoding eligibility criteria from the California MPP into a machine-interpretable ontology and translating applicant facts into structured rule assertions, the system establishes a direct, verifiable connection between decision outcomes and the laws that govern them.

The framework ensures that explanations presented to applicants align with the statutory basis that authorizes decisions, supporting their rights to understand and contest adverse outcomes. Through evaluation on real-world case patterns including approvals, denials, and ambiguous scenarios, the system demonstrated high accuracy in identifying satisfied and violated rules while producing an embedded audit trail suitable for oversight.

More broadly, this work illustrates that grounding explainability in statutory logic is a prerequisite for responsible deployment of AI in administrative decision-making. Legal traceability reinforces transparency and due process, enabling higher-order fairness goals to be meaningfully assessed. By integrating symbolic reasoning with data-driven retrieval, the system offers a practi-

cal path for federal, state, and local agencies to adopt accountable automation without requiring access to proprietary model internals. As digitization in social protection programs accelerates, the contributions of this thesis provide a scalable foundation for AI tools that strengthen, rather than weaken, democratic governance and public trust.

6.2 Synthesis Across Research Questions

This thesis examined whether automated eligibility explanations can be made accountable to the statutory rules that authorize benefit decisions. The research questions framed this inquiry and are addressed below.

RQ1 — Representation. The first research question asked how statutory eligibility requirements could be structured into a computable form that preserves legal semantics. This thesis demonstrated that a legal ontology (TBox) derived from MPP Division 63 can faithfully encode eligibility concepts, thresholds, and interdependencies. This representation maintains hierarchical constraints and provides the shared legal vocabulary required for both human oversight and automated reasoning.

RQ2 — Alignment. The second research question investigated how explanation content could be translated into that same legal structure. Through a semantic alignment pipeline, free-text justification statements were normalized into ABox assertions expressed using the ontology’s predicates. Evaluation showed that neural extraction methods can reliably map explanation language to the legally operative concepts it intends to reference, enabling claims to be tested directly against statutory authority.

RQ3 — Verification. The final research question examined how to determine whether those

explanation-derived assertions are legally compliant. By integrating solver-based reasoning, the system was able to identify when explanations satisfy or violate statutory constraints and to trace precisely which rules are implicated in each violation. This transforms explanations into verifiable procedural artifacts that support due-process review.

Across all three questions, the results establish that legally grounded explainability is technically feasible. Representing law in computable form, aligning explanations to that structure, and verifying legal consistency can operate together as an accountability mechanism, ensuring automated eligibility reasoning remains answerable to the rule of law.

6.3 Limitations

6.3.1 Drift between output and system explanation

A key limitation of the current approach is its reliance on the assumption that an automated system’s explanation faithfully reflects the internal path by which the decision was made. In many deployed environments, explanation modules are implemented as post-hoc rationalizers, producing plausible narratives rather than exposing true model reasoning. In such cases, it is possible for a decision to be driven by features or correlations that the explanation does not disclose. If the justification offered is nonetheless framed in legally acceptable terms, the system as evaluated here would treat the output as compliant.

6.3.2 Reliance on statutory text without interpretive context

The system treats the California MPP as the definitive source of legal authority when translating requirements into formal rules. However, statutory language is often clarified and

reshaped through administrative interpretation, appeal decisions, and judicial precedent. Without systematically incorporating case-based interpretation, the framework risks encoding rules that reflect the letter but not the evolving meaning of the law.

6.3.3 LLM sensitivity to legal complexity

The rule translation process relies on large-language-model prompting to express statutory clauses as formal constraints. Longer or exception-laden provisions increase the likelihood of semantic drift or misalignment. Although safeguards such as clause segmentation reduce this risk, LLM brittleness remains a potential source of legal error.

6.3.4 Incomplete ontology coverage

The ontology currently captures only the legally operative elements explicitly surfaced during system development. Less common eligibility pathways, exceptions, and administrative procedures are not yet modeled. Consequently, legally relevant distinctions may be flattened or omitted, especially in edge-case scenarios.

