TRACE: A TRAFFIC REGULATION AGENT WITH COMPOSITIONAL EVIDENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Traffic regulation compliance requires reasoning over temporal events, participant roles, thresholds, and exceptions across continuous driving processes. However, fixed regulatory specifications offer limited flexibility in handling context-dependent legal conditions, while LLM-based assessments with static context are constrained by the evidence provided upfront. We propose TRACE, a traffic regulation agent with compositional evidence that separates reusable evidence computation from regulation-specific reasoning. A stateful runtime maintains factual and temporal evidence, while Law Skills guide an event-centered workflow to acquire and compose the evidence needed for compliance assessment. To extend its regulatory coverage, TRACE supports constructing initial skills for new regulations and refining their evidence procedures through execution feedback within fixed legal and tool boundaries. TRACE achieves 96.67% strict accuracy on 751 real-world traffic regulation compliance cases. Starting from automatically initialized skills, evolution improves strict accuracy from 46.6% to 81.4% on held-out driving cases. These results demonstrate accurate traffic compliance assessment and the ability to improve regulation-specific evidence procedures through execution feedback.

1 INTRODUCTION

With autonomous driving systems advancing toward real-world deployment, driving decisions must comply with traffic regulations. Even without a collision, speeding, running a red light, or failing to yield as required can violate traffic regulations, making collision occurrence alone insufficient for assessing driving behavior (Sun et al., 2022). Beyond assessing collision risk from geometric and kinematic states, compliance assessment requires interpreting regulatory conditions, participant roles, thresholds, exceptions, and open-ended legal concepts such as "ensuring safety" and "not blocking normal traffic". The ambiguity and context dependence of these concepts, together with potential rule conflicts, make machine-executable representations of regulations challenging (Prakken, 2017; Bhuiyan et al., 2024). Across a continuous driving process, these legal conditions must also be linked to observations of the relevant participants at the appropriate times. Figure 1 contrasts fixed specifications, static-context LLM assessment, and TRACE's adaptive evidence acquisition and composition.

Translating natural-language regulations into executable specifications is a major approach to driving compliance assessment. Formal methods encode regulatory requirements in temporal and deontic logic (Esterle et al., 2020; Bhuiyan et al., 2024), online monitoring methods perform compliance checks through atomic propositions, provision-specific triggers, and state constraints (Yu et al., 2024), and regulation-driven testing methods use executable specifications to search for violations or generate test scenarios from regulations (Sun et al., 2022; Deng et al., 2025). Executable specifications support deterministic checks for encoded regulations; however, when new or revised provisions change applicability conditions, participant relations, or exceptions, the corresponding specifications must be updated and validated. A regulation typically combines applicability conditions, participant relations, quantitative criteria, and temporal constraints, requiring multiple observable facts to be composed according to its requirements (Esterle et al., 2020; Yu et al., 2024). When regulatory requirements change and existing factual capabilities still cover them, the underlying evidence computation can be reused while evidence composition and decision logic are adapted. The

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

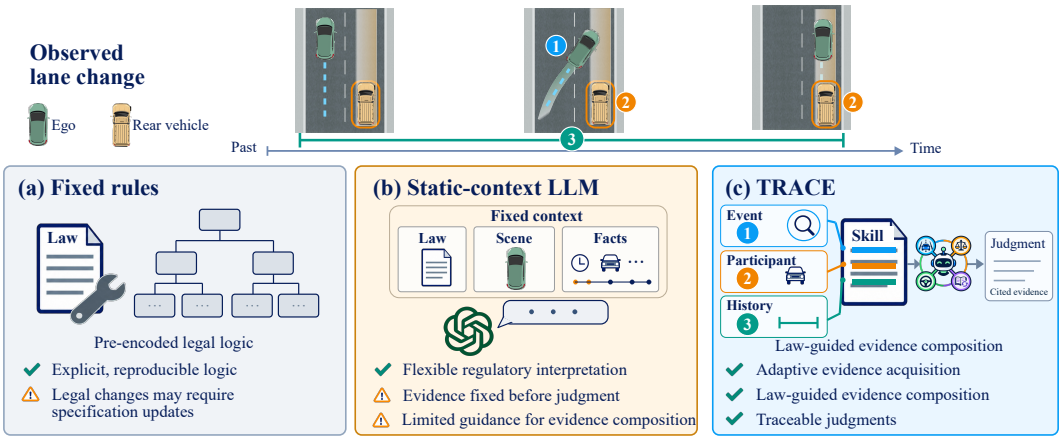


Figure 1: **Beyond fixed rules and static-context reasoning.** TRACE uses Law Skills to turn regulatory knowledge into procedural guidance for adaptive evidence acquisition and composition, grounding flexible compliance reasoning in relevant events, participants, and temporal evidence.

central challenge we address is therefore how to reuse low-level fact computation while adapting evidence composition and compliance judgments to regulatory semantics.

