Embedding doctrinal evaluation logic in legal AI: a constraint-based model of criminal offence evaluation

Jessica Herrmann, Frank Breitinger

Angaben zur Veröffentlichung / Publication details:

Herrmann, Jessica, and Frank Breitinger. 2026. "Embedding doctrinal evaluation logic in legal AI: a constraint-based model of criminal offence evaluation." In *RELAF 2026 - Reasoning with Evidence in Law Enforcement and Forensics 2026: proceedings of the Workshop on Reasoning with Evidence in Law Enforcement and Forensics (RELAF 2026) co-located with the International Conference on Artificial Intelligence and Law (ICAIL 2026), Singapore, June 8, 2026*, edited by Floris Bex and Sungmi Park, 93–106. Aachen: CEUR-WS. https://nbn-resolving.org/urn:nbn:de:0074-4253-x.

Embedding Doctrinal Evaluation Logic in Legal AI: A Constraint-Based Model of Criminal Offence Evaluation

Jessica Herrmann[1,2], Frank Breitinger[1]

[1]Chair for Cybersecurity, Institute of Computer Science, University of Augsburg, Augsburg, Germany
[2]Technology, Logistics and Service Directorate, Police Baden-Württemberg, Stuttgart, Germany

Abstract

AI-based forensic tools can identify patterns that may indicate criminally relevant behavior. Such technical signals do not establish whether the legal requirements of a criminal offence are met. In substantive criminal law, legal assessment follows a doctrinal evaluation process in which individual elements must be examined in a fixed order and according to dependencies. Intermediate findings may terminate an evaluation branch or activate an alternative path, such as the assessment of an attempt. Existing AI approaches address different aspects of legal reasoning, but are not necessarily designed to represent the meta-level constraints governing doctrinal offence evaluation. This paper presents a conceptual constraint-based model representing this evaluation logic through dynamically activated evaluation nodes, separating hierarchical legal structure from constraints governing the evaluation path. In a future implementation, semantic assessment may be performed locally at individual nodes, while selection and sequencing remain controlled by the constraint model rather than a language model. The representation is intended to support both complete doctrinal assessment and more limited investigative evaluations focused on selected offence elements and evidentiary gaps. By making evaluation nodes and dependencies explicit, the approach provides explainability by design and ensures traceability and auditability. The manual reconstruction of doctrinal constraints indicates that many rules governing the evaluation path are not expressed in statutory text, but must be reconstructed from legal doctrine and jurisprudence. Implementation and empirical evaluation remain future work.

Keywords

legal reasoning, doctrinal evaluation logic, criminal law, constraint-based modelling, evaluation paths

1. Introduction

AI-based forensic tools can detect patterns of potentially criminal behavior in large communication datasets such as chat logs containing indicators of cyber-grooming or similar harms [1, 2, 3]. Commercial forensic tools already use machine learning to analyze large volumes of communication data and flag relevant content. For example, Magnet.AI, integrated into the Magnet AXIOM platform, uses machine learning to identify and prioritize chat conversations indicating child grooming or related criminal activity [4]. These systems act as early-stage filters, highlighting content that may warrant further legal assessment and contribute to establishing initial suspicion. In policing contexts, such systems are subject to strict requirements due to their potential impact on fundamental rights, a discussion reinforced by the EU AI Act, which introduces specific obligations for high-risk AI systems used in law enforcement [5]. These regulatory requirements do not themselves define how criminal-law assessments must be structured, but they reinforce the need for legal evaluation processes that remain explicit and reconstructable.

This raises a central question: *what constitutes legal correctness when AI-supported systems contribute to assessing potentially criminal conduct?* Unlike traditional classification tasks, correctness cannot be evaluated solely against annotated datasets or similarity to past cases. In substantive criminal law, whether conduct constitutes an offence is determined through a structured evaluation process defined by doctrinal patterns rather than by textual similarity. Here, *doctrinal evaluation logic* denotes the meta-level rules governing this process, not criminal procedure or procedural law. These rules determine which legal elements become relevant, in which order they may be examined, and how intermediate

RELAF'26: Reasoning with Evidence in Law Enforcement and Forensics, Workshop at ICAIL 2026, Singapore.
jessica.herrmann@uni-a.de (J. Herrmann); frank.breitinger@uni-a.de (F. Breitinger)

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073
published 2026-08-31

findings may terminate or redirect the evaluation path. Given a candidate offence structure and case facts, the model represents the activated evaluation path, node states, and evidentiary gaps. German substantive criminal law provides a suitable domain because criminal liability is already organized through a structured doctrinal framework [6, 7], whose evaluation logic must be made explicit for computational representation. While statutory law primarily specifies what must be fulfilled, doctrine determines how these elements are evaluated in a concrete case. The present work constitutes a conceptual prototype: it reconstructs and models the meta-level constraints governing this evaluation process, but does not yet provide an implemented or empirically evaluated reasoning system.

The remainder of the paper is organized as follows. Section 2 reviews related work. Section 3 formulates the problem and outlines the main contributions. Section 4 introduces the constraint-based procedural model. Section 5 presents its instantiation through activation patterns and a case study. Section 6 discusses implications. Section 7 addresses limitations and future work. Section 8 concludes.

2. Related Work

Data-driven approaches to detecting criminal behavior in communication data, such as online grooming detection, rely on machine learning methods applied to chat logs [1, 2, 3]. Related case-based approaches derive outcomes from similarities to prior cases rather than from explicit doctrinal evaluation paths [8, 9]. Such approaches are useful for filtering, retrieval, or outcome-oriented reasoning, but parts of the reasoning process may remain implicit [10]. Argumentation- and dialogue-based models capture justification structures and formal relations between arguments [11, 12], but they do not directly represent the internal dependency structure through which offence elements, unlawfulness, and culpability are doctrinally evaluated. A further challenge concerns the relation between legal text and formal representation. Even simple logical structures are not reliably preserved when legal language is translated into formal representations [13]. Taken together, these limitations suggest that doctrinal evaluation logic cannot be inferred from statutory text or case similarity alone, but must be reconstructed from legal doctrine and modelled explicitly.

