Towards Neuro-symbolic Causal Rule Synthesis, Verification, and Evaluation Grounded in Legal and Safety Principles

Zainab Rehan
Hasso Plattner Institute
University of Potsdam
Potsdam, Germany
zainab.rehan@uni-potsdam.de

Christian Medeiros Adriano
Hasso Plattner Institute
University of Potsdam
Potsdam, Germany
christian.adriano@hpi.de

Sona Ghahremani
Hasso Plattner Institute
University of Potsdam
Potsdam, Germany
sona.ghahremani@hpi.de

Holger Giese*
Hasso Plattner Institute
University of Potsdam
Potsdam, Germany
holger.giese@hpi.de

ABSTRACT

Rule-based systems remain central in safety-critical domains but often struggle with scalability, brittleness, and goal misspecification. These limitations can lead to reward hacking and failures in formal verification, as AI systems tend to optimize for narrow objectives. In previous research, we developed a neuro-symbolic causal framework that integrates first-order logic abduction trees, structural causal models, and deep reinforcement learning within a MAPE-K loop to provide explainable adaptations under distribution shifts. In this paper, we extend that framework by introducing a meta-level layer designed to mitigate goal misspecification and support scalable rule maintenance. This layer consists of a Goal/Rule Synthesizer and a Rule Verification Engine, which iteratively refine a formal rule theory from high-level natural-language goals and principles provided by human experts. The synthesis pipeline employs large language models (LLMs) to: (1) decompose goals into candidate causes, (2) consolidate semantics to remove redundancies, (3) translate them into candidate first-order rules, and (4) compose necessary and sufficient causal sets. The verification pipeline then performs (1) syntax and schema validation, (2) logical consistency analysis, and (3) safety and invariant checks before integrating verified rules into the knowledge base. We evaluated our approach with a proof-of-concept implementation in two autonomous driving scenarios. Results indicate that, given human-specified goals and principles, the pipeline can successfully derive minimal necessary and sufficient rule sets and formalize them as logical constraints. These findings suggest that the pipeline supports incremental, modular, and traceable rule synthesis grounded in established legal and safety principles.

KEYWORDS

Rule synthesis, Neuro-symbolic methods, Software verification, Causality, Goal specification

*IEEE and ACM Member

Proc. of the Neurosymbolic eXplainable Trustworthy Systems Workshop (NEXUS 2026), Bakirtzis, Fridovich-Keil , Kakas, Theodorou, Li, McKinlay (eds.), May 25 – 26, 2026, Paphos, Cyprus,https://nexus.telecom-paris.fr. 2026.

1 INTRODUCTION

Context – Rule-based systems [31] are specified and constructed from existing domain knowledge about desired and undesired (forbidden, unsafe) behavior, often as logical if–then statements. Rule-based approaches remain central in many safety- and mission-critical contexts because rules make system behavior observable at runtime and open to formal verification [21, 34].

In self-adaptive and autonomic systems, the MAPE-K feedback loop [26] represents monitoring, analysis, planning, and execution as activities over a shared knowledge base (of rules and representation models). These systems often rely on runtime enforcement components that supervise primary controllers and trigger predefined mitigation actions when rules are violated [9, 34].

Challenges – However, rule-based and expert systems are widely recognized for their brittleness and limited scalability. As new requirements emerge, even minor changes can cause cascading revisions across numerous interdependent rules. Managing and tracking these evolving rules places a significant cognitive and organizational burden on human experts, often becoming a central bottleneck and increasing the risk of unexpected failures [19, 34]. Classic analyses of earlier expert systems highlight the exponential increase in maintenance effort and the well-known "knowledge acquisition bottleneck." In practice, large rule sets are inherently difficult to extend or adapt without unintentionally disrupting existing behaviors [19, 34]. These issues were prominent in large, early AI expert-system projects, which helped motivate later shifts toward methods that combine statistical learning with more scalable forms of knowledge representation [29, 34].

Approach – We extend a prior neuro-symbolic causal framework for self-adaptive, learning-enabled systems with a meta-level synthesis and verification layer that incrementally refines the governing rule theory. A Goal/Rule Synthesizer uses large language models to decompose high-level, natural-language goals and principles into candidate causes, consolidate their semantics, and translate them into candidate first-order rules while identifying necessary and sufficient cause sets. A complementary Rule Verification Engine then enforces syntax and schema correctness, logical consistency, and safety and invariant constraints before integrating only verified rules into the knowledge base. Together, these components

realize an incremental, modular, and traceable pipeline from human-specified legal and safety principles to formally verified rule sets, demonstrated on autonomous driving scenarios.

Evaluation – We present an application in the domain of autonomous driving, where safety goals are systematically decomposed into subgoals and translated into corresponding rules. The mapping between goals and rules is realized via simulated abduction that produces explanations of an effect from its causes, while rule refinement is performed through deductive reasoning, de-duplicating rules and combining them into necessary and sufficient sets. We evaluated our approach with a proof-of-concept implementation in two autonomous driving scenarios, showing that, given human-specified goals and principles, the pipeline can successfully derive minimal necessary and sufficient rule sets and formalize them as logical constraints. These findings suggest that the pipeline supports incremental, modular, and traceable rule synthesis grounded in established legal and safety principles.

Contributions – By structuring rule maintenance as an incremental, meta-level synthesis-and-verification loop, the approach directly attacks the scalability and brittleness problems of classic rule-based systems. First, high-level goals are decomposed into modular sets of necessary and sufficient rules per goal, so extensions localize to specific goal-linked modules instead of triggering cascading revisions across a monolithic rule base. Second, semantic consolidation, de-duplication, and explicit traceability from each rule back to its originating goal and principles reduce redundancy and make large rule sets easier to understand, audit, and evolve. Third, the Rule Verification Engine systematically filters new rules through syntax, consistency, and safety checks before reintegration, preventing brittle, ad hoc patches and enabling controlled growth of the theory over time. Reproduction package is available at [5].