Large language models (LLMs) offer a new route to interpreting and adapting natural-language regulations. Existing studies use LLMs to translate traffic regulations into metric temporal logic (Manas et al., 2024), or to retrieve relevant provisions, evaluate candidate actions, and derive driving requirements for specific scenarios (Cai et al., 2026; Jian et al., 2026). These studies offer ways to reduce manual effort in regulation interpretation and scenario adaptation, but regulatory reasoning still requires explicit grounding in the driving scenario. Without structured scenario grounding, LLMs can retrieve irrelevant provisions or miss applicable ones, yielding imprecise driving requirements (Jian et al., 2026). Compliance assessment over continuous driving processes further requires grounding legal conditions in factual evidence about specific events, participants, and time intervals. Tool-augmented agents combine language-model reasoning with external computation and information acquisition through specialized modules and external interaction (Karpas et al., 2022; Yao et al., 2022). This mechanism provides a basis for separating regulatory reasoning from fact computation: LLMs interpret regulations and identify required evidence, while external tools compute observable facts. Execution feedback can further support revisions to reusable workflows and skill procedures (Zhang et al., 2025; 2026), providing a basis for improving evidence acquisition and composition within fixed legal requirements and tool semantics.

Building on this idea, we propose TRACE (Traffic Regulation Agent with Compositional Evidence), an auditable agent that explicitly separates reusable fact computation from regulation-dependent decision rules. TRACE identifies applicable regulations, queries tools for evidence, and produces compliance decisions with traceable evidence chains. When existing evidence capabilities cover a new regulation, adaptation only updates skill-level evidence acquisition and composition procedures while leaving the underlying fact computation unchanged. To support this adaptation, TRACE can create or revise skills for new regulations or execution failures, with all updates externally verified before deployment.

Our main contributions are summarized as follows:

- We propose **TRACE, a traffic regulation compliance agent with compositional evidence**. TRACE separates stateful factual evidence computation from regulation-specific reasoning. Law Skills guide event anchoring, participant binding, and on-demand temporal evidence acquisition to compose traceable provision-level compliance judgments.
- We propose **a Law Skill evolution mechanism constrained by fixed legal and tool boundaries**. It uses execution feedback to refine evidence-acquisition and composition procedures and independent validation to select effective revisions, supporting the extension of regulatory capabilities through reusable evidence tools.

- We **systematically evaluate TRACE on real-world driving compliance cases** and use component ablations to examine the role of evidence acquisition and composition. We further evaluate constrained evolution from automatically initialized skills, demonstrating improvements in compliance assessment on held-out cases through execution feedback.

2 Related Work

2.1 Traffic Regulation Monitoring and Testing

Traffic regulation monitoring commonly translates natural-language provisions into symbolic specifications before evaluating driving behavior. Semantic state representations and temporal logic encode behavioral and temporal requirements (Esterle et al., 2019; 2020), while defeasible deontic logic represents exceptions and rule conflicts (Bhuiyan et al., 2024). Online legal monitoring operationalizes this approach through atomic propositions, provision-specific triggers, and logical judgments (Yu et al., 2024). These representations support explicit, deterministic checks, but fix the interpretation of a provision in its executable specification. A change in applicability, participant roles, or exceptions therefore requires revising and validating that specification, even when the underlying observations remain usable. This motivates separating reusable factual evidence from the regulation-specific logic that composes it. TRACE places this composition in Law Skills, allowing adaptation over an unchanged evidence runtime when its factual capabilities cover the new requirements.

Regulation-driven testing relies on a compliance oracle to determine whether an execution violates the requirements under test. LawBreaker couples traffic-law specifications with coverage-guided fuzzing to search for violations (Sun et al., 2022). TARGET automates rule-to-scenario translation through validated LLM-generated representations in a compositional domain-specific language (Deng et al., 2025). FixDrive uses specification-based trace analysis to identify problematic behavior and guide driving-policy repairs (Sun et al., 2025). These approaches expand the executions explored or improve the driving system under test, while violation detection remains tied to the encoded requirements. Generating additional scenarios does not by itself resolve how a changed provision should be interpreted against a recorded driving process. TRACE addresses this assessment problem: acquiring and composing temporal evidence for provision-level judgments. Its role is complementary to test generation and policy repair, with adaptation focused on the assessment procedure.

2.2 LLM-Based Compliance Agents and Skills

LLMs reduce the effort of interpreting regulations, but their role in the assessment pipeline matters. TR2MTL automates translation into metric temporal logic, retaining a symbolic specification as the output of regulation understanding (Manas et al., 2024). DriveReg instead retrieves regional regulations and reasons about candidate actions, providing explanations with references to the applicable rules (Cai et al., 2026). Related approaches derive scenario-aware driving requirements or connect visual observations with regulation databases for path planning (Jian et al., 2026; Han et al., 2025). DriveCombo further documents difficulties in multi-rule integration and conflict resolution (Ma et al., 2026). For assessment over continuous driving observations, regulatory understanding must additionally be grounded in the relevant event, participant, and time interval. A retrieved provision or plausible action explanation alone does not establish these facts. TRACE makes their acquisition explicit through stateful evidence queries and Law Skills that associate legal conditions with the corresponding observations, preserving uncertainty when decisive evidence is missing.