The problem becomes particularly relevant where contexts require different depths of evaluation. Judicial reasoning requires complete doctrinal evaluation, whereas investigative settings may initially focus on selected objective elements. A suitable representation must therefore support variation in evaluation depth while preserving doctrinal validity, transparency, and accountability [6, 14]. Early symbolic approaches pursued different forms of explicit legal knowledge representation. TAXMAN modelled legal concepts and their relations [15], while LEGOL provided a formal language for specifying legal rules [16]. These approaches illustrate the long-standing role of explicit representation in AI and law [17]. The present work builds on this tradition, but focuses on a narrower modelling problem in substantive criminal law. In doctrinal offence evaluation, a complete evaluation path is not selected as a predefined case constellation. Rather, it emerges step by step from the assessment of individual elements and the intermediate findings produced during that assessment. The rules governing these path transitions are reconstructed from legal doctrine and represented here as constraints over evaluation nodes. Process-oriented representations, such as workflow-based or BPMN-style models, can describe predefined procedural sequences, but usually address organizational or legal procedures rather than the internal structure of doctrinal offence evaluation.

Knowledge-representation-based approaches, including LegalRuleML [18] and expert-annotated legal knowledge bases such as JUREX-4E [19], represent legal rules, norms, offence elements, and their relations in machine-readable form, but remain closer to local representation than to global path control. Deontic and permission-based approaches formalize normative reasoning at the level of permissions, obligations, and violations [20]. They address normative status rather than the path-generation logic that determines which evaluation steps become relevant, how they are sequenced, and when a branch is terminated or redirected. Recent large language model (LLM)-based approaches offer greater flexibility in working with textual legal materials and case-specific factual descriptions [21]. This flexibility is relevant for local subsumption, where evaluation nodes are assessed against variable case facts. Purely LLMdriven systems, by contrast, do not reliably reproduce doctrinally valid evaluation sequences, especially in multi-step legal assessments [22, 23]. Hybrid approaches that combine LLMs with structured representations partially address this limitation [24, 25, 26, 27, 28]. The central requirement for the present task is therefore a separation between local semantic assessment and global path control: LLMs support node-level assessment, while constraints govern the evaluation path.

3. Problem Statement and Contributions

Statutory law defines legal elements and conditions but only partially specifies the meta-level logic governing their evaluation. The problem addressed here is the representation of doctrinal offence evaluation as a constraint-governed process. A suitable model must combine hierarchical legal structure with constraints on activation, prerequisites, ordering, exclusion, and termination, while allowing local interpretation at evaluation nodes. Building on prior work on structured legal reasoning and explainability [17, 11, 12, 29], this work focuses on the constraint-based representation of doctrinal offence evaluation. The proposed approach uses constraints to control the evaluation path, while a future implementation may use language models for local semantic assessment. The model is conceptual and has not yet been implemented or empirically evaluated. The paper makes the following contributions:

- We reconstruct meta-level constraints of doctrinal offence evaluation and distinguish those derivable from statutory text from those requiring doctrinal sources.
- We introduce a conceptual constraint-based model that separates the hierarchical representation of legal elements from constraints governing the admissible evaluation path.
- We illustrate how the model represents context-dependent evaluation paths with explicit node states, transitions, and evidentiary gaps.

4. Constraint-Based Procedural Model

This section introduces a constraint-based procedural model that represents legal reasoning as a structured evaluation process. A central design principle is the separation between hierarchical legal structure and procedural evaluation logic.

4.1. Structural Backbone

The model adopts the classical doctrinal structure of criminal liability assessment in German criminal law. At the highest level, evaluation follows the established sequence of *fulfillment of offence elements* (FOE), *unlawfulness*, and *culpability*. These stages form the hierarchical backbone of the model, as illustrated in Figure 1.

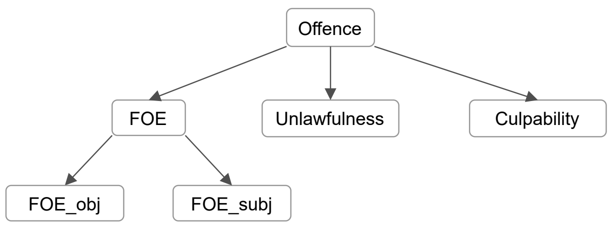

Figure 1: Hierarchical backbone of criminal liability assessment.

Within this hierarchy, the *FOE* level is subdivided into *objective* and *subjective* components, which are further organized into more fine-grained elements and substructures. A more detailed representation of these relations is provided in Figure 3. The hierarchy does not define a sequence of evaluation steps; it organizes legal elements into nested structural containers of varying granularity. This allows dependencies to be defined not only between individual elements, but also across groups or higher-level structures. The hierarchy provides a static organization of statutory elements and doctrinal refinements, but does not prescribe how they are evaluated. Some elements interact across multiple parts of the structure without belonging to a single branch. Modelling these relations within the hierarchy alone would create cross-cutting dependencies and structural entanglement. Procedural aspects such as activation, ordering, dependencies, and termination are defined separately through constraints. This separation between structural representation and procedural control enables flexible, context-dependent evaluation.

4.2. Mapping Norms to Evaluation Nodes

The model does not operate directly on statutory text, but on a reconstructed procedural structure derived from legal doctrine. Statutory provisions typically describe offence elements, while the ordered evaluation process emerges from doctrinal interpretation and jurisprudence. Legal norms must therefore be mapped onto an abstract evaluation structure before constraint-based modelling can be applied. This separates textual legal formulations from procedural evaluation logic and allows different offences to share a common structural backbone. Statutory offences are classified into structural types, such as result-based (RBO), conduct-based (CBO), or truncated result-based offences (TRBO). These types determine the general evaluation pattern by specifying which categories of evaluation nodes are included or omitted. They are not represented as hierarchical nodes, but control which parts of the shared structure are activated.

On this basis, statutory elements are mapped onto evaluation nodes within the predefined structure. This mapping does not mirror the textual organization of the norm, but assigns each element to its functional role within the evaluation process. As a result, different offences can be represented as variations over a shared structural framework, while preserving their doctrinal evaluation logic. For example, unauthorized use of a vehicle (§248b StGB) is conduct-based, theft (§242 StGB) is truncated result-based, and extortion (§253 StGB) is result-based. These classifications directly affect the evaluation structure. RBO requires nodes such as result, causation, and attribution, whereas CBO omits them.