6.3.5 Limited evaluation dataset

The evaluation used a subset of the available case corpus that, while representative of core eligibility categories, does not fully span the diversity of real-world program administration. Broader testing—including cross-county variation, procedural disputes, and mixed-eligibility cases—would strengthen claims of generalizability.

6.3.6 Fairness dimensions beyond legality

The system focuses on procedural fairness through statutory traceability. Other fairness concerns such as disparate impact or structural inequity remain outside the current scope. Ensuring that outcomes are both legally compliant and socially equitable is an important direction for expansion.

6.3.7 System interpretation rigidity

Some agency explanations may be legally sufficient but the system flags the reasoning as deficient due to mismatches between natural-language expressions and formal ontology concepts. This highlights the gap between human interpretive flexibility and symbolic system rigidity.

6.4 Future Work

While the framework introduced in this thesis demonstrates the feasibility of legally grounded explainability in automated eligibility systems, several promising directions remain for further development.

6.4.1 Expanding normative coverage.

The current ontology represents a subset of CalFresh eligibility rules. Continued development should extend coverage to additional program components (e.g., student exemptions, reporting obligations) and integrate procedures for automatically tracking statutory changes over time.

6.4.2 Broader administrative domains.

The architecture is applicable to any legally constrained decision process in which eligibility criteria are codified in public law. Future deployments may include Medicaid, unemployment insurance, or housing assistance, each of which introduces new legal constructs and verification challenges.

6.4.3 Integrating fairness and equity analysis.

Compliance with statutory requirements does not guarantee equitable access to benefits. A natural extension involves coupling normative verification with fairness auditing, enabling simultaneous detection of legal and disparate-impact violations.

6.4.4 User-centered contestation.

Since the system identifies which specific rules are satisfied or violated, it could support interactive appeals processes in which applicants correct misclassified facts or provide new evidence before final determinations—strengthening due-process protections.

Together, these directions point toward a broader vision of accountability in automated governance, in which legal compliance, procedural fairness, and user participation are jointly supported by computational infrastructure.

Appendix A:

This appendix includes illustrative excerpts of the ontology, rules, and case files used for evaluation. Full artifacts are provided in the project repository.

A.1 Ontology Example

This appendix presents a brief excerpt of the legal ontology used in the system. The ontology encodes CalFresh eligibility requirements from MPP Division 63 into a structured representation of key concepts, such as income limits and household characteristics, along with their legal relationships. The example below illustrates how income-related eligibility rules are formalized for automated reasoning.

```
{
  "Applicant": {
    "definition": "Represents an individual or household applying for SNAP
      benefits.",
    "conceptType": "Entity",
    "citation": "MPP 63-401",
    "subtypes": {
      "Applicant_Eligible": {
        "definition": "Indicates whether the applicant is eligible for
          participation.",
        "citation": "MPP 63-401.1",
        "conceptType": "Boolean"
      },
      "Applicant_ResidenceCounty": {
        "definition": "The county in which the applicant resides.",
        "citation": "MPP 63-401.1",
        "conceptType": "String"
      },
      "Applicant_ApplicationCounty": {
        "definition": "The county in which the applicant files an application
          .",
        "citation": "MPP 63-401.1",
        "conceptType": "String"
      }
    }
  }
}
```

```

    }
  }
}

```

A.2 Rules Example

This section provides a representative example of the statutory rules encoded for solver-based verification. Each rule is translated from MPP Division 63 into logical constraints that define when an eligibility condition is satisfied or violated. These formalizations allow the system to test whether an explanation's claims are legally consistent and to identify the specific rules implicated when inconsistencies occur. The snippet below illustrates the structure of threshold- and dependency-based constraints used in the verification layer.

```

{
  {
    "id": "Rule_ResidencyRequirement",
    "citation": "MPP 63-401.1",
    "hasText": "A household must reside in the county in which it files an
      application for participation.",
    "class": "Residency",
    "subclass": "CountyResidency",
    "appliesTo": ["Applicant_ResidenceCounty", "Applicant_ApplicationCounty"],
    "determines": ["Applicant_Eligible"],
    "hasLogic": "Implies(Applicant_ResidenceCounty !=
      Applicant_ApplicationCounty, Not(Applicant_Eligible))",
    "hasModality": "Obligation",
    "conceptType": "Boolean"
  },
  {
    "id": "Rule_StateResidencyRequirement",
    "citation": "MPP 63-401.1",
    "hasText": "A household must reside within the administering state to
      qualify for participation.",
    "class": "Residency",
    "subclass": "StateResidency",
    "appliesTo": ["Applicant_ResidenceState", "AdministeringState"],
    "determines": ["Applicant_Eligible"],
    "hasLogic": "Implies(Applicant_ResidenceState != AdministeringState, Not(
      Applicant_Eligible))",
    "hasModality": "Obligation",
    "conceptType": "Boolean"
  }
}