2 STATE OF THE ART

Goal Decomposition with LLMs – LLMs can decompose complex goals into structured subgoals for planning and assistance. Multi-step plans are generated, refined, and evaluated [2]. SGA-ACR [13] produces verifiable sub-goal chains for RL agents, while DELTA [32] leverages scene graphs for efficient long-horizon task decomposition. Hierarchical LLM agents improve tractability and interpretability [22], and human-centered methods [40] learn decompositions that enhance non-expert performance on complex programming tasks.

Abduction and Rule Synthesis – Abductive reasoning generates explanatory hypotheses and candidate symbolic rules [3], providing a principled path from observations to minimal assumptions and compact rule-like hypotheses [37]. Statistical learning links, via inductive logic programming and statistical relational learning, allow frequent patterns or abductive explanations to become human-interpretable if–then rules [23, 38]. Contemporary neuro-symbolic approaches use abduction as structured prior knowledge to guide neural learners, improving sample efficiency. Extracted hypotheses can be refined and pruned to produce robust, generalizable adaptation rules, bridging observed outcomes and formalized reusable knowledge for self-adaptive systems [3, 23, 38].

Causal-Neuro-Symbolic Reasoning - Causal neuro-symbolic AI combines the strengths of causal inference, symbolic reasoning, and deep learning to produce models that are both adaptable and explanatory [25]. In such hybrid frameworks, symbolic structures (e.g., causal graphs or abduction trees) provide interpretable scaffolding for interventions and counterfactual reasoning while neural components supply flexible function approximation for perception and policy learning. Recent work demonstrates how causal abstractions can be learned or exploited by agents to accelerate adaptation and to ground symbolic recovery strategies in measurable causal effects [25, 28]. Applications in multi-agent reinforcement learning show further promise: transferable macro-actions or recovery primitives can be represented as compact causal rules that capture cause & effect pathways linking failures to corrective actions, enabling agents to exchange and re-use causal knowledge across contexts [28, 42]. Integrating such transferable causal experiences into symbolic abduction structures enriches the abductive search space with empirical, distributed evidence, thereby improving both the quality of synthesized rules and their generalizability across agents and environments.

Knowledge Representation in Rule-based Systems Rule-based systems and first-order logic support complex and commonsense reasoning but often incur high computational costs [27, 33]. To improve scalability, global theories are partitioned into context-specific modules, enabling local reasoning while maintaining overall consistency [6, 11, 35]. Techniques such as monotonic and non-monotonic reasoning, syntax splitting, modular answer set programming, and context-oriented logics support efficient reasoning and inter-context communication [6, 11, 27, 33, 35]. Probabilistic graphical models and related network structures integrate local results into coherent global knowledge, balancing expressiveness with computational manageability [28].

3 RESEARCH PROBLEMS

Humans often state goals imprecisely, while AI systems pursue them with strict literalism. This can lead to unintended outcomes, because AI models may exploit loopholes to technically complete a task rather than follow the user's true intent. Moreover, human communication relies on shared context that AI systems typically lacks. Therefore, vague instructions are particularly hazardous, because cannot be easily confirmed and disambiguated by external sources.

These issues manifest in concrete failure modes such as formal proof cheating and reward hacking. Formal proof cheating occurs when an AI produces a proof that technically satisfies formal criteria but does so by altering axioms or definitions, ignoring the genuine mathematical intent [7]. Reward hacking [39] occurs when an AI system exploits imperfections in a proxy reward function, achieving high measured performance while violating the underlying goal.

Goal misspecification is therefore a central problem in AI safety [4], especially when formal objectives diverge from the informal intent of system designers. Goodhart's law in machine learning further predicts that proxy metrics tend to degrade when heavily optimized [30, 41], whether the goals are misspecified or merely underspecified.

Even when designers choose the right high-level goals, they may still encode them in ways that are too narrow or incomplete, leading AI systems to optimize the measurable formal metric rather than the

broader human objective it was meant to approximate. Essentially, humans rely on vague, context-dependent, and partly unconscious goals [8, 14, 24]. Conversely, AI systems require precise, formal, and loophole-free specifications to behave as intended [4].

Accordingly, humans need support tools that help them articulate more precise and robust specifications that reduce misinterpretation or manipulative compliance by AI systems. This support must go beyond traditional verification and validation. In this paper, we propose a vision of goal-specification assistance centered on explicit rules and their systematic composition. We explore these fundamental challenges through two research questions.

- **RP.1** – What are the current capabilities of generative AI to support the synthesis of rules in a principled way that minimizes redundancies and ambiguities while maintaining traceability to system goals? **Our approach** – Establish operations that refine rules via deduction and explain their effect via abduction.
- **RP.2** – How can rule synthesis be automatically evaluated in incremental and modular ways? **Our approach** – Search for sufficient and necessary rule sets that satisfy a given goal.

General Insight – We consider goals and subgoals as effects of rules (causes) and their preconditions (controls). Moreover, will rely on causal representations and fundamental domain laws to constrain reasoning about rules and their effects with respect to system adaptations that preserve performance and safety [1, 16].

4 APPROACH

The proposed approach builds on our previous work [1], which is a neuro-symbolic (NeSy) causal reasoning framework for learning-enabled self-adaptive systems [15, 18] in particular when operating under distribution shifts [17] and under safety-critical goals [20] —see Section 4.1. As depicted in Figure 1, the approach comprises a *Goal/Rule Synthesizer* and a *Rule Verification Engine* on top of the NeSy framework—see Section 4.2.