Act_specific Object

Further obj. Elements

(1) Whoever takes movable property belonging to another away from another with the intent of unlawfully appropriating it for themselves or a third party incurs a penalty of imprisonment for a term not exceeding five years or a fine.

(2) The attempt is punishable. Attempt

Further_subj_Elements

Figure 2: Example decomposition of §242 StGB (Theft) into procedural evaluation nodes.

Figure 2 illustrates this mapping for §242 StGB. Statutory language is decomposed into evaluation nodes according to functional role rather than textual order. For example, the phrase "takes movable property belonging to another away from another" is linked to the nodes *Object* and *Act_specific*, while intent-related phrases are linked to separate subjective nodes. This fine-grained decomposition captures distinctions that become relevant in later evaluation steps. Standard doctrinal schemata do not always separate elements such as *Act* and *Result* at this level. These distinctions become critical in subsequent reasoning. Variations such as attempt, omission, or error doctrines depend on which specific component of the offence fails.

They can only be represented if individual elements are modelled as separate evaluation nodes. This element-level mapping differs from IRAC-style reasoning (Issue, Rule, Application, Conclusion), where the norm is treated as a single unit for subsumption. While IRAC supports coherent local reasoning, dependencies between individual components remain implicit and cannot be controlled independently. The division into individual elements is not defined in the law itself. It is derived from criminal law doctrine. As a result, only a limited subset of procedural constraints can be derived directly from statutory provisions. Most dependencies between evaluation steps originate from doctrine.

Type	Offence	Norm (StGB)	Object	Result	Act specific	Further obj Elements	Causation Attribution	Further subj Elements	Attempt
CBO	Unauthorized use of vehicle	§248b	+	-	+	+	-	-	+
RBO	Extortion	§253	-	+	+	+	+	+	+
TRBO	Theft	§242	+	-	+	+	-	+	+

Table 1: Distribution of statutory elements across evaluation nodes for selected offences. The table shows how different structural types lead to different activation patterns over the same shared structure. Depending on the offence type, certain evaluation nodes are included or omitted.

Once elements are mapped onto evaluation nodes, procedural behavior is defined by constraints operating on the shared structure. Structure types determine which parts of this structure are relevant for a given offence and define offence-specific relation patterns without modifying the underlying representation. Norms assigned to the same structural type therefore share the same evaluation logic. This allows generalization at the level of structural types.

4.3. Constraint Operators

Procedural relations are represented using constraint operators: ACTIVATES for conditional activation, REQUIRES for prerequisite dependencies, ORDER for evaluation precedence, STOP for termination, EXCLUDES for incompatibility, XOR for exclusive alternatives, and REF for cross-level references. For example, ORDER(A, B) expresses evaluation precedence, and REQUIRES(B, A) encodes prerequisite dependencies. Together, the operators cover activation, ordering, exclusion, termination, alternative evaluation paths, and cross-level dependencies in doctrinal reasoning.

We developed the current set of constraint operators iteratively. We initially modelled classical case examples manually as decision trees for different structure types, covering both objective and subjective offence elements. This helped us identify the required operator types. Before the doctrinal extraction, we conducted an exploratory LLM-assisted analysis of the General Part of the German Criminal Code. This step required approximately 75 hours, including manual review, and supported the mapping of statutory elements, but yielded only about 40 constraints relevant to the evaluation process. This reinforced the finding that most meta-level evaluation rules are not expressed directly in statutory text and must be reconstructed from doctrinal sources. We therefore continued with manual extraction from doctrinal literature. We first tested and refined the operator set using Roxin/Greco [6] and subsequently applied it systematically to Rengier [7].

At the current stage of the ongoing reconstruction, we have manually analyzed 134 of 632 pages of Rengier's textbook. The extraction and subsequent review required approximately 61 hours. Using a pattern-based qualitative approach, we abstracted doctrinal statements into constraint templates and consolidated duplicate instances. The resulting model currently comprises 161 evaluation nodes and 272 deduplicated constraints across the identified operator types. The extraction and coding were performed by one researcher and required interpretive decisions when doctrinal statements did not map unambiguously onto a single node or operator type.

The current model spans multiple layers of doctrinal structure, including core offence evaluation as well as additional legal prerequisites and related structural components. As the extraction is based on a limited subset of doctrinal sources, no claims are made regarding the completeness or distribution of procedural dependencies across the full legal structure. Each operator represents a generalized pattern observed across multiple doctrinal instances.

Table 2 shows how recurring formulations in doctrinal texts can be mapped to constraint operators. Core evaluation sequences are described consistently across sources, but some dependencies remain underspecified or vary between representations. This is especially visible in cross-level relations and alternative evaluation paths. The constraint-based model makes these differences explicit by requiring defined relations between evaluation nodes.

The operator set is not intended to be formally minimal. Some operators may overlap in their effect, but they capture different semantic roles in legal reasoning. The model prioritizes semantic alignment with legal doctrine over formal minimality. At the same time, the operator set should be understood as a structured representation of recurring procedural patterns observed in doctrinal reasoning, not as a fully formalized system with complete semantics.

Source	Paraphrase	Constraint
[7], §10 RN 18	Liability for omission requires the existence of a legal duty to act.	`Omission REQUIRES Duty_to_Act`
[7], §12 RN 6	In CBO, result, causation, and objective attribution are not part of the evaluation.	`structure_type_CBO EXCLUDES Result`
[7], §12 RN 9	Causation constitutes an implicit objective element in RBO.	`structure_type_RBO ACTIVATES Causation`
[7], §12 RN 11	Within the subjective elements of the offence, intent is evaluated prior to additional subjective elements, which are only considered if intent is established.	`ORDER(Intent, Further_Subj_Elements)`
[7], §13 RN 2	If objective attribution of the result to the offender fails, the evaluation of completion is terminated.	`Obj_Attribution=false STOP Completion`
[7], §14 RN 5	Subjective elements refer to objective elements	`REF(Intent, Obj_Elements)`
[7], §14 RN 6	Exactly one form of intent (purpose, direct intent, or conditional intent) applies in a given case.	`XOR(Purpose, Direct_Intent, Conditional_Intent)`

Table 2: Mapping of doctrinal reasoning patterns to constraint operators.