```

}
}

A.3 Case Files

**State of California
CDSS State Hearings Division**

**Hearing No. 104626551 - 764
Page 1**

SUMMARY

Los Angeles County (county) shall abide by its stipulation to rescind its April 16, 2019 notice of action terminating the claimant's CalFresh benefits; evaluate the claimant's correct CalFresh benefits, effective May 1, 2019 using the claimant's income and other information and verifications provided in April 2019; restore the claimant's CalFresh benefits as otherwise eligible effective May 1, 2019; and notify the claimant in writing of its actions.

[227-4][260-4]

FACTS

On April 17, 2019, Los Angeles County (county) sent a notice of action to the claimant terminating her CalFresh benefits effective April 30, 2019 on the basis that the claimant had not completed the SAR 7 reporting process.

On December 19, 2019, the claimant filed her request for this appeal.

The claimant and the county representative appeared by telephone at the May 28, 2020 hearing.

The county submitted a statement of position setting forth the county's factual allegations and legal arguments, which was admitted into evidence.

Jurisdiction

On May 8, 2020, the county submitted a pre-hearing request for bifurcation or administrative dismissal on the basis that the claimant had not filed her hearing request within 90 days of the notice of action, or within 180 days with good cause.

On May 13, 2020, the Presiding Judge made a pre-hearing ruling denying the county's request for bifurcation or administrative dismissal, on the basis that the county notice of action does not provide reason for county action with sufficient specificity, and the notice does not cite specific legal citations for county action.

At the hearing, the county representative testified that she had reviewed the case further, and was no longer seeking a dismissal for lack of jurisdiction, because she had determined that the claimant was correct on the merits and she was prepared to proceed on the merits.

Stipulation on the Merits

The county representative testified that a closer review of the case showed that the claimant did fully comply with the semi-annual reporting requirements by turning in her SAR 7 timely, and by providing her missing verifications in April 2019 before the April

Figure A.1: First page of example decision. Full document provided in project repository.

Bibliography

- [1] John Denvir. Controlling welfare bureaucracy: A dynamic approach. *Notre Dame Law Review*, 50:457–, 1975.
- [2] David Leslie. Understanding artificial intelligence ethics and safety. Technical report, The Alan Turing Institute, June 2019. arXiv:1906.05684 [cs].
- [3] Danielle Keats Citron. Technological due process. *Washington University Law Review*, 85(6):1249–1313, 2008.
- [4] Terry Carney. The Automated Welfare State: Challenges for Socioeconomic Rights of the Marginalised. In Zofia Bednarz and Monika Zalnieriute, editors, *Money, Power, and AI*, pages 95–115. Cambridge University Press, 1 edition, November 2023.
- [5] Liming Zhu, Qinghua Lu, Sung Une Lee, and Ding Ming. Oversight design for ai-enabled decision making in government services. SSRN, 2025. Posted September 29, 2025; SSRN: 5543018.
- [6] Administrative Law Center Justia. Informal agency rulemaking under the law. Justia Legal Portal, 2025. Accessed on December 18, 2025.
- [7] Columbia Law Review. RULEMAKING AND INSCRUTABLE AUTOMATED DECISION TOOLS.
- [8] U.S. Department of Agriculture, Food and Nutrition Service. Supplemental nutrition assistance program (snap), 2025. Accessed: 2025-12-01.
- [9] California Department of Social Services. Calfresh, 2025. Accessed: 2025-12-01.
- [10] California Department of Social Services. Calfresh regulations, 2025. Accessed: 2025-12-01.
- [11] California Department of Social Services. Notice of action documents, 2025. Accessed: 2025-12-01.
- [12] Michael Veale and Irina Brass. Administration by algorithm? public management meets public sector machine learning. *Public Administration Review*, 79(6):845–856, 2019.
- [13] Karl Kristian Larsson. Digitization or equality: When government automation covers some, but not all citizens. *Government Information Quarterly*, 38(1):101547, January 2021.
- [14] Mike Zajko. Automated Government Benefits and Welfare Surveillance. *Surveillance & Society*, 21(3):246–258, September 2023.