Tool-augmented agents provide a basis for separating semantic reasoning from external computation: MRKL uses specialized modules, and ReAct interleaves reasoning with tool interaction (Karpas et al., 2022; Yao et al., 2022). Reusable skills and workflows retain execution experience across tasks, as demonstrated by Voyager, AutoGuide, and Agent Workflow Memory (Wang et al., 2023; Fu et al., 2024; Wang et al., 2025); AFlow and SkillEvolver further optimize procedures through execution feedback (Zhang et al., 2025; 2026). These mechanisms support procedural adaptation, but compliance assessment also requires an explicit boundary on what may change. Feedback can improve event selection, participant binding, and evidence acquisition, while legal requirements and the meaning of computed facts must remain fixed. To address this gap, we propose TRACE,

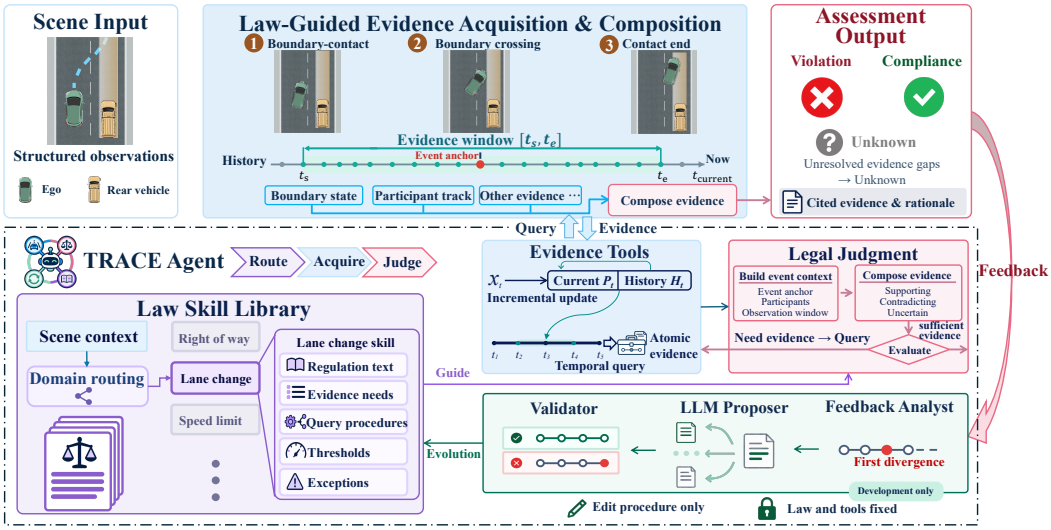


Figure 2: **The TRACE framework.** TRACE decouples stateful evidence computation from regulation-specific reasoning. Guided by Law Skills, the agent iteratively queries and composes event-, participant-, and time-specific evidence for compliance judgments. During development, execution feedback drives procedural revisions selected through independent validation, while legal requirements and tool semantics remain fixed.

which decouples stateful fact computation from regulatory reasoning and uses Law Skills to guide evidence acquisition and composition, refining assessment procedures through execution feedback while keeping legal requirements and tool semantics fixed.

3 Methodology

3.1 Problem Formulation and TRACE Overview

TRACE assesses traffic compliance at time t using current and historical structured driving observations $\mathcal{X}_{0:t} = (x_0, \ldots, x_t)$ and available regulations $\mathcal{R}$. Observations include ego motion, road context, traffic controls, surrounding participants, and geometric measurements. For each regulation r, the output is

$$y_r = \langle a_r, c_r, E_r, z_r \rangle, \tag{1}$$

where a_r is applicability, c_r is compliance status, E_r contains evidence references, and z_r explains the finding. A driving process can involve several regulations, yielding $Y_t = \{y_r\}$. The task is to ground these findings in the observed process while preserving uncertainty when applicability or decisive evidence cannot be established.

As shown in Figure 2, TRACE produces Y_t through a Route–Acquire–Judge workflow: it selects relevant regulation domains, acquires the evidence required by their Law Skills, and returns provision-level findings.

3.2 Stateful Evidence Runtime

Atomic evidence. The runtime converts driving observations into regulation-neutral evidence packets. An atomic evidence item is represented as

$$e_k = \langle q_k, v_k, \tau_k, \rho_k, \pi_k, \nu_k \rangle, \tag{2}$$

where q_k identifies the factual property, v_k its value, τ_k a time point or interval, ρ_k the participant, and π_k the observation or computation provenance. The validity state ν_k distinguishes usable observations from missing or unreliable evidence. An unavailable participant observation, for example, does not establish that the participant is absent. Evidence values therefore retain their validity and source information when passed to TRACE.