A complete formalization would require a comprehensive and systematically validated reconstruction of doctrinal dependencies, which is beyond the scope of this work. The current model does not yet provide full formal semantics for interactions between multiple constraints or consistency guarantees for the overall evaluation structure. For example, it does not define how to resolve conflicting constraints or how to prevent circular dependencies between evaluation steps. These aspects remain subjects for future work.

5. Instantiation of Legal Reasoning through Activation

5.1. Integrated Representation of Legal Evaluation

Legal evaluation results from the interaction between hierarchical structure and constraint operators. The model represents this interaction as dynamic activation over a shared structural backbone, as shown in Figure 3. Hierarchical relations organize legal elements, while constraint operators such as `ORDER` and `XOR` determine admissible evaluation paths. Cross-level dependencies are represented using the reference operator `REF`. Evaluation proceeds through the activation of nodes based on these constraints.

For readability, Figure 3 presents a reduced abstraction of the full model. In practice, doctrinal evaluation exhibits much deeper hierarchical refinement. This is omitted here to highlight the interaction between structural and procedural components. The visual differentiation of node types reflects a central limitation of statutory law. Some elements (blue) can be derived directly from statutory or ontological structures, such as act, object, or result. A substantial part of the evaluation relies on doctrinally constructed concepts (orange), including causation, attribution, and intent. This distinction also affects the procedural structure that governs their evaluation.

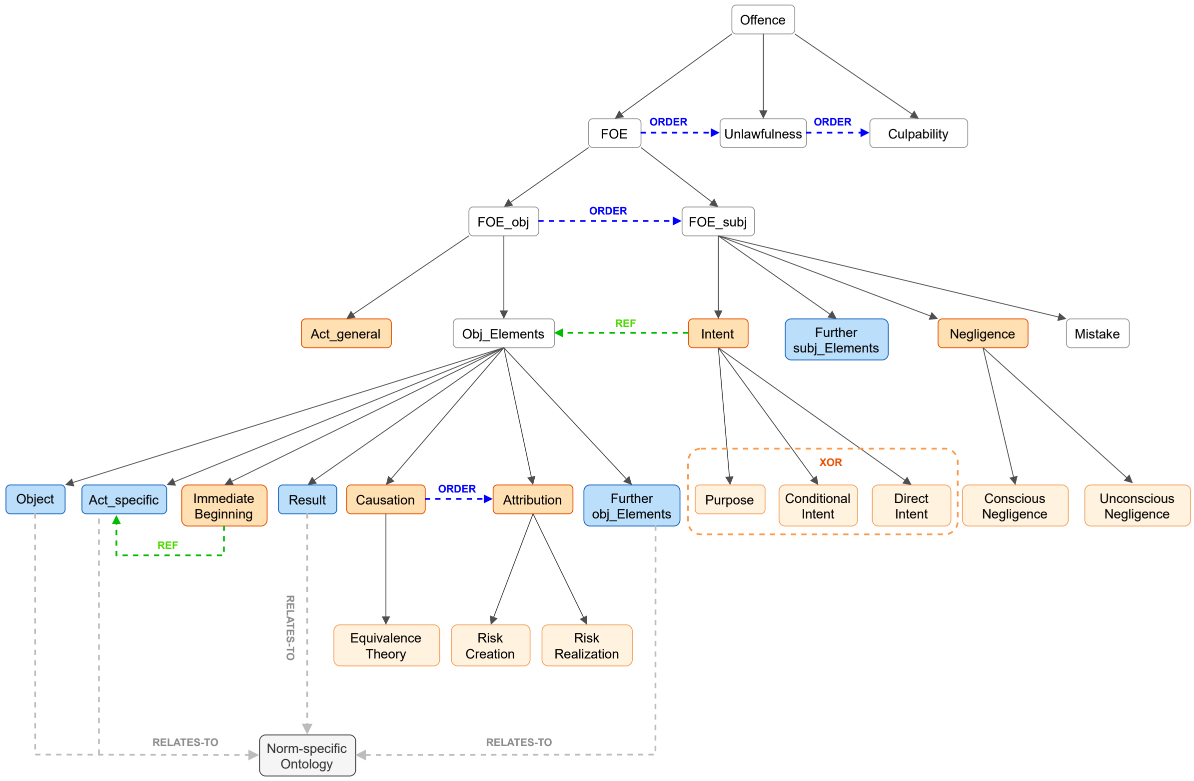

Figure 3: Integrated representation of hierarchical structure, procedural constraints, and semantic dependencies. Solid edges denote hierarchical relations. Dashed blue edges represent ORDER constraints, dashed green edges indicate reference relations (REF), and dashed grey edges indicate exemplary links from selected objective offence elements to a norm-specific ontology (RELATES-TO). Nodes are colour-coded according to their role. Blue nodes correspond to elements derived from statutory or ontological structures, white nodes denote structural containers, and orange nodes represent doctrinal concepts. Darker orange indicates core doctrinal concepts such as causation, attribution, and intent, while lighter orange denotes their internal refinement.

5.2. Procedural Structure and Constraint Interaction

Ordering constraints (ORDER) play a central role in separating structural decomposition from procedural sequencing. ORDER(A, B) means that A must be evaluated before B, but not necessarily directly before it. Unlike hierarchical relations, ordering constraints are not limited to parent-child structures, but may span across different levels of abstraction. Legal reasoning is therefore represented as a partially ordered evaluation process with dependencies across hierarchical levels rather than as a strictly linear sequence. For example, a general evaluation step, such as the concept of action, may precede an entire class of objective offence elements, not just a single node. Ordering relations can therefore be propagated across the hierarchy, allowing evaluation sequences to be reconstructed from a limited set of constraints.