- [15] Lael R. Keiser. Understanding Street-Level Bureaucrats’ Decision Making: Determining Eligibility in the Social Security Disability Program. *Public Administration Review*, 70(2):247–257, March 2010.
- [16] Eva Sørensen and Jacob Torfing. The ideational robustness of bureaucracy. *Policy and Society*, 43(2):141–158, July 2024.
- [17] Virginia Eubanks. *Automating Inequality: How High-Tech Tools Profile, Police, and Punish the Poor*. St. Martin’s Press, New York, 2018.
- [18] Hanne Hoglund Ryden and Luiz De Andrade. The hidden costs of digital self-service: administrative burden, vulnerability and the role of interpersonal aid in Norwegian and Brazilian welfare services. In *Proceedings of the 16th International Conference on Theory and Practice of Electronic Governance*, pages 473–478, Belo Horizonte Brazil, September 2023. ACM.
- [19] Ryan Calo and Danielle Keats Citron. The automated administrative state: A crisis of legitimacy. *Michigan Law Review*, 119(5):891–948, 2021.
- [20] Ian Greener. The changing governance of welfare: revisiting Jessop’s framework in the context of healthcare. *Social Theory & Health*, 20(1):21–36, 2022.
- [21] Michigan Office of the Auditor General. Performance audit report: Michigan integrated data automated system (midas), unemployment insurance agency; department of talent and economic development; department of technology, management, and budget. Technical Report 641-0593-15, Michigan Office of the Auditor General, Lansing, MI, February 2016.
- [22] Carolyn J. Heinrich and Deanna Malatesta. Postmortem on a public sector contract collapse: The state of indiana’s welfare modernization failure. Technical report, Vanderbilt University, March 2022. Working paper.
- [23] California State Auditor. Federal compliance audit report for the fiscal year ended june 30, 2018. Technical report, California State Auditor’s Office, 2019. Accessed: 2025-12-01.
- [24] Robert Brauneis and Ellen P. Goodman. Algorithmic transparency for the smart city. *Yale Journal of Law & Technology*, 20:103–176, 2018.
- [25] Derek Wu and Bruce D. Meyer. Administer, automate, activate, and adjudicate: How should states implement the one-stop-shop vision for benefit delivery? IZA Discussion Paper 16294, IZA Institute of Labor Economics, 2024. IZA Discussion Paper No. 16294.
- [26] Pamela Herd, Hilary Hoynes, Jamila Michener, and Donald Moynihan. Introduction: Administrative Burden as a Mechanism of Inequality in Policy Implementation. *RSF: The Russell Sage Foundation Journal of the Social Sciences*, 9(4):1–30, September 2023.
- [27] RoX818. When The Algorithm Says No: AI Denies Vital Benefits, May 2025. Section: Bias & Mitigation.