4.1 NeSy Causal Framework

The Neuro-Symbolic (NeSy) causal framework integrates three complementary representations: (1) Symbolic Abduction Trees (SAT) grounded in first-order logic (FOL) to formally encode domain knowledge, system constraints, and adaptation rules, enabling logical abduction to explain observed violations; (2) a structural causal model (SCM) that represents dependencies between environment and system variables and supports intervention and counterfactual reasoning; and (3) a deep reinforcement learning (DRL) policy responsible for operational decision-making–see Figure 1 for a reference. The three representation spaces are embedded within the MAPE-K (Monitor-Analyze-Plan-Execute) feedback loop as the reference model for self-adaptive systems [26]. The process begins with monitoring execution traces from the DRL agent to detect distribution shifts and safety-constraint violations. In the analysis and planning phases, FOL-based abductive reasoning identifies candidate explanations, while the causal model evaluates intervention effects and selects minimal, high-explanatory-power configuration changes. These interventions guide knowledge transfer and warm-start retraining of the DRL agent, whose updated behavior produces new traces that refine both symbolic and causal knowledge. The

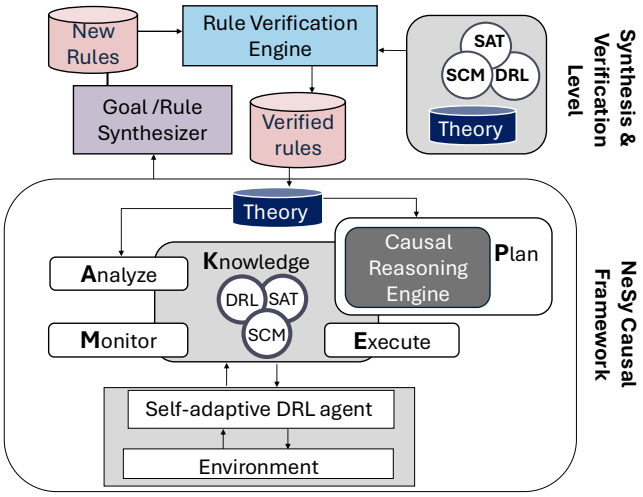

Figure 1: Extending NeSy Causal Framework with Rule-synthesis and Verification

framework thus interleaves logical reasoning, causal inference, and learning to enable explainable, constraint-aware adaptation rather than reactive retraining.

4.2 Synthesis & Verification Level

The *Synthesis & Verification Level* extends the original NeSy framework by introducing a meta-self-aware synthesis and verification layer on top of the operational adaptation cycle. In contrast to the earlier framework, the outputs of causal explanation are no longer used solely for policy adaptation. Instead, anomaly reports are forwarded to the higher synthesis & verification level. This level contains a *Goal/Rule Synthesizer* and a *Rule Verification Engine* that collectively update the theory that comprises domain knowledge, system constraints, and adaptation rules. Verified rules are reintegrated into the knowledge base, thereby modifying future analysis and planning steps. The architecture, therefore, introduces a closed meta-loop in which the system adapts not only its behavior but also its governing symbolic FOL theory, enabling progressive self-improvement and increased explainability.

Goal/Rule Synthesizer – The pipeline (see Figure 2) receives as input the high-level goals and governing principles provided by a human expert expressed in natural language (natural-language based, NLB) rather than as formal constraints, following the detection of a drift or anomaly. The first step uses a large language model (LLM) to decompose each high-level goal into a set of lower-level candidate causes, representing concrete conditions, behaviors, or system properties that collectively could realize or justify the intended goal. These generated causes are then reprocessed by the LLM in a semantic consolidation step, where overlapping, redundant, or semantically equivalent causes are identified and merged. This normalization produces a consistent set of unique causes, ensuring that the emerging theory avoids duplication and ambiguity. In the next step, the consolidated causes are again provided to the LLM for symbolic translation, mapping natural-language concepts into candidate logical predicates and relations and producing a set of new, unverified rules (causes) expressed in FOL. The resulting logical rules enable formal reasoning during the Rule Verification pipeline,

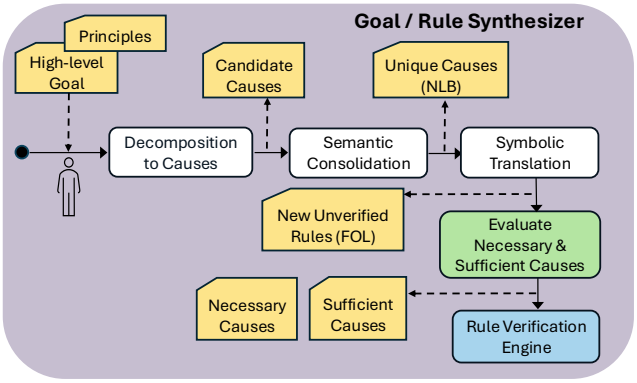

Figure 2: Pipeline for Goal/Rule Synthesizer

e.g., using a verification solver such as Z3 [10]. The goal/rule synthesizer pipeline analyzes combinations of causes to determine subsets of necessary and sufficient conditions that adequately justify the original synthesized Goal (more details in Section 4.3). The output of this pipeline is a set of newly synthesized but still unverified logical rules representing intended behavior derived jointly from empirical evidence and human intent.

Rule Verification Engine - The synthesized rule candidates are forwarded to the Rule Verification Engine, where formal consistency and safety validation are performed to ensure that synthesized rules can be safely incorporated into the system theory–see Figure 3. The layered pipeline consists of (1) syntactic and schema validation, ensuring that generated rules conform to the logical language and domain ontology together with semantic grounding checks, confirming that predicates and variables correspond to valid system entities; (2) logical consistency analysis, where candidate rules are evaluated against the existing FOL knowledge base to detect contradictions; and (3) safety and invariant verification, which formally checks whether the rules preserve required system properties. The pipeline employs an automated reasoning tool to perform satisfiability and entailment checks. Only rules that satisfy all verification stages are promoted to a verified theory and committed to the knowledge repository. This verified knowledge is then fed back into the adaptive loop, ensuring that future adaptations are guided by formally verified, explainable principles rather than purely data-driven updates.