The hierarchical structure is not intended to cover all levels of legal abstraction. The required depth is determined by the presence of procedural dependencies, such as activation, termination, or requirement relations. Nodes below this level mainly serve a definitional role and can be represented using ontological descriptions without explicit constraint modelling. Some dependencies cannot be captured through hierarchy or ordering alone. For example, intent is defined in relation to objective elements, creating cross-level dependencies between otherwise separate branches. These relations are represented using reference operators (REF), extending the model beyond purely hierarchical or sequential representations.

This reflects a structural asymmetry between statutory and doctrinal sources. Statutory norms define elements, but the procedural dependencies governing their evaluation must be reconstructed from doctrine rather than extracted from legal text. Legal reasoning, therefore, emerges as a multi-layered process that combines statutory content, doctrinal interpretation, and procedural structuring.

5.3. Activation Patterns Across Offence Types

The model represents legal reasoning as a dynamic activation over a shared structural backbone rather than as predefined offence-specific evaluation trees. Different offence types correspond to activation patterns that select different subsets of nodes. Constraint operators determine which nodes become active, in which order they are evaluated, and under which conditions evaluation paths terminate. Figure 4 illustrates this behavior for two structurally different offence types. For clarity, the activation patterns are reduced to the objective offence elements and their transition to the subjective stage.

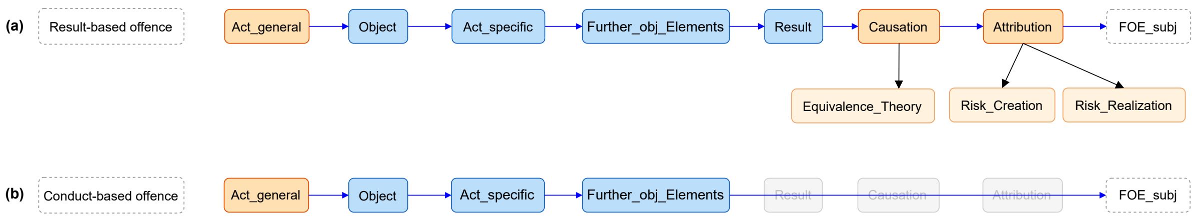

Figure 4: Activation patterns for (a) a result-based offence and (b) a conduct-based offence within the same structural model.

In a result-based offence, the objective evaluation follows the sequence *Act_general* → *Object* → *Act_specific* → *Further_obj_Elements* → *Result* → *Causation* → *Attribution*, before transitioning to *FOE_subj*. In a conduct-based offence the evaluation is limited to *Act_general* → *Object* → *Act_specific* → *Further_obj_Elements*, followed by a direct transition to *FOE_subj*. This comparison shows that different offence types correspond to structurally distinct subgraphs within the same model. Both patterns are derived from the same underlying structure. Differences arise from constraint-governed activation, not from changes in representation itself. Legal reasoning can therefore be modelled as a process in which offence-specific evaluation paths emerge dynamically from a shared conceptual structure.

5.4. Illustrative Case Study

Example: To illustrate the process, consider a simplified forensic scenario. A smartphone extraction reveals a chat in which the suspect repeatedly demands money and threatens to publish private images if payment is not made. The account used in the chat can be attributed to the suspect's device. The victim refuses payment, and no transfer of money occurs. The threatening content indicates intentional conduct, but the available evidence does not allow a further specification of the form of intent beyond this general indication.

Initial offence selection: Before activating a specific evaluation pattern, candidate offence structures must be identified based on the available facts. These candidates can be obtained using standard retrieval or classification methods, such as embedding-based similarity or LLM-assisted classification. The model does not depend on a specific method for this step, as it only requires a set of candidate structures as input. In the given scenario, the case description yields candidate offence types such as extortion, coercion, and fraud, which may be ranked according to plausibility. The highest-ranked candidate determines the structural pattern used for subsequent constraint-based evaluation.

Input
Case facts: "The suspect repeatedly demands money and threatens to publish private images if payment is not made."
Candidates: extortion, coercion, fraud

Output
```
Extortion = likely (confidence: high)
```

This initial selection step does not determine the correctness of the subsequent evaluation path itself, as alternative candidate structures can be evaluated iteratively if required. The selected result-based offence structure activates the corresponding evaluation nodes and constraint relations. The resulting activation state reflects the absence of a completed offence and the transition to an attempt structure. Table 3 summarizes the resulting node states.

Step	Node	Status
1	Act_general	evaluated (+)
2	Act_specific	evaluated (+)
3	Further_obj_Elements	evaluated (+)
4	Result	not satisfied
5a	Completion	not satisfied
5b	Causation / Attribution	not activated (REQUIRESResult)
6	Attempt	activated (+)
7	Intent	evaluated (unspecified)
8	Purpose / Direct_Intent / Conditional_Intent	XOR unresolved (Intent types)
9	Unlawfulness	not activated
10	Culpability	not activated
11	Mistake-related nodes	not activated

Table 3: Node states for the illustrative case.

The absence of a result prevents the activation of downstream nodes such as *Causation* and *Attribution*, which depend on a satisfied result. Instead, the evaluation is redirected to the attempt branch, where *Attempt* is activated. The subjective element *Intent* can be evaluated at a general level, as the threatening content indicates intentional conduct but remains unspecified due to the lack of confirming evidence. As a consequence, the exclusive alternatives represented through XOR (e.g., purpose, direct intent, conditional intent) remain unresolved. Because the transition to subsequent stages depends on the completion of prior evaluation steps, nodes such as *Unlawfulness*, *Culpability*, and mistake-related elements are not activated. The evaluation path is therefore determined by constraint dependencies that block completion, activate the attempt branch, and prevent further progression until unresolved elements are addressed.

Beyond illustrating the evaluation process, the example also highlights the model's ability to identify incomplete evidentiary coverage. While several objective elements can be evaluated based on the available evidence, other components, such as the specification of intent or the conditions required for completion, remain unresolved. This makes it possible to systematically detect which elements of a legally relevant evaluation path are not sufficiently supported by the current evidence. In an investigation context, such incomplete activation patterns can indicate where additional information is required before a case can be forwarded for prosecutorial or judicial review. The model, therefore, not only supports the assessment of existing evidence but also helps identify gaps in the investigative process.