Table 1: Evidence families provided by the stateful proposition runtime.

Evidence family	Role	Representative evidence
Context semantics	Road and traffic-control context	Road and lane types, signs, regulated areas, signal channels
Maneuver and control	Ego-vehicle behavior	Lateral motion, boundary contact, steering intent, turn signals, braking
Object roles	Relevant participant identities	Same-lane and adjacent-lane candidates, object class, travel relation
Geometry and topology	Actor–road geometry	Boundary intersection, signed clearance, stop-line and crosswalk relations, path geometry
Quantitative relations	Measurable relations	Speed, acceleration, gap, closing rate, alignment time, predicted arrival time
Temporal evidence	Changes over time	Change frames, durations, stored snapshots, bounded observation intervals

Temporal state. The runtime processes the trajectory incrementally:

$$(P_t, H_t) = F(x_t, P_{t-1}, H_{t-1}), \tag{3}$$

where P_t contains current factual evidence and H_t retains history needed for temporal queries. Archived packets, state changes, durations, and object identities allow TRACE to inspect an earlier event without supplying the entire observation sequence to the language model. Current observations and historical observations remain associated with their own timestamps; a later state cannot silently substitute for evidence at the event under review.

Queries and responsibility boundary. The interface supports current- and specified-frame queries, bounded intervals, and searches for factual changes. Table 1 summarizes the evidence families. Queries return facts, quality signals, temporal state, and references; TRACE applies legal conditions, thresholds, and exceptions. This separation lets revised skills reuse factual computations.

3.3 Law Skill-guided Reasoning in TRACE

Law Skill interface. A domain skill specifies the regulation-dependent procedure used by TRACE:

$$S_d = \langle R_d, A_d, E_d, Q_d, \Theta_d, X_d, O_d \rangle, \tag{4}$$

where R_d is the regulation set, A_d the applicability conditions, E_d the evidence requirements, Q_d the permitted queries, Θ_d the legal thresholds and system calibrations, X_d the exceptions and missing-evidence handling, and O_d the output contract. Appendix B details the skill resource structure and editing boundaries.

Route. The Router selects candidate regulation domains from a compact scene summary. Each selected domain is processed in an isolated context containing its skill, provision index, permitted evidence queries, and scene summary. Routing identifies the scope of review; the domain agent must still establish applicability from evidence before producing a legal finding.

Acquire. Within a domain, the domain agent organizes evidence around a revisable event hypothesis:

$$h = \langle \alpha, I, P, E^+, E^-, E^? \rangle, \tag{5}$$

where α is the event anchor, I the observation interval, P the bound participants, and E^+, E^-, and $E^?$ the supporting, contradicting, and uncertain evidence. Skill instructions guide the search for candidate anchors in state changes and temporal records. The domain agent binds participants to relevant roles and retrieves the interval needed to assess them. Lane-change review, for example, may require source-lane evidence before crossing and target-lane rear-vehicle evidence during the maneuver. If an event began before recording, frame zero marks the observation boundary rather than proving the event's onset.

Evidence acquisition is iterative. A query can reveal an incorrect participant assignment, shift the relevant interval, or expose a missing condition, requiring revision of h. TRACE begins with an initial scene evidence package and uses it to narrow the relevant provisions, participants, and unresolved conditions. It then requests additional geometry, distance, temporal, or prediction evidence

as needed to resolve those conditions. Skill-permitted historical or alternative sources can resolve gaps, but missing evidence is not converted into a favorable or adverse observation. Acquisition stops when the decisive conditions are supported or a critical gap remains unresolved after the permitted evidence paths have been considered.

Judge. The domain agent follows the Law Skill to associate event- and participant-specific evidence with the regulation's applicability conditions, behavioral requirements, and exceptions. It evaluates these conditions using the prescribed thresholds and combines them according to their relationships in the regulation. The evidence required for a finding follows this legal structure. For obligations involving multiple participant roles, evidence establishing a violation of an applicable obligation may support a violation finding, while a compliant finding requires sufficient evidence that the relevant obligations are satisfied across applicable roles, accounting for the regulation's exceptions. Missing observations or unresolved participant identities leave the corresponding conditions unresolved. Applicability and compliance are reported separately. Each finding records its legal basis, supporting and conflicting evidence, unresolved gaps, and an explanation. Query traces retain call order, arguments, and warnings, while frame and participant references connect the finding to the observed process. Domain findings are aggregated by provision.

3.4 Constrained Law Skill Evolution

Initialization and editable scope. For a new regulation r, TRACE automatically constructs an initial skill S_r^0 from the regulation text and available evidence-tool interfaces. Building on the reasoning procedure in Section 3.3, evolution improves how the skill acquires and composes evidence through labeled development trajectories. The editable procedure at iteration k is

$$\Pi_r^k = \langle s_{\text{anchor}}, s_{\text{bind}}, s_{\text{interval}}, s_{\text{evidence}}, s_{\text{stop}} \rangle, \tag{6}$$

covering event anchoring, participant binding, observation-interval selection, evidence acquisition and composition, and stopping conditions. Regulation text, statutory thresholds, fixed calibrations, proposition semantics, query implementations, and missing-data definitions remain fixed.