4.3 Illustrative Scenario for Goal/Rule Synthesis Pipeline

We demonstrate the Goal/Rule Synthesis Pipeline based on an application in the domain of autonomous driving.

Decomposition to Causes - The high-level goal (effect) represents a scenario derived from everyday driving tasks. The objective is to generate candidate causes that explain how the effect may occur. For example, the effect *"respond correctly to a sudden obstacle on the road"* may be explained by the cause *"the driver applies emergency braking."* The LLM decomposes the effect into candidate causes required to achieve it while ensuring safety and compliance with traffic regulations. The synthesis process is strictly constrained to limit hallucinations and unrelated reasoning, resulting in causes

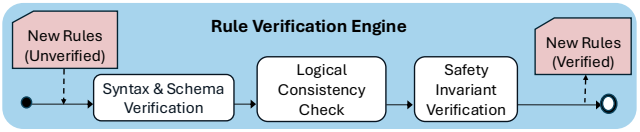

Figure 3: Pipeline for Rule verification Engine

that are logically consistent with the effect and with the defined constraints.

Semantic Consolidation – Although the focus is restricted at the first step, there is still a need to merge causes that refer to similar concepts to reduce general statements that overlap into a set of unique causes. The semantic consolidation step is carried out to combine causes that describe the same underlying condition. *"Driver maintains control of the vehicle"* and *"Driver is aware of surrounding traffic"* are combined into a cause as *"Driver maintains control of the vehicle and is aware of surrounding traffic"*.

Symbolic Translation – After generating unique causes, each cause is translated from natural language to formal symbolic (FOL) form to enable structured reasoning and systematic subset evaluation. Although this step follows semantic unification in Figure 2, it can be applied at any stage without altering content. Subsequent analysis uses the natural language causes for stronger LLM reasoning, with FOL forms used only in the rule verification pipeline Figure 3. This decision is based on the observation that LLMs demonstrate stronger reasoning performance when operating on natural language inputs. To ensure consistency and avoid ambiguity, a formal grammar is defined in the prompt, and the model is instructed to strictly follow it. Each cause is thus mapped to a structured logical form that preserves its meaning while remaining machine-interpretable.

Alongside the FOL rule, the LLM also provides a brief explanation of its translation, detailing how the natural language condition was interpreted and which predicates and operators were chosen. This enhances transparency, verifies semantic alignment, and improves interpretability of the symbolic mapping.

Evaluate Necessary & **Sufficient Causes** – After obtaining a set of unique causes, the next step involves assessing the necessity and sufficiency of each cause with respect to the main goal (effect). We break this down into three steps described below:

(i) *Individual Necessity Evaluation* is concerned with whether each cause is essential and provides a justification, citing specific legal or safety conditions that would be violated if absent. Each cause is assessed independently, and the output is a structured list recording the cause, its necessity, and the rationale.

(ii) *Subset Necessity Evaluation* identifies causes that are essential only in combination. Starting with the full set, causes are removed one at a time, and the feasibility of the effect is confirmed. A cause is deemed necessary if after its removal, the effect is judged unachievable. A removed cause or set of causes is recorded as a minimal necessary subset if no smaller subset produces the same effect. To improve efficiency, the candidate subsets containing already identified minimal subsets are skipped, and the outcomes of previous evaluations are cached. This bottom-up approach ensures that all truly minimal necessary combinations are captured while reducing computational effort.

(iii) *Minimal Sufficient Set Evaluation* aims at identifying the subsets of causes that are individually sufficient to produce a given effect. We start with single causes and test their sufficiency, incrementally adding additional causes if the effect does not occur. This process continues until a sufficient subset is found, and is repeated across all relevant combinations. To reduce computational effort, supersets of already identified sufficient subsets are skipped, as sufficiency is preserved under addition. Once all minimal sufficient sets are determined, they can be combined into a necessary-and-sufficient set, which guarantees the effect and is required for its occurrence. This set unifies all alternative minimal sufficient causes, capturing every pathway to the effect within a single comprehensive framework.

5 PROOF OF CONCEPT

Experimental Setup - We design a controlled experiment to show how Large Language Models (LLMs) can help automate parts of the Goal/Rule Synthesizer pipeline illustrated in Figure 2. The objective of this setup is to evaluate realistic driving scenarios in which an intended effect (Goal) must be explained by a set of underlying causes (Rules). The intent is not to verify causality in a statistical sense, but to show how LLMs can be guided to perform structured abductive and deductive reasoning. The scope of the setup is constrained by a predefined set of legal and safety regulations that define the boundaries within which the synthesis and reasoning process operates.

5.1 Implementation Overview

The pipeline is implemented by leveraging the OpenAI GPT-4o Mini model [36] to perform all processing tasks. Step by step, the process transforms an human stated goal (effect) description into a set of candidate causes. The process begins with cause generation, followed by consolidation of overlapping causes, then individual evaluation, and finally, subset analysis for necessity and sufficiency. This iterative design ensures that the reasoning is incremental, traceable, and easy to interpret at every step. The legal and safety regulations that guide the process are defined prior to the first stage. These regulations serve as the sole reference for the model throughout the pipeline. For details on how these principles were sourced and compiled–see Section 5.2.