5.5. Structured Representation of Constraint-Based Activation

5.5.1. Graph Representation and Node States

The model represents doctrinal offence evaluation as a directed graph $G = (N, E_h, E_c)$, where N is the set of evaluation nodes, E_h represents hierarchical relations, and E_c represents constraint-based relations between nodes. Each node $n \in N$ is assigned an activation state

$$s(n) \in \{inactive, active, evaluated, excluded\}.$$

For evaluated nodes, the assessment result is represented separately as

$$r(n) \in \{satisfied, not\ satisfied, unresolved\}.$$

This distinction separates whether a node is part of the evaluation path from the outcome of its legal assessment. All nodes are defined within a controlled vocabulary with unique identifiers and fixed hierarchical positions.

5.5.2. Activation Process

The activation process proceeds as follows:

1. Initialize all nodes as inactive.
2. Select one or more candidate legal provisions and activate the nodes associated with their corresponding offence structure.
3. Activate further nodes when their prerequisites and applicable activation conditions are satisfied.
4. Restrict the evaluation sequence according to applicable ORDERconstraints.
5. Assess active nodes and assign an assessment result of *satisfied*, *not satisfied*, or *unresolved*.
6. Update the evaluation path by applying constraints such as ACTIVATES, EXCLUDES, STOP, XOR, and REF.

Initialization requires selecting one or more candidate legal provisions as entry points for the evaluation, corresponding in practice to an early-stage case assessment based on the available evidence. The model evaluates concrete statutory provisions, while offence categories determine the corresponding structural pattern. These provisions may themselves be hierarchically related, for example, as basic offences and their qualified forms.

Candidate provisions can be identified using similarity-based methods, such as embedding-based retrieval or case analogy. Depending on a predefined threshold, multiple candidate norms may be selected and evaluated sequentially. Their evaluation order may be guided by structural dependencies. For example, basic offence definitions may be evaluated before qualified or aggravated variants. Without this initialization step, the model would either require exhaustive evaluation across all offence types or remain non-operational.

5.5.3. Activation Conditions and Assessment Results

A node becomes active when it is included in the selected offence structure, its applicable prerequisites are satisfied, and no constraint excludes or terminates its evaluation. Activation does not mean that the corresponding legal condition is satisfied. It only indicates that the node forms part of the legally relevant evaluation path and is eligible for assessment. Prerequisite relations such as REQUIRESdetermine whether dependent nodes may become active, while ACTIVATES, EXCLUDES, and STOPmay add nodes to the evaluation path, remove them, or terminate a branch. By contrast, ORDERdoes not determine activation, but restricts the sequence in which active nodes may be evaluated. Later assessment results may exclude nodes that were previously active. In such cases, their activation state is revised to *excluded*. The current formulation is intentionally semi-formal. Full formal semantics, consistency guarantees, and the resolution of interactions between multiple constraints remain subjects for future work.

5.6. Execution Model and LLM Integration

The model does not delegate doctrinal path control to the language model. Instead, the LLM is used only for local subsumption at individual evaluation nodes. Procedural aspects of the evaluation, including activation, ordering, dependencies, and termination, remain governed by the constraint system. The LLM therefore does not control the evaluation order, select offence types, or decide transitions between evaluation branches, such as completion or attempt.

At the semantic level, statutory norms and doctrinal definitions are associated with individual evaluation nodes. Rather than encoding all semantic detail within the procedural model, nodes act as interfaces to underlying representations of legal concepts, for example, in structured ontological formats. During evaluation, the constraint system determines which nodes are eligible for assessment. For each eligible node, the LLM receives the relevant case facts together with a node-specific representation of the corresponding legal condition. It returns a structured node assessment, such as *satisfied*, *not satisfied*, or *unresolved*, together with a short justification grounded in the provided facts and legal conditions. This output updates the state of the evaluated node and may trigger further constraint-based activation, but it does not determine which node is evaluated next.

Evaluation generally proceeds node by node, but closely related groups of nodes may be processed jointly to preserve semantic coherence. This differs from rule-based approaches such as LegalRuleML, where legal evaluation is represented through explicit rules, and relevant conditions must be specified within the rule definitions. The proposed model allows semantically related elements to be assessed within a shared context window, while the overall evaluation process remains controlled by constraints. This avoids unnecessary fragmentation of related elements without delegating activation, ordering, or termination to the LLM.

For example, when evaluating an *Intent* node, the LLM may receive a description of the suspect's behavior together with a structured definition of intent. It may return that intent is *satisfied*, *not satisfied*, or *unresolved*, together with a short justification. The resulting node assessment is then processed by the constraint system, which determines whether further subjective elements, alternative intent types, or downstream evaluation steps become active. Global reasoning thus emerges from a sequence of locally controlled assessments rather than from a single language-model inference. This separation between constraint-based control and LLM-based interpretation keeps procedural decisions and semantic evaluations explicitly represented.

6. Discussion

The proposed model treats the evaluation path as part of legal correctness. Correctness, therefore, depends not only on the outcome but also on compliance with ordering, dependency, and termination constraints. This reflects doctrinal practice, where the validity of a legal conclusion depends on the correct sequence and structure of evaluation steps. Two systems may therefore reach the same outcome while differing in legal correctness, depending on whether the evaluation path adheres to doctrinal requirements.

This has direct implications for explainability and forensic use. Each evaluation step is explicitly represented as a node with defined activation conditions, making the reasoning process traceable. Unlike post hoc explanation methods that reconstruct justifications from a model's internal reasoning trace [30], explainability here is embedded in the procedural structure itself, resulting in explainability by design. This is particularly relevant in investigative contexts, where partial evaluation may already inform decision-making. Judicial settings, by contrast, require complete and procedurally valid evaluation paths. The approach also enables early-stage prototyping, as AI components can be evaluated on controlled sub-paths without requiring a fully automated system.

The reconstruction of procedural constraints in this work highlights structural limitations in both statutory and doctrinal sources. Only a limited subset of constraints could be derived directly from statutory text, while most procedural dependencies had to be reconstructed from doctrine. In addition, some dependencies remained underspecified even at the doctrinal level.