Regulation-guided execution feedback. The Proposer analyzes predictions, query traces, and evidence against the skill's applicability conditions, obligations, exceptions, and evidence requirements. It checks event relevance, participant roles, temporal coverage, and evidential support to identify procedural failures. Incorrect role assignments motivate binding revisions; incomplete temporal coverage motivates interval changes or follow-up queries. Successful executions supply steps to retain. Feedback f_i combines outcome correctness with execution evidence, and proposed repairs are tested through re-execution.

Revision under fixed legal requirements. The Proposer converts feedback into reusable edits specifying queries, argument sources, evidence-dependent follow-ups, and stopping conditions. These edits preserve the legal conditions and evidence sufficiency requirements in Section 3.3, as well as the meanings of missing, unresolved, and explicitly absent evidence.

Given a development batch B_k, the candidate set includes the parent skill and M proposed revisions:

$$\mathcal{C}_k = \{S_r^k\} \cup \{\text{Patch}_j(S_r^k, B_k, \{f_i\}_{i \in B_k})\}_{j=1}^{M}. \tag{7}$$

Candidate checks enforce the permitted resource scope, frozen content, and published query interfaces. Revised procedures must be expressed in reusable evidence conditions rather than case identifiers or stored answers.

Evaluation and selection. Development execution evaluates candidate revisions, and replay of earlier cases assesses whether useful behavior is retained. The parent remains eligible during selection. Independent validation selects the retained skill version, which is frozen before held-out evaluation or deployment. Test outcomes are excluded from revision and selection.

4 Experiments

4.1 Setup

Dataset construction. The benchmark contains 751 provision-level cases constructed from randomly sampled driving logs collected on real roads. A rule-based system provides preliminary

screening, followed by independent review of every case by three graduate researchers in autonomous driving. Reviewers use Chinese traffic regulations as the legal basis and adopt the corresponding thresholds from Yu et al. (2024) for conditions requiring quantitative operationalization. Disagreements are resolved through discussion, and cases with indeterminate labels are excluded. Each case is evaluated at a manually specified time, with TRACE accessing driving facts up to that time through atomic propositions and evidence-query interfaces. Appendix A details case construction and annotation.

Benchmark and protocols. The benchmark draws on 749 driving logs. The evaluation covers 13 regulations across six review domains, involving threshold-based judgments, temporal events, participant-role dependencies, and exceptions. Main evaluation and component ablations use the same benchmark. Main evaluation uses frozen hand-authored skills. Evolution uses 104 development and 104 validation cases, with 543 benchmark cases forming the held-out test set, to evaluate regulation adaptation from automatically initialized skills. Full protocols, data overlap, and selection information appear in Appendix A, Table 7.

Baselines. Table 2 compares TRACE with a rule engine, DriveReg (adapted), and three controlled LLM baselines. The rule engine deterministically checks recorded driving states against explicitly encoded traffic rules (Yu et al., 2024). DriveReg uses retrieval-augmented regulation reasoning (Cai et al., 2026). Our controlled baselines vary access to regulation text and factual evidence. LLM-Evidence makes a single-prompt judgment using factual evidence collected by TRACE, testing whether factual access alone explains TRACE's advantage. The DriveReg adaptation and evidence-matched comparison are detailed in Appendices A.5 and A.6, respectively. For Law Skill evolution, we compare cumulative experiential memory (Zhao et al., 2024), structured reflection (Shinn et al., 2023), textual-gradient editing (Yüksekgönül et al., 2025), and population-based prompt evolution (Guo et al., 2024) under the same development budget and held-out test set.

Implementation details. We use GPT-5.6-Luna and DeepSeek-V4-Flash as TRACE executors. For evolution, GPT-5.6-Luna performs development-stage execution and updates. Our method selects checkpoints by validation Strict Accuracy for each regulation and budget, then freezes them for evaluation with DeepSeek. Test labels are used only for evaluation. Detailed update rules, selection procedures, and information-access boundaries appear in Appendix A.

Evaluation metrics. Our primary metric is Strict Accuracy against binary compliance labels; unknown, not-triggered, and missing outputs count as errors. Violation recall and false positive rate (FPR) measure the fractions predicted as violations among all true violation and compliant cases, respectively. We also report Macro-F1, unknown rate, and queries per case. For evolution, normalized area under the learning curve (AULC) measures learning efficiency by summarizing accuracy across development budgets. Detailed scoring conventions and statistical procedures appear in Appendix C.