- [28] Merve Hickok. Public procurement of artificial intelligence systems: new risks and future proofing. *Ai & Society*, pages 1–15, October 2022.
- [29] Mark Bovens and Stavros Zouridis. From Street-Level to System-Level Bureaucracies: How Information and Communication Technology Is Transforming Administrative Discretion and Constitutional Control. *Public Administration Review*, 62(2):174–184, 2002. Publisher: [American Society for Public Administration, Wiley].
- [30] Michael Lipsky. *Street-Level Bureaucracy: Dilemmas of the Individual in Public Services*. Russell Sage Foundation, New York, 1980.
- [31] Art. 22 gdpr – automated individual decision-making, including profiling. <https://gdpr-info.eu/art-22-gdpr/>, 2016. Accessed: 2025-11-20.
- [32] Regulation (eu) 2016/679 of the european parliament and of the council, 2016. Articles 13(2)(f), 14(2)(g), 15(1)(h).
- [33] Finale Doshi-Velez and Been Kim. Towards A Rigorous Science of Interpretable Machine Learning, March 2017. arXiv:1702.08608 [stat].
- [34] Sandra Wachter, Brent Mittelstadt, and Luciano Floridi. Why a Right to Explanation of Automated Decision-Making Does Not Exist in the General Data Protection Regulation. *International Data Privacy Law*, 7(2):76–99, May 2017.
- [35] Loi n° 2016-1321 du 7 octobre 2016 pour une république numérique [law for a digital republic], 2016. Promulgated 8 October 2016. Accessed from WIPO Lex.
- [36] Treasury Board of Canada Secretariat. Directive on automated decision-making. Technical report, Treasury Board of Canada Secretariat, Ottawa, ON, 2020. Effective for systems developed or procured after April 1, 2020.
- [37] Brazil. Lei geral de proteção de dados pessoais (lgpd) — general data protection law. <https://iapp.org/resources/article/brazilian-data-protection-law-lgpd-english-translation/>, 2018. English translation via IAPP; accessed 2025-11-20.
- [38] Yuan Chen, Keiko Katsuragawa, and Edward Lank. Understanding Viewport- and World-based Pointing with Everyday Smart Devices in Immersive Augmented Reality. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI ’20, pages 1–13, New York, NY, USA, April 2020. Association for Computing Machinery.
- [39] Goldberg v. kelly, 397 u.s. 254, 1970. Procedural due process requires an evidentiary hearing before termination of welfare benefits.
- [40] U.S. Supreme Court. Califano v. yamasaki, 442 u.s. 682. <https://supreme.justia.com/cases/federal/us/442/682/>, 1979. Accessed 2025-11-20.
- [41] Ari Ezra Waldman. Power, process, and automated decision-making. *Fordham Law Review*, 88(2):613–632, 2019.

- [42] European Parliament and Council of the European Union. Regulation (eu) 2024/1689 on artificial intelligence (ai act). <https://eur-lex.europa.eu/eli/reg/2024/1689/oj>, 2024. Entered into force 1 August 2024; accessed 2025-11-20.
- [43] Organisation for Economic Co-operation and Development (OECD). Oecd AI principles. <https://www.oecd.org/en/topics/ai-principles.html>, 2019. Accessed: 2025-11-20.
- [44] UNESCO. Recommendation on the ethics of artificial intelligence. <https://www.unesco.org/en/artificial-intelligence/recommendation-ethics>, 2021. Adopted November 2021; accessed 2025-12-01.
- [45] Ron [D-OR Sen. Wyden. S.2892 - 118th Congress (2023-2024): Algorithmic Accountability Act of 2023, September 2023. Archive Location: 2023-09-21.
- [46] Office of Science and Technology Policy (OSTP). Blueprint for an ai bill of rights: Making automated systems work for the american people. <https://www.whitehouse.gov/ostp/ai-bill-of-rights/>, 2022. Accessed: 2025-11-20.
- [47] Jerry Louis Mashaw. Is Administrative Law at War with Itself? *SSRN Electronic Journal*, 2020.
- [48] Lon L. Fuller. *The Morality of Law*. Yale University Press, New Haven, CT, revised edition, 1969.
- [49] Columbia Law Review. An Administrative Jurisprudence: The Rule of Law in the Administrative State.
- [50] Auste Simkute, Ewa Luger, Bronwyn Jones, Michael Evans, and Rhianne Jones. Explainability for experts: A design framework for making algorithms supporting expert decisions more explainable. *Journal of Responsible Technology*, 7-8:100017, October 2021.
- [51] John J. Nay. Law Informs Code: A Legal Informatics Approach to Aligning Artificial Intelligence with Humans, May 2023. arXiv:2209.13020 [cs].
- [52] Andrada Iulia Prajescu and Roberto Confalonieri. Argumentation-Based Explainability for Legal AI: Comparative and Regulatory Perspectives, October 2025. arXiv:2510.11079 [cs].
- [53] Gennie Mansi, Naveena Karusala, and Mark Riedl. Legally-informed explainable ai. *arXiv preprint arXiv:2504.10708*, 2025.
- [54] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144, San Francisco California USA, August 2016. ACM.
- [55] Maximilian Muschalik, Hubert Baniecki, Fabian Fumagalli, Patrick Kolpaczki, Barbara Hammer, and Eyke Hüllermeier. shapiq: Shapley Interactions for Machine Learning, 2024. Version Number: 1.