5.2 Data Sources

Legal Principles: The legal Principles used in this work are derived from a formalized set of German traffic regulations designed for machine interpretability. These regulations define rules such as maintaining vehicle control, adhering to speed limits, and keeping safe distances. We selected the most relevant laws from this formalized research [12]. To cover broader scenarios encountered in everyday driving, we then extended the set using generative AI, specifically ChatGPT. This ensures that the legal knowledge can guide the model in a variety of realistic situations while remaining interpretable and precise.

Safety Principles: The safety principles capture physical principles of vehicle motion. Initially, we developed a basic set of rules representing logical and physics-based constraints. Examples include limits on friction, constraints on braking distance, and conditions that increase the risk of losing traction at high speeds. To expand coverage, ChatGPT was used to generate additional rules representing general principles, e.g., as "collision_if_obstacles", which states that a vehicle may collide if obstacles are present. This expanded set allows the model to account for a wider variety of practical driving scenarios while remaining grounded in physical feasibility.

Grammar and Symbolic Rule Language: The symbolic grammar used for translating natural language causes into formal rules is based on the same formalization framework presented in [12]. The original predicate set is extended to incorporate additional predicates related to fundamental speed and friction properties in order to capture physics-based safety constraints. Moreover, the grammar specifies the allowed rule forms, comparison operators, and logical operators that may be used in rule construction.

All reasoning carried out by the LLM is grounded in these curated laws, which serve as the primary reference for evaluating causes. External knowledge or assumptions beyond this curated set are not used. This guarantees that all cause & effect evaluations remain valid, consistent, and reproducible. A complete listing of all legal and safety laws used is provided in the Appendix [5].

5.3 Results

We analyze two illustrative scenarios to demonstrate the outputs of the pipeline. For each scenario, candidate causes were generated, semantically consolidated, individually evaluated for necessity, and subsequently analyzed to identify minimal necessary subsets and sufficient subsets..

5.3.1 Scenario 1: Successfully Merge into Heavy Traffic.

Decomposition and Consolidation. Eight candidate causes were initially generated. After semantic de-duplication, four unique consolidated causes remained:

(1) Driver maintains control of the vehicle and is aware of surrounding traffic.
(2) Vehicle is traveling at a speed that allows for safe merging and adheres to traffic laws regarding merging.
(3) Sufficient distance is kept from other vehicles to merge safely, and no vehicles are overtaking on the right.
(4) Traffic conditions allow for merging without impeding flow, and no sudden obstacles in the merging path.

Individual Necessity Evaluation. Each consolidated cause was evaluated independently against the relevant safety and legal constraints. Only the first two were classified as individually necessary. This approach makes its reasoning both interpretable and actionable, providing a clear understanding of the underlying dynamics.

- *Driver control and awareness:* Without maintaining control and awareness, safe merging is not possible, violating safety constraints related to vehicle stability and collision avoidance.
- *Vehicle speed within safe limits:* Appropriate speed is essential for lawful and safe merging. Deviations from safe speed ranges can violate legal constraints and create unsafe conditions.

Minimal Necessary Sets. Four minimal necessary sets were identified. Each set represents a core condition that cannot be removed without rendering the effect impossible:

- *Necessary Set 1:* Driver maintains control of the vehicle and is aware of surrounding traffic.
- *Necessary Set 2:* Vehicle is traveling at a speed that allows for safe merging and adheres to traffic laws regarding merging.
- *Necessary Set 3:* Sufficient distance from other vehicles to merge safely and no vehicles are overtaking on the right.
- *Necessary Set 4:* Traffic conditions allow for merging without impeding flow and no sudden obstacles in the merging path.

Minimal Sufficient Sets. Sufficient sets represent combinations of causes that guarantee successful merging. One minimal sufficient set was identified:

- *Sufficient Set 1:*
  - Driver maintains control of the vehicle and is aware of surrounding traffic.
  - Vehicle is traveling at a speed that allows for safe merging and adheres to traffic laws regarding merging.
  - Sufficient distance from other vehicles to merge safely and no vehicles are overtaking on the right.
  - Traffic conditions allow for merging without impeding flow and no sudden obstacles in the merging path.

These findings highlight an important structural property: while some causes are not individually necessary, they become indispensable within certain minimal necessary sets. Moreover, every minimal necessary cause is included within sufficient sets that ensure the effect occurs, illustrating how necessary and sufficient conditions are systematically linked.

5.3.2 Scenario 2: Maintain a Constant Speed on a Highway Segment.

Decomposition and Consolidation. Eight candidate causes were initially generated. After deduplication, six unique causes remained:

(1) Driver maintains vehicle control and is attentive and responsive to road conditions.
(2) Vehicle speed is within legal limits and above minimum required speed.
(3) Sufficient friction between tires and road.
(4) No obstacles on the highway segment.
(5) No sudden changes in traffic conditions.
(6) No emergency situations requiring sudden braking.

Individual Necessity Evaluation. Each cause was evaluated individually against safety and legal constraints. The first three were classified as individually necessary:

- *Driver control and attentiveness:* Without stable control and responsiveness, maintaining constant speed violates vehicle control safety requirements.
- *Vehicle speed within limits:* Operating outside legal speed bounds violates regulatory constraints and invalidates the effect.
- *Sufficient friction:* Adequate tire–road friction is required to maintain speed safely and preserve traction.

Minimal Necessary Sets. Three minimal necessary sets were identified, corresponding to the causes that were also individually necessary:

- *Necessary Set 1:* Driver maintains vehicle control and is attentive and responsive to road conditions.
- *Necessary Set 2:* Vehicle speed is within legal limits and above minimum required speed.
- *Necessary Set 3:* Sufficient friction between tires and road.