4.2 Main Results

End-to-end compliance assessment. In Table 2, TRACE achieves 96.67% Strict Accuracy on the full benchmark, compared with 86.55% for the rule engine and 85.22% for LLM-Evidence. The latter receives the driving facts collected by TRACE, supporting the benefit of TRACE's skill-guided execution beyond factual access alone. Disabling DeepSeek thinking changes Strict Accuracy to 88.28% and Macro-F1 to 91.77%. GPT-5.6-Luna also achieves 94.14% Strict Accuracy. Detailed Macro-F1 and class-wise recall results are provided in Appendix A.7, Table 13.

Regulation adaptation through Law Skill evolution. Our method achieves the highest accuracy at every nonzero development budget. With the full development budget, the validation-selected skills reach 81.4% Strict Accuracy, improving over the initial seed by 34.8 percentage points and exceeding the strongest baseline by 13.3 points. Its 73.9% normalized AULC indicates more effective use of limited development examples. These results support learning regulation-specific evidence procedures from automatically initialized skills through execution feedback.

4.3 Qualitative Analysis

Figure 3 illustrates how TRACE translates human feedback into a revised evidence procedure. The initial skill identifies prolonged boundary contact and returns a violation finding, but leaves the

Table 2: Traffic compliance assessment on the full benchmark (%). DS-T/NT: DeepSeek-V4-Flash with/without thinking; Luna: GPT-5.6-Luna; Det.: deterministic. Input columns indicate information categories, not identical evidence across methods. Best results are bold.

Method	Model	Input				Strict Acc. ↑	Viol. Recall ↑	FPR ↓
		Scene summary	Regulatory input	Extra facts	Fixed calib.			
Rule Engine (Yu et al., 2024)	Det.	*Encoded rules + driving state*				86.55	83.27	1.23
DriveReg (adapted) (Cai et al., 2026)	DS-T	✓	Retrieved text	–	–	56.59	47.53	38.11
Controlled LLM baselines								
LLM-Base	DS-T	✓	–	–	–	57.12	66.54	47.95
LLM-Law	DS-T	✓	Target-law text	–	–	59.12	53.99	38.11
LLM-Evidence	DS-T	✓	Target-law text	✓	✓	85.22	71.48	7.38
TRACE	Luna	✓	Full	✓	✓	94.14	**92.78**	1.23
	DS-NT	✓	Full	✓	✓	88.28	89.35	2.87
	DS-T	✓	Full	✓	✓	**96.67**	**92.78**	**0.20**

Table 3: Law Skill evolution evaluated on the held-out test set (%). All methods start from the same automatically initialized seed. Fractions indicate cumulative development-set budgets; All denotes the full budget. Panel (b) compares the seed with our validation-selected evolved skills.

(a) Learning efficiency

Skill update method	Strict Accuracy (%)				AULC ↑	Macro-F1 (All) ↑
	1/4	1/2	3/4	All		
Static seed	46.6	46.6	46.6	46.6	46.6	61.2
Memory (Zhao et al., 2024)	65.4	66.3	64.8	68.1	63.5	78.7
Reflexion (Shinn et al., 2023)	65.9	65.9	66.5	66.9	63.8	76.9
TextGrad (Yüksekgönül et al., 2025)	60.6	65.9	63.2	63.0	61.1	75.5
EvoPrompt (Guo et al., 2024)	63.7	63.7	63.7	63.5	61.6	76.5
Ours	**74.4**	**76.1**	**81.0**	**81.4**	**73.9**	**85.4**

(b) Full-budget performance

Setting	Strict Acc. ↑	Macro-F1 ↑	Viol. R. ↑	FPR ↓	Unknown	Not Trig.
TRACE + Seed	46.59	61.24	46.63	**4.21**	20.99	28.36
TRACE + Evolved	**81.40**	**85.37**	**77.30**	4.47	**11.79**	**0.18**

surrounding circumstances unexplored. Human feedback identifies a possible obstacle-avoidance justification and requests investigation of the event while retaining the violation label. The revised skill directs TRACE to inspect historical observations within the contact interval, bind the relevant vehicle, and query its motion and local passage space.

With the same input and tools, the subsequent execution identifies a nearly stationary truck and a local clearance smaller than the configured ego-vehicle width. These observations support an obstacle-avoidance explanation, which TRACE reports alongside the unchanged violation finding. This example illustrates how feedback can improve an assessment by changing what evidence the agent seeks and how it combines that evidence. The improvement lies in grounding the contextual explanation through additional investigation while preserving the established violation criterion.

4.4 Ablation Study

We evaluate component removals on the full benchmark with DeepSeek thinking, using the main TRACE results as reference. Routed variants share frozen routing decisions (Appendix A.3).