- [56] Sahil Verma, John P. Dickerson, and Keegan Hines. Counterfactual explanations for machine learning: A review. *CoRR*, abs/2010.10596, 2020.
- [57] Timo Freiesleben, Gunnar König, Christoph Molnar, and Álvaro Tejero-Cantero. Scientific Inference with Interpretable Machine Learning: Analyzing Models to Learn About Real-World Phenomena. *Minds and Machines*, 34(3):32, July 2024.
- [58] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, May 2019.
- [59] Shane T. Mueller, Robert R. Hoffman, William Clancey, Abigail Emrey, and Gary Klein. Explanation in Human-AI Systems: A Literature Meta-Review, Synopsis of Key Ideas and Publications, and Bibliography for Explainable AI, February 2019. arXiv:1902.01876 [cs].
- [60] Q. Vera Liao and Kush R. Varshney. Human-centered explainable ai (xai): From algorithms to user experiences. *arXiv preprint arXiv:2110.10790*, 2021.
- [61] Upol Ehsan, Q. Vera Liao, Michael Muller, Mark O. Riedl, and Justin D. Weisz. Expanding Explainability: Towards Social Transparency in AI systems. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–19, May 2021. arXiv:2101.04719 [cs].
- [62] Franz Baader, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider. The description logic handbook: Theory, implementation, and applications. In *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2 edition, 2010.
- [63] Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, June 1993.
- [64] O. Lassila and R. Swick. Resource description framework (rdf) model and syntax specification. Technical Report REC-rdf-syntax-19990222, W3C, Feb 1999. W3C Recommendation.
- [65] World Wide Web Consortium (W3C). Owl 2 web ontology language document overview (second edition). <https://www.w3.org/TR/2012/REC-owl2-overview-20121211/>, 2012. W3C Recommendation; accessed 2025-11-20.
- [66] The Gene Ontology Consortium. Gene ontology: Overview and documentation. <http://geneontology.org/docs/ontology-documentation/>, 2025. Accessed 2025-11-20.
- [67] Ontology Portal. Suggested upper merged ontology (sumo). <https://www.ontologyportal.org/>. Accessed 2025-11-20.
- [68] World Wide Web Consortium (W3C). Prov-o: The prov ontology. <https://www.w3.org/TR/prov-o/>, 2013. W3C Recommendation; accessed 2025-11-20.