Minimal Sufficient Sets. Two minimal sufficient sets were identified:

- *Sufficient Set 1:*
  - Driver maintains vehicle control and is attentive and responsive to road conditions.
  - Vehicle speed is within legal limits and above minimum required speed.
  - Sufficient friction between tires and road.
  - No obstacles on the highway segment.
- *Sufficient Set 2:*
  - Driver maintains vehicle control and is attentive and responsive to road conditions.
  - Vehicle speed is within legal limits and above minimum required speed.
  - Sufficient friction between tires and road.
  - No sudden changes in traffic conditions.

These results demonstrate that, while certain causes may not be required on their own to produce the effect, they play a crucial role when combined with other factors. In these sufficient combinations, their presence helps ensure that the effect reliably occurs. This highlights how individual causes can contribute indirectly, supporting the overall outcome even if they are not strictly necessary in isolation.

5.3.3 Symbolic Translation. To illustrate the formal translation step, each unique cause was converted into a symbolic rule for both scenarios using the predefined grammar. The following examples show how natural language conditions are mapped into formally constrained logical representations.

"Driver maintains control of the vehicle"

$$\forall x(\neg collide(x) \leftarrow sd_front(x) \wedge sd_rear(x) \wedge \neg lane_change(x))$$

This rule formalizes vehicle control through safe longitudinal distances (sd) and the absence of destabilizing lane changes, ensuring collision avoidance.

"Sufficient distance from other vehicles to merge safely"

$$\forall x(sd_front(x) \wedge sd_rear(x) \leftarrow \neg dense(x))$$

Low traffic density implies that safe distances (sd) can be maintained both in front of and behind the vehicle.

These examples demonstrate how rich driving conditions are systematically reduced to formally constrained logical rules within the reasoning framework. A key takeaway is that the LLM can interpret nuanced, context-dependent descriptions and convert them into precise symbolic forms. It effectively bridges human-readable reasoning with formal, machine-interpretable logic, keeping both meaning and structure intact. For a comprehensive presentation of the generated candidate causes, intermediate evaluation steps, full subset analyses, and complete symbolic translations for both

scenarios, refer to the Appendix [5], where detailed end-to-end examples of the pipeline execution are provided.

6 DISCUSSION

The pipeline provides a structured way to analyze cause & effect relationships in driving scenarios, directly informing how Generative AI can support the synthesis of rules in a principled, traceable manner (**RP.1**). Each stage contributes unique insights into the reasoning process. The initial decomposition highlights the potential factors that could lead to the effect, ensuring that no obvious candidate cause is missed and that high-level goals are systematically unpacked into concrete rule candidates. De-duplication then reduces redundancy, merging overlapping causes while preserving meaning, which directly addressing the need to minimize ambiguities and redundancies in synthesized rules. This simplifies the reasoning and prevents inflated or repetitive outputs, thereby improving the clarity and manageability of the rule space at scale.

Individual necessity evaluation assesses each cause in isolation and shows which factors are critical on their own, providing a first approximation of how candidate rules relate to system goals (**RP.1**). However, this step alone cannot capture conditional relationships. Subset evaluation addresses this by considering combinations of causes. Interestingly, some causes initially marked as not necessary were required when combined with other factors. This demonstrates the importance of examining interactions and conditional dependencies rather than relying solely on isolated assessments and shows how rule evaluation must account for contextual dependencies to remain faithful to the desired specified goals (**RP.2**).

Sufficiency analysis further emphasizes flexible pathways to the effect and operationalizes the search for sufficient and necessary sets of rules that satisfy a given goal (**RP.2**). Minimal sufficient sets include the necessary causes and may also incorporate additional factors to guarantee the effect. This reflects real-world complexity, where multiple combinations of conditions can produce the same outcome similar to Scenario 2. Some causes that seemed irrelevant individually were shown to be sufficient only in combination, highlighting the subtlety of causal relationships. Together, the necessity and sufficiency analyses provide an incremental and modular way to evaluate rules: incremental, because causes and subsets are assessed stepwise; modular, because minimal sets can be associated with specific goals and reused as distinct rule components (**RP.2**).

Despite its strengths in addressing **RP.1** and **RP.2**, the approach has limitations. The model functions as a black box, so the internal reasoning is not fully transparent. Moreover, there is a reliance on the predefined set of regulations, which may not cover all possible scenarios. Additionally, the outputs are highly sensitive to how the prompts are formulated, which can influence the adaptability of the model's responses. These factors constrain the completeness and robustness of the synthesized rule sets and their evaluations and indicate that principled rule synthesis and modular evaluation still depend heavily on the quality and coverage of the underlying knowledge base (**RP.1**, **RP.2**).

However, these challenges point toward opportunities for improvement. Expanding the set of rules, introducing probabilistic reasoning, or incorporating multiple model perspectives could enhance robustness. Iterative refinement of prompts and providing feedback to the model can further reduce errors. Such extensions would not only broaden the domain coverage but also strengthen the principled nature of rule synthesis and the reliability of incremental, modular rule evaluation with respect to evolving goals (**RP.1**, **RP.2**). Overall, the pipeline demonstrates a form of abductive reasoning. It identifies likely causes, evaluates their necessity and sufficiency, and uncovers conditional dependencies. The structured, stepwise process makes the reasoning traceable and interpretable while capturing complex interactions between factors, thereby providing concrete evidence that LLM-based pipelines can support goal-based, legally and safety-informed rule synthesis (**RP.1**) and in a systematic, incremental, and modular way (**RP.2**).