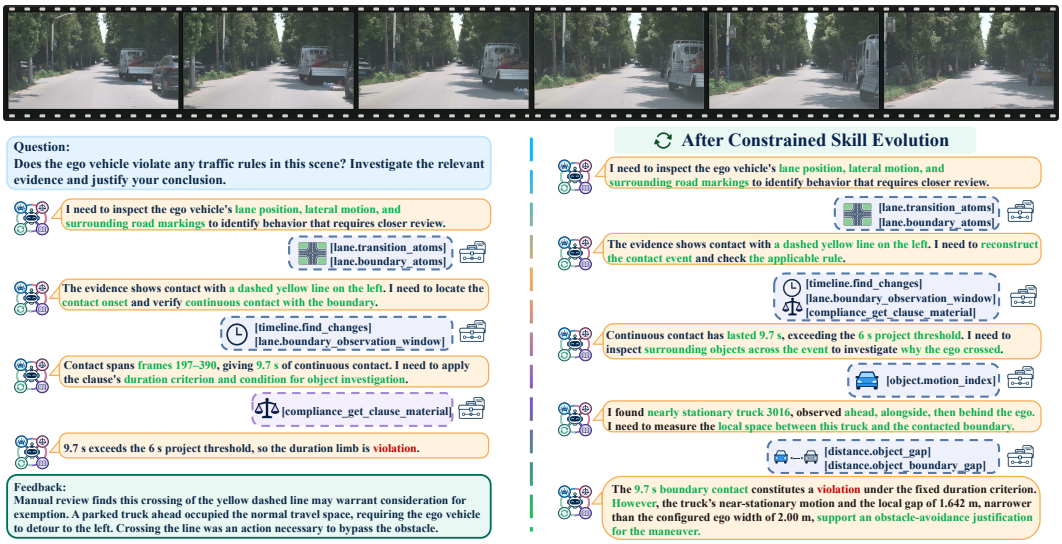


Figure 3: **Evidence acquisition, composition, and skill evolution in TRACE.** Human feedback extends an initial duration-based finding to investigate historical object motion and local passage space. The resulting evidence supports an obstacle-avoidance justification alongside the violation finding. The same 6 s project threshold is retained.

Table 4: Component ablations on the full benchmark with DeepSeek thinking. Metrics are percentages except queries per case.

Setting	Strict Acc. ↑	Viol. Recall ↑	FPR ↓	Unknown	Queries / Case
TRACE	**96.67**	92.78	**0.20**	0.27	25.4
w/o Law Skill	56.46	55.13	4.30	23.44	24.5
w/o Temporal Evidence	65.11	67.30	2.66	25.43	21.0
w/o Adaptive Querying	42.21	35.36	0.41	52.33	0.0
w/o Router	96.01	**93.16**	1.02	**0.00**	32.9

Contribution of evidence procedures. Removing adaptive querying causes the largest accuracy drop (96.67% to 42.21%) and raises unknown outputs to 52.33%, highlighting the need to acquire additional evidence during assessment. Removing temporal evidence or Law Skills also substantially reduces accuracy, indicating that factual access must be complemented by historical context and regulation-specific guidance on evidence acquisition and composition.

Role of routing. Reviewing all six domains yields similar accuracy (96.01% vs. 96.67%) but increases recorded queries per case from 25.4 to 32.9. This suggests that routing primarily reduces unnecessary domain investigation, while evidence acquisition and composition account for the larger performance differences.

5 Conclusion

We presented TRACE, a Law Skill-guided framework that decouples stateful evidence computation from regulatory reasoning. TRACE acquires and composes event-centered evidence and improves evidence procedures through constrained skill evolution. TRACE achieves 96.67% Strict Accuracy on the full benchmark, while skill evolution improves held-out accuracy from 46.6% to 81.4%, supporting evidence composition and procedural adaptation for traffic compliance assessment.

Limitations. TRACE depends on the quality of upstream observations and evidence tools. Errors in their outputs may propagate to compliance judgments. Extending TRACE to new regulations may require evidence capabilities beyond the existing tools. Skill evolution requires labeled development cases and independent validation to select revised procedures.

AI USE STATEMENT

We used generative AI tools to assist with code implementation and debugging, literature search, manuscript drafting, translation, language editing, and figure preparation. Generative image editing was used to remove recording overlays from case-study camera images, as disclosed in Appendix D. These tools were used under author direction; the core research ideas and methodology were developed by the authors. The authors take full responsibility for the implementation, experimental results, references, and final manuscript.

ETHICS STATEMENT

This work studies traffic-regulation assessment using recorded real-world driving data and human-reviewed compliance labels. The experiments evaluate judgments on recorded observations and do not control vehicles. Incorrect judgments can have safety and legal consequences, so use in operational driving systems or enforcement would require further validation and appropriate human oversight. Data use and any future release must respect applicable privacy, confidentiality, and licensing requirements.

REPRODUCIBILITY STATEMENT

Sections 3 and 4.1 describe the method and evaluation setup. Appendix A documents dataset construction, annotation, data splits, executor settings, baseline adaptations, ablations, and fixed thresholds. Appendix B specifies Law Skill resources and editing boundaries, and Appendix C defines scoring conventions and statistical procedures. Appendix D presents recorded agent inputs, tool evidence, and findings. Execution traces, skill versions, and validation records are retained with the experimental artifacts. The benchmark data can be made publicly available after de-identification.