By requiring explicit modelling of dependencies, the approach exposes implicit assumptions and structural ambiguities that remain hidden in traditional representations of legal reasoning. The concrete instantiation of the model is domain-specific and reflects the doctrinal structure of German criminal law, which provides a particularly well-developed evaluation framework. At the same time, the modelling approach itself is not tied to a specific jurisdiction. Representing legal reasoning as a constraint-governed activation process provides a general modelling paradigm that can be adapted to other legal systems by redefining nodes, constraints, and structural relations. While the resulting evaluation graphs are jurisdiction-dependent, the underlying modelling approach remains transferable and can support explainable AI in high-stakes legal domains.

7. Limitations and Future Work

The proposed model represents a preliminary reconstruction based on a limited subset of doctrinal sources. The extraction and paraphrasing of constraints from doctrinal textbooks was performed manually by a single researcher and is therefore subjective. This manual process was necessary because the publisher explicitly prohibits using these materials for model training or automated processing. While statutory texts in Germany are freely available, many doctrinal sources are subject to strict copyright and licensing restrictions. This limits automated or semi-automated constraint extraction from doctrinal materials. Accordingly, a future implementation would not require training a language model on copyrighted doctrinal sources. Instead, the LLM would operate on legally permissible, node-specific representations derived from the manually reconstructed constraint model and associated legal conditions.

A further challenge arises from the scalability of this approach and the variability of doctrinal reasoning. Extending the analysis to larger domains, additional textbooks, commentaries, or international legal codes may require substantial manual effort. Different authors and legal authorities adopt diverging approaches to structuring evaluation processes. This indicates that procedural representations are not uniquely determined, but depend on interpretative choices. Accordingly, the proposed model should be understood as a conceptual formalization of one coherent doctrinal perspective rather than a definitive representation of legal reasoning.

Future work will focus on extending constraint extraction across a broader range of doctrinal sources. This will enable systematic comparison between alternative doctrinal models. It will also explore legally permissible semi-automated or LLM-based approaches where copyright or licensing restrictions allow. This may support the identification of stable core structures as well as context-dependent variations and contribute to refining the operator s et. Future work also includes implementing the model and evaluating it on legal case scenarios. The evaluation will distinguish between node-level correctness, the doctrinal validity of the resulting evaluation path, and the identification of unresolved evidentiary requirements. The model-generated paths will be compared with expert-validated legal assessments.

8. Conclusion

This paper addressed the explicit representation of doctrinal evaluation structure in AI-based legal reasoning. While statutory law defines legal elements and conditions, the ordered sequence of evaluation steps, their dependencies, and activation conditions emerge from legal doctrine and are not directly encoded in legal texts. For the task considered here, legal correctness cannot be reduced to predictive accuracy or similarity to annotated data. It also depends on adherence to a doctrinally valid evaluation path. Based on this, the paper introduced a constraint-based activation model that reconstructs doctrinal evaluation logic as a system of dynamically activated evaluation nodes governed by constraints. The model separates hierarchical legal structure from doctrinal evaluation logic and represents evaluation sequences across different contexts.

The model embeds activation, ordering, dependency, and termination conditions directly into the evaluation process. This supports traceability, auditability, and explainability by design. In a future implementation, language models can be used as local subsumption tools without controlling the evaluation structure. Modelling doctrinal offence evaluation requires explicit representations of evaluation logic in addition to text-based or outcome-oriented approaches. In high-stakes applications such as forensic AI, predictive performance alone is not sufficient.

Declaration on Generative AI The authors used ChatGPT (OpenAI) to assist with text revision, restructuring, and formulation of passages, as well as to improve clarity and conciseness. The tool was also used to support the iterative refinement of argumentation and the organization of sections. All AI-generated suggestions were critically reviewed, adapted, and validated by the authors to ensure correctness, coherence, and alignment with the intended scientific contribution.

References

[1] P. R. Borj, K. Raja, P. Bours, Online grooming detection: A comprehensive survey of child exploitation in chat logs, Knowledge-Based Systems 259 (2023) 110039. URL: https://doi.org/10.1016/j.knosys.2022.110039. doi:10.1016/j.knosys.2022.110039

[2] L. Hamm, S. McKeever, Comparing machine learning models with a focus on tone in grooming chat logs, Frontiers in Pediatrics 13 (2025) 1591828. URL: https://doi.org/10.3389/fped.2025.1591828. doi:10.3389/fped.2025.1591828

[3] G. Michelet, H. Henseler, H. v. Beek, M. Scanlon, F. Breitinger, Fine-tuning large language models for digital forensics: Case study and general recommendations, Digital Threats: Research and Practice 6 (2025) 1–18.

[4] Magnet Forensics, Introducing magnet.ai: Putting machine learning to work in digital forensics, https://www.magnetforensics.com/blog/introducing-magnet-ai-putting-machine-learning-work-forensics/, 2017. Accessed: 2026-04-20.

[5] European Union, Regulation (eu) 2024/1689 of the european parliament and of the council of 13 june 2024 laying down harmonised rules on artificial intelligence (artificial intelligence act), https://eur-lex.europa.eu/eli/reg/2024/1689/oj, 2024. Official Journal of the European Union, L 1689, 2024.

[6] C. Roxin, L. Greco, Strafrecht Allgemeiner Teil Band I: Grundlagen – Der Aufbau der Verbrechenslehre, 5 ed., C.H. Beck, München, 2020.

[7] R. Rengier, Strafrecht Allgemeiner Teil, 17 ed., C.H. Beck, München, 2025.