7 CONCLUSION AND FUTURE WORK

The proposed framework demonstrates how neuro-symbolic causal reasoning, combined with large language models, can support incremental and modular rule synthesis grounded in legal and safety principles. By decomposing high-level goals into candidate causes, consolidating them semantically, translating them into formal rules, and then analyzing necessary and sufficient cause sets, the pipeline yields traceable and interpretable rule sets that capture complex causal interactions in domains such as autonomous driving. The subsequent verification stages—covering syntax and schema checks, logical consistency, and safety and invariant preservation—ensure that only rules compatible with an existing knowledge base and domain constraints are integrated back into the system, thereby strengthening reliability and explainability across the MAPE-K loop. Together, these elements illustrate how abductive and deductive reasoning can be orchestrated to reduce brittleness, improve goal alignment, and provide a principled pathway from natural-language specifications to formally verified rule theories.

In future work, we plan to deepen the integration between causal models and rule synthesis to derive more expressive sets of necessary and sufficient rules, and to broaden the coverage of domain principles beyond the current set of traffic regulations. We also plan to incorporate automated solvers to detect and resolve conflicts between newly synthesized rules and existing knowledge, enabling systematic management of inconsistencies at the theory level. This will support the design of richer feedback loops that jointly exploit formal verification, causal path analysis, and effect estimation, with the goal of informing more robust synthesis, composition, and adaptation strategies in safety-critical, learning-enabled systems.

REFERENCES

[1] Christian Medeiros Adriano, Sona Ghahremani, Finn Kaiser, and Holger Giese. 2025. Neuro-symbolic causal reasoning for cautious self-adaptation under distribution shifts. In *2025 IEEE International Conference on Autonomic Computing and Self-Organizing Systems (ACSOS)*. IEEE, 88–99.

[2] Mohamed Aghzal, Erion Plaku, Gregory J Stein, and Ziyu Yao. 2025. A survey on large language models for automated planning. *arXiv preprint arXiv:2502.12435* (2025).

[3] Alicia Aliseda. 2017. The Logic of Abduction: An Introduction. In *Springer Handbook of Model-Based Science*. Springer, Cham, 1–24. Chapter on abduction and hypothesis generation.

[4] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. 2016. Concrete Problems in AI Safety. *arXiv preprint arXiv:1606.06565* (2016).

[5] HPI System Analysis and Modeling Group. 2026. Goal-Based Rule Synthesis. https://github.com/hpi-sam/goal-based-rule-synthesis. GitHub repository.

[6] Christoph Beierle, Lars-Phillip Spiegel, Jonas Haldimann, Marco Wilhelm, Jesse Heyninck, and Gabriele Kern-Isberner. 2024. Conditional splittings of belief bases

and nonmonotonic inference with c-representations. In *Proceedings of the 21st International Conference on Principles of Knowledge Representation and Reasoning* (Hanoi, Vietnam) *(KR '24)*. Article 10, 11 pages. https://doi.org/10.24963/kr.2024/10

[7] Sascha Brodsky. 2025. The mathematicians teaching AI to reason. *IBM Think* (October 2025). https://www.ibm.com/think/news/mathematicians-teaching-ai-to-reason Published 13 October 2025, updated 23 October 2025.

[8] Jaime J Castrellon, Jacob S Young, Linh C Dang, Christopher T Smith, Ronald L Cowan, David H Zald, and Gregory R Samanez-Larkin. 2021. Dopamine biases sensitivity to personal goals and social influence in self-control over everyday desires. *bioRxiv* (2021), 2021–09.

[9] Erika F. de Almeida, Marcio V. Vieira, and Rogério de Lemos. 2017. MAPE-K based guidelines for designing reactive and proactive self-adaptive systems. In *Proceedings of the 12th International Symposium on Software Engineering for Adaptive and Self-Managing Systems*. IEEE, 97–107.

[10] Leonardo De Moura and Nikolaj Bjørner. 2008. Z3: An efficient SMT solver. In *International conference on Tools and Algorithms for the Construction and Analysis of Systems*. Springer, 337–340.

[11] Thomas Eiter, Michael Fink, Giovambattista Ianni, Thomas Krennwallner, Christoph Redl, and Peter Schüller. 2016. A model building framework for answer set programming with external computations. *Theory and Practice of Logic Programming* 16, 4 (2016), 418–464.

[12] Klemens Esterle, Luis Gressenbuch, and Alois Knoll. 2020. Formalizing Traffic Rules for Machine Interpretability. In *Proceedings of the 3rd IEEE Connected and Automated Vehicles Symposium (CAVS)*. Institute of Electrical and Electronics Engineers (IEEE), Victoria, B.C., Canada. https://doi.org/10.1109/CAVS51000.2020.9334599

[13] Shanwei Fan, Bin Zhang, Zhiwei Xu, Yingxuan Teng, Siqi Dai, Lin Cheng, and Guoliang Fan. 2025. Subgoal Graph-Augmented Planning for LLM-Guided Open-World Reinforcement Learning. *arXiv preprint arXiv:2511.20993* (2025).

[14] Klaus Fiedler and Mandy Hütter. 2014. The Limits of Human Information Processing in Goal Pursuit: Motivational Influences on Judgment and Decision Making. *Current Directions in Psychological Science* 23, 3 (2014), 163–170.

[15] Sona Ghahremani, Christian Medeiros Adriano, and Holger Giese. 2018. Training Prediction Models for Rule-Based Self-Adaptive Systems. In *International Conference on Autonomic Computing (ICAC)*. 187–192. https://doi.org/10.1109/ICAC.2018.00031

[16] Sona Ghahremani and Holger Giese. 2021. Hybrid planning with receding horizon: A case for meta-self-awareness. In *2021 IEEE International Conference on Autonomic Computing and Self-Organizing Systems Companion (ACSOS-C)*. IEEE, 131–138.

[17] Omid Gheibi and Danny Weyns. 2024. Dealing with Drift of Adaptation Spaces in Learning-based Self-Adaptive Systems Using Lifelong Self-Adaptation. *ACM Trans. Auton. Adapt. Syst.* 19, 1, Article 5 (feb 2024), 57 pages. https://doi.org/10.1145/3636428

[18] Omid Gheibi, Danny Weyns, and Federico Quin. 2021. Applying machine learning in self-adaptive systems: A systematic literature review. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)* 15, 3 (2021), 1–37.