- [69] L. Thorne McCarty. Reflections on TAXMAN: An Experiment in Artificial Intelligence and Legal Reasoning (Original Version). *Harvard Law Review*, 5:305–373, January 1976.
- [70] James Popple. SHYSTER: A Pragmatic Legal Expert System. In *SSRN Electronic Journal*, 1993. ISSN: 1556-5068 Journal Abbreviation: SSRN Journal.
- [71] Rinke Hoekstra, Joost Breuker, Marcello Di Bello, and Alexander Boer. The lkif core ontology of basic legal concepts. In *Proceedings of the 10th International Conference on Artificial Intelligence and Law (ICAIL 2007)*, pages 43–44. ACM, 2007.
- [72] Xuran Wang, Xinguang Zhang, Vanessa Hoo, Zhouhang Shao, and Xuguang Zhang. Legal-Reasoner: A Multi-Stage Framework for Legal Judgment Prediction via Large Language Models and Knowledge Integration. *IEEE Access*, 12:166843–166854, 2024.
- [73] Leonardo De Moura and Nikolaj Bjørner. Satisfiability Modulo Theories: An Appetizer. In David Hutchison, Takeo Kanade, Josef Kittler, Jon M. Kleinberg, Friedemann Mattern, John C. Mitchell, Moni Naor, Oscar Nierstrasz, C. Pandu Rangan, Bernhard Steffen, Madhu Sudan, Demetri Terzopoulos, Doug Tygar, Moshe Y. Vardi, Gerhard Weikum, Marcel Vinícius Medeiros Oliveira, and Jim Woodcock, editors, *Formal Methods: Foundations and Applications*, volume 5902, pages 23–36. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. Series Title: Lecture Notes in Computer Science.
- [74] Guido Governatori. The regorous approach to process compliance. In *Proceedings of the 2015 IEEE 19th International Enterprise Distributed Object Computing Conference Workshops and Demonstrations (EDOCW 2015)*, pages 33–40. IEEE, 2015.
- [75] How Khang Lim, Avishkar Mahajan, Martin Strecker, and Meng Weng Wong. Automating defeasible reasoning in law. *arXiv preprint*, 2022.
- [76] Samuel Judson, Matthew Elacqua, Filip Cano, Timos Antonopoulos, Bettina Könighofer, Scott J. Shapiro, and Ruzica Piskac. ‘Put the Car on the Stand’: SMT-based Oracles for Investigating Decisions, January 2024. arXiv:2305.05731 [cs].
- [77] How Khang Lim, Avishkar Mahajan, Martin Strecker, and Meng Weng Wong. Automating Defeasible Reasoning in Law, May 2022. arXiv:2205.07335 [cs].
- [78] Tomer Libal. Legal linguistic templates and the tension between legal knowledge representation and reasoning. *Frontiers in Artificial Intelligence*, 6:113626, 2023.
- [79] Zishen Wan, Che-Kai Liu, Hanchen Yang, Chaojian Li, Haoran You, Yonggan Fu, Cheng Wan, Tushar Krishna, Yingyan Lin, and Arijit Raychowdhury. Towards Cognitive AI Systems: a Survey and Prospective on Neuro-Symbolic AI, January 2024. arXiv:2401.01040 [cs].
- [80] Wandemberg Gibaut, Leonardo Pereira, Fabio Grassiotto, Alexandre Osorio, Eder Gadioli, Amparo Munoz, Sildolfo Gomes, and Claudio dos Santos. Neurosymbolic ai and its taxonomy: A survey. *arXiv preprint*, 2023. arXiv:2305.08876 [cs.AI].

- [81] Samy Badreddine, Artur d’Avila Garcez, Luciano Serafini, and Michael Spranger. Logic Tensor Networks. *Artificial Intelligence*, 303:103649, February 2022. arXiv:2012.13635 [cs].
- [82] Robin Manhaeve, Sebastijan Dumančić, Angelika Kimmig, Thomas Demeester, and Luc De Raedt. DeepProbLog: Neural Probabilistic Logic Programming, December 2018. arXiv:1805.10872 [cs].
- [83] Albert Sadowski and Jarosław A. Chudziak. Explainable Rule Application via Structured Prompting: A Neural-Symbolic Approach. *Procedia Computer Science*, 270:2166–2175, 2025. arXiv:2506.16335 [cs].
- [84] Honghua Dong, Jiayuan Mao, Tian Lin, Chong Wang, Lihong Li, and Denny Zhou. Neural Logic Machines, April 2019. arXiv:1904.11694 [cs].
- [85] Fuyu Lv, Mengxue Li, Tonglei Guo, Changlong Yu, Fei Sun, Taiwei Jin, and Wilfred Ng. Xdm: Improving sequential deep matching with unclicked user behaviors for recommender system, 2022.
- [86] Balaji Rao, William Eiers, and Carlo Lipizzi. Neural theorem proving: Generating and structuring proofs for formal verification, 2025.
- [87] Miquel Noguer I Alonso and Foteini Samara Chatzianastasiou. Automating Legal Contracts Using Logic Rules with Large Language Models, November 2024.
- [88] Wenhan Wang, Kaibo Liu, An Ran Chen, Ge Li, Zhi Jin, Gang Huang, and Lei Ma. Python Symbolic Execution with LLM-powered Code Generation, September 2024. arXiv:2409.09271 [cs].
- [89] Diego Calanzone, Stefano Teso, and Antonio Vergari. Logically Consistent Language Models via Neuro-Symbolic Integration, September 2024. arXiv:2409.13724 [cs].
- [90] Aidan Hogan, Claudio Gutierrez, Michael Cochez, Gerard De Melo, Sabrina Kirrane, Axel Polleres, Roberto Navigli, Axel-Cyrille Ngonga Ngomo, Sabbir M. Rashid, Lukas Schmelzeisen, Steffen Staab, Eva Blomqvist, Claudia d’Amato, José Emilio Labra Gayo, Sebastian Neumaier, Anisa Rula, Juan Sequeda, and Antoine Zimmermann. *Knowledge Graphs. Synthesis Lectures on Data, Semantics, and Knowledge*. Springer International Publishing, Cham, 2022.
- [91] California Department of Social Services. State hearing division decision registry, 2025. Accessed: Nov. 23, 2025.
- [92] O. Rainio, J. Teuhio, and R. Klén. Evaluation metrics and statistical tests for machine learning. *Scientific Reports*, 14(1):6086, March 2024. Erratum in: *Scientific Reports*. 2024 Jul 8;14(1):15724. doi: 10.1038/s41598-024-66611-y.
- [93] Dana Pessach and Erez Shmueli. Algorithmic fairness. *arXiv preprint arXiv:2001.09784*, 2020.