[8] C. D. Hafner, Conceptual organization of case law knowledge bases, in: Proceedings of the 1st International Conference on Artificial Intelligence and Law (ICAIL 1987), Association for Computing Machinery, 1987, pp. 35–42. doi:10.1145/41735.41740

[9] B. Xu, X. Dai, Z. Liu, H. Xie, X. Yi, S. Wang, Y. Yan, L. Yang, Y. Gu, G. Yu, Legalduet: Learning fine-grained representations for legal judgment prediction via a dual-view contrastive learning, in: Advanced Data Mining and Applications, volume 16197 of *Lecture Notes in Computer Science*, Springer, Singapore, 2026, pp. 337–352. doi:10.1007/978-981-95-3453-1_23

[10] A. Bibal, M. Lognoul, A. de Streel, B. Frénay, Legal requirements on explainability in machine learning, Artificial Intelligence and Law 29 (2021) 149–169. doi:10.1007/s10506-020-09270-4

[11] H. Prakken, A logical framework for modelling legal argument, in: Proceedings of the 4th International Conference on Artificial Intelligence and Law (ICAIL 1993), Association for Computing Machinery, 1993, pp. 1–9. doi:10.1145/158976.158977

[12] H. Prakken, Formal systems for persuasion dialogue, The Knowledge Engineering Review 21 (2006) 163–188. doi:10.1017/S0269888906000865

[13] B. Steffes, D. Sasdelli, Catch the platypus! negated conditionals as a challenge for machine translation from natural language into logical formalisms using large language models, Journal of Computational Law and Legal Technology (2026) 1–7. URL: https://ojs.bonviewpress.com/index.php/JCLLT/article/view/9092. doi:10.47852/bonviewJCLLT62029092

[14] D. Barys.e, R. Sarel, Algorithms in the court: Does it matter which part of the judicial decision-making is automated?, Artificial Intelligence and Law 32 (2024) 117–146. doi:10.1007/s10506-022-09343-6

[15] L. T. McCarty, Reflections on taxman: An experiment in artificial intelligence and legal reasoning, Harvard Law Review 90 (1977) 837–893.

[16] S. Jones, P. Mason, R. K. Stamper, Legol 2.0: A relational specification language for complex rules, Information Systems 4 (1979) 293–305. URL: https://www.sciencedirect.com/science/article/pii/0306437979900231. doi:10.1016/0306-4379(79)90023-1

[17] T. Bench-Capon, M. Araszkiewicz, K. Ashley, et al., A history of ai and law in 50 papers: 25 years of the international conference on ai and law, Artificial Intelligence and Law 20 (2012) 215–319. doi:10.1007/s10506-012-9131-x

[18] M. Palmirani, G. Governatori, A. Rotolo, S. Tabet, H. Boley, A. Paschke, Legalruleml: Xml-based

rules and norms, in: Rule-Based Modeling and Computing on the Semantic Web, volume 7018 of *Lecture Notes in Computer Science*, Springer, 2011, pp. 298–312. doi:10.1007/978-3-642-24908-2_30.
[19] H. Liu, Q. Huang, Q. Chen, Y. Hu, J. Ma, Y. Liu, W. Shen, Y. Feng, JUREX-4E: Juridical expert-annotated four-element knowledge base for legal reasoning, in: Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Suzhou, China, 2025, pp. 3794–3814. URL: https://aclanthology.org/2025.emnlp-main.188/. doi:10.18653/v1/2025.emnlp-main.188
[20] A. Rotolo, C. Di Florio, G. Governatori, Rule-based deontic case-based reasoning, in: Proceedings of the 20th International Conference on Artificial Intelligence and Law, 2025, pp. 288–297.
[21] I. Chalkidis, A. Jana, D. Hartung, M. Bommarito, I. Androutsopoulos, D. Katz, N. Aletras, Lexglue: A benchmark dataset for legal language understanding in english, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 4310–4330. URL: https://aclanthology.org/2022.acl-long.297/. doi:10.18653/v1/2022.acl-long.297
[22] N. Guha, J. Nyarko, D. E. Ho, C. Ré, A. Chilton, et al., Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models, in: Advances in Neural Information Processing Systems, volume 36, Curran Associates, Inc., 2023, pp. 44123–44279. URL: https://proceedings.neurips.cc/paper_files/paper/2023/file/89e44582fd28ddfea1ea4dcb0ebbf4b0-Paper-Datasets_and_Benchmarks.pdf.
[23] W. Deng, H. Yang, L. Ma, W. Li, G. Wang, Explainable legal judgment prediction via concept tree and concept forest reasoning with collegiate bench mechanism, in: Artificial Intelligence and Human-Computer Interaction, volume 385 of *Frontiers in Artificial Intelligence and Applications*, IOS Press, 2024, pp. 194–201. doi:10.3233/FAIA240152
[24] K. Zhang, W. Yu, Z. Sun, J. Xu, An explicit syllogistic legal reasoning framework for large language models, arXiv (2025). URL: https://arxiv.org/abs/2504.04042. doi:10.48550/arXiv.2504.04042 arXiv:2504.04042
[25] K. Zhang, H. Yang, X. Tang, W. Yu, J. Xu, Beyond guilt: Legal judgment prediction with trichotomous reasoning, arXiv preprint arXiv:2412.14588 (2024). URL: https://arxiv.org/abs/2412.14588.
[26] H. Westermann, A demonstration of a semi-structured legal reasoning framework, in: Proceedings of the 20th International Conference on Artificial Intelligence and Law, ACM, 2025, pp. 504–505.
[27] P. Minh Nguyen, T. Ha Nguyen, M. Zin, K. Satoh, Data augmented pipeline for legal information extraction and reasoning, in: Proceedings of the 20th International Conference on Artificial Intelligence and Law, 2025, pp. 481–482.
[28] F. Bomfim, J. Neto, A. Neves, V. Furtado, S. Silva, V. Pinheiro, Automating legal reasoning with generative ai: A framework for judicial decision support, in: Proceedings of the 20th International Conference on Artificial Intelligence and Law (ICAIL '25), ACM, 2025, pp. 229–238. doi:10.1145/3769126.3769244
[29] J. Herrmann, F. Breitinger, KI in der Rechtsprüfung: Warum XAI allein nicht ausreicht, SIAK-Journal: Zeitschrift für Polizeiwissenschaft und polizeiliche Praxis (2026) 85–101. doi:10.7396/2026_1_G
[30] G. Michelet, J. Schneider, A. Withanage, F. Breitinger, Hey gpt-oss, looks like you got it–now walk me through it! an assessment of the reasoning language models chain of thought process for digital forensics, Forensic Science International: Digital Investigation 56 (2026) 302052.