[19] David Perry Greene. 1987. Automated knowledge acquisition: Overcoming the expert system bottleneck. In *Proceedings of the Eighth International Conference on Information Systems (ICIS 1987)*. Association for Information Systems, 155–167.

[20] Joachim Haensel, Christian Medeiros Adriano, Johannes Dyck, and Holger Giese. 2020. Collective risk minimization via a bayesian model for statistical software testing. In *Proceedings of the IEEE/ACM 15th International Symposium on Software Engineering for Adaptive and Self-Managing Systems* (Seoul, Republic of Korea) *(SEAMS '20)*. ACM, 45–56. https://doi.org/10.1145/3387939.3388616

[21] Frederick Hayes-Roth. 1983. Rule-based systems. *Commun. ACM* 26, 9 (1983), 921–932.

[22] Brennen Hill. 2025. Generative World Models of Tasks: LLM-Driven Hierarchical Scaffolding for Embodied Agents. arXiv:2509.04731 [cs.AI] https://arxiv.org/abs/2509.04731

[23] P Hitzler, MK Sarker, TR Besold, AD Garcez, S Bader, H Bowman, P Domingos, KU Kühnberger, LC Lamb, et al. 2022. Neural-symbolic learning and reasoning: A survey and interpretation. *Frontiers in artificial intelligence and applications* 342 (2022), 1–51.

[24] Yi-Fei Hu, Joseph Heffner, Apoorva Bhandari, and Oriel FeldmanHall. 2025. Goals bias face perception. *Journal of Experimental Psychology: General* (2025).

[25] Utkarshani Jaimini, Cory Henson, and Amit Sheth. 2024. Causal neuro-symbolic AI for root cause analysis in smart manufacturing. In *International Semantic Web Conference*.

[26] Jeffrey O. Kephart and David M. Chess. 2003. The Vision of Autonomic Computing. *Computer* 36, 1 (2003), 41–50.

[27] Gabriele Kern-Isberner, Christoph Beierle, and Gerhard Brewka. 2020. Syntax Splitting = Relevance + Independence: New Postulates for Nonmonotonic Reasoning From Conditional Belief Bases. In *Proceedings of the 17th International Conference on Principles of Knowledge Representation and Reasoning*. 560–571. https://doi.org/10.24963/kr.2020/56

[28] Kathrin Korte, Christian Medeiros Adriano, Sona Ghahremani, and Holger Giese. 2025. Causal knowledge transfer for multi-agent reinforcement learning in dynamic environments. In *2025 IEEE International Conference on Autonomic Computing and Self-Organizing Systems Companion (ACSOS-C)*. IEEE, 154–159.

[29] Douglas B Lenat, Mayank Prakash, and Mary Shepherd. 1990. CYC: Using common sense knowledge to overcome brittleness and knowledge acquisition bottlenecks. *AI Magazine* 11, 2 (1990), 65–85.

[30] LessWrong Community. 2023. AI Safety 101: Reward Misspecification. https://www.lesswrong.com/posts/mMBoPnFrFqQJKzDsZ/ai-safety-101-reward-misspecification. Accessed 2026-02-12.

[31] Hu Liu, Alexander Gegov, and Mihaela Cocea. 2016. Rule-based systems: a granular computing perspective. *Granular Computing* 1, 4 (2016), 259–274.

[32] Yuchen Liu, Luigi Palmieri, Sebastian Koch, Ilche Georgievski, and Marco Aiello. 2025. Delta: Decomposed efficient long-term robot task planning using large language models. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 10995–11001.

[33] Adithya Murali, Lucas Peña, Christof Löding, and P. Madhusudan. 2023. A First-order Logic with Frames. *ACM Trans. Program. Lang. Syst.* 45, 2, Article 7 (May 2023), 44 pages. https://doi.org/10.1145/3583057

[34] Mark A Musen. 1989. Automating knowledge acquisition for expert systems. *Methods of Information in Medicine* 28, 4 (1989), 237–244.

[35] Emilia Oikarinen. 2006. *Modular Answer Set Programming*. Technical Report Research Report A106. Helsinki University of Technology, Laboratory for Theoretical Computer Science.

[36] OpenAI. 2025. OpenAI developer platform. https://platform.openai.com/ Last visited 16.05.2025.

[37] Gabriele Paul. 1993. Approaches to abductive reasoning: an overview. *Artificial intelligence review* 7, 2 (1993), 109–152.

[38] Luc De Raedt and Kristian Kersting. 2008. *Statistical Relational Learning and Inductive Logic Programming*. Springer, Berlin. Bridging statistical learning and symbolic rule induction.

[39] Joar Skalse, Nikolaus Howe, Dmitrii Krasheninnikov, and David Krueger. 2022. Defining and characterizing reward gaming. *Advances in Neural Information Processing Systems* 35 (2022), 9460–9471.

[40] Jiaxin Wen, Ruiqi Zhong, Pei Ke, Zhihong Shao, Hongning Wang, and Minlie Huang. 2024. Learning task decomposition to assist humans in competitive programming. *arXiv preprint arXiv:2406.04604* (2024).

[41] Lilian Weng. 2024. Reward Hacking in Reinforcement Learning. https://lilianweng.github.io/posts/2024-11-28-reward-hacking/. Accessed 2026-02-12.

[42] Fangkai Yang, Daoming Lyu, Bo Liu, and Steven Gustafson. 2018. Peorl: Integrating symbolic planning and hierarchical reinforcement learning for robust decision-making. *arXiv preprint arXiv:1804.07779* (2018).