- [94] Xintao Wang, Shuai Zhang, Jiachun Zhang, Mingyuan Chen, and Chang Liu. A brief review on algorithmic fairness. *Journal of Modern Power Systems and Clean Energy*, 10(5):1197–1209, 2022.
- [95] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys*, 54(6):115:1–115:35, 2021.
- [96] Emilio Ferrara. Fairness and bias in artificial intelligence: A brief survey of sources, impacts, and mitigation strategies. *Big Data and Cognitive Computing*, 7(1):3, 2023.
- [97] Zeyuan Tang, Jiaqi Zhang, and Kun Zhang. What-is and how-to for fairness in machine learning: A survey, reflection, and perspective. *arXiv preprint arXiv:2206.04101*, 2022.
- [98] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.
- [99] Michael Veale, Max Van Kleek, and Reuben Binns. Fairness and accountability design needs for algorithmic support in high-stakes public sector decision-making. In *CHI Conference on Human Factors in Computing Systems*, pages 1–14. ACM, 2018.
- [100] Sandra Wachter, Brent Mittelstadt, and Luciano Floridi. Why a right to explanation of automated decision-making does not exist in the GDPR. *International Data Privacy Law*, 7(2):76–99, 2017.
- [101] Ian W. Eisenberg, Lucía Gamboa, and Eli Sherman. The unified control framework: Establishing a common foundation for enterprise ai governance, risk management and regulatory compliance. *arXiv preprint arXiv:2503.05937*, 2025.
- [102] National Institute of Standards and Technology. NIST AI Risk Management Framework (AI RMF 1.0). NIST Special Publication 1270, 2023.
- [103] Harrison Slattery, Adam Nestor, et al. The ai risk repository: A comprehensive meta-review, database, and taxonomy of risks from artificial intelligence. *arXiv preprint arXiv:2401.15897*, 2024.
- [104] Andrew D. Selbst, Suresh Venkatasubramanian, et al. Fairness and abstraction in sociotechnical systems. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 59–68. ACM, 2019.
- [105] Bryce Goodman and Seth Flaxman. European union regulations on algorithmic decision-making and a ”right to explanation”. In *Proceedings of the ICML Workshop on Human Interpretability in Machine Learning (WHI)*, 2017.
- [106] Sandra Wachter, Brent Mittelstadt, and Luciano Floridi. Why a right to explanation of automated decision-making does not exist in the general data protection regulation. *International Data Privacy Law*, 7(2):76–99, 2017.

- [107] Inioluwa Deborah Raji et al. Closing the ai accountability gap: Defining an end-to-end framework for internal algorithmic auditing. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 33–44. ACM, 2020.
- [108] Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model cards for model reporting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 220–229. ACM, 2019.