REFERENCES

Hanif Bhuiyan, Guido Governatori, Andy Bond, and Andry Rakotonirainy. Traffic rules compliance checking of automated vehicle maneuvers. *Artif. Intell. Law*, 32(1):1–56, 2024. doi: 10.1007/S10506-022-09340-9.

Tianhui Cai, Yifan Liu, Zewei Zhou, Haoxuan Ma, Seth Z. Zhao, Zhiwen Wu, Xu Han, Zhiyu Huang, and Jiaqi Ma. Driving with regulation: Trustworthy and interpretable decision-making for autonomous driving with retrieval-augmented reasoning. In Sven Koenig, Chad Jenkins, and Matthew E. Taylor (eds.), *Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2026, Singapore, January 20-27, 2026*, pp. 38287–38295. AAAI Press, 2026. doi: 10.1609/AAAI.V40I45.41168.

Yao Deng, Zhi Tu, Jiaohong Yao, Mengshi Zhang, Tianyi Zhang, and James Xi Zheng. TARGET: traffic rule-based test generation for autonomous driving via validated llm-guided knowledge extraction. *IEEE Trans. Software Eng.*, 51(7):1950–1968, 2025. doi: 10.1109/TSE.2025.3569086.

Klemens Esterle, Vincent Aravantinos, and Alois C. Knoll. From specifications to behavior: Maneuver verification in a semantic state space. In *2019 IEEE Intelligent Vehicles Symposium, IV 2019, Paris, France, June 9-12, 2019*, pp. 2140–2147. IEEE, 2019. doi: 10.1109/IVS.2019.8814241.

Klemens Esterle, Luis Gressenbuch, and Alois C. Knoll. Formalizing traffic rules for machine interpretability. In *3rd IEEE Connected and Automated Vehicles Symposium, CAVS 2020, Victoria, BC, Canada, November 18 - December 16, 2020*, pp. 1–7. IEEE, 2020. doi: 10.1109/CAVS51000.2020.9334599.

Yao Fu, Dong-Ki Kim, Jaekyeom Kim, Sungryull Sohn, Lajanugen Logeswaran, Kyunghoon Bae, and Honglak Lee. Autoguide: Automated generation and selection of context-aware guidelines for large language model agents. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information*

Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024, 2024.

Qingyan Guo, Rui Wang, Junliang Guo, Bei Li, Kaitao Song, Xu Tan, Guoqing Liu, Jiang Bian, and Yujiu Yang. Connecting large language models with evolutionary algorithms yields powerful prompt optimizers. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.

Xu Han, Zhiwen Wu, Xin Xia, and Jiaqi Ma. Traffic regulation-aware path planning with regulation databases and vision-language models. In *IEEE International Conference on Robotics and Automation, ICRA 2025, Atlanta, GA, USA, May 19-23, 2025*, pp. 11024–11030. IEEE, 2025. doi: 10.1109/ICRA55743.2025.11128730.

Bowen Jian, Rongjie Yu, Hong Wang, Liqiang Wang, and Zihang Zou. Towards lawful autonomous driving: Deriving scenario-aware driving requirements from traffic laws and regulations. *CoRR*, abs/2604.24562, 2026. doi: 10.48550/ARXIV.2604.24562.

Ehud Karpas, Omri Abend, Yonatan Belinkov, Barak Lenz, Opher Lieber, Nir Ratner, Yoav Shoham, Hofit Bata, Yoav Levine, Kevin Leyton-Brown, Dor Muhlgay, Noam Rozen, Erez Schwartz, Gal Shachaf, Shai Shalev-Shwartz, Amnon Shashua, and Moshe Tennenholtz. MRKL systems: A modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning. *CoRR*, abs/2205.00445, 2022. doi: 10.48550/ARXIV.2205.00445.

Enhui Ma, Jiahuan Zhang, Guantian Zheng, Tao Tang, Shengbo Eben Li, Yuhang Lu, Xia Zhou, Xueyang Zhang, Yifei Zhan, Kun Zhan, Zhihui Hao, Xianpeng Lang, and Kaicheng Yu. Drivecombo: Benchmarking compositional traffic rule reasoning in autonomous driving. *CoRR*, abs/2603.01637, 2026. doi: 10.48550/ARXIV.2603.01637.

Kumar Manas, Stefan Zwicklbauer, and Adrian Paschke. TR2MTL: LLM based framework for metric temporal logic formalization of traffic rules. In *IEEE Intelligent Vehicles Symposium, IV 2024, Jeju Island, Republic of Korea, June 2-5, 2024*, pp. 1206–1213. IEEE, 2024. doi: 10.1109/IV55156.2024.10588650.

Henry Prakken. On the problem of making autonomous vehicles conform to traffic law. *Artif. Intell. Law*, 25(3):341–363, 2017. doi: 10.1007/S10506-017-9210-0.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: language agents with verbal reinforcement learning. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

Yang Sun, Christopher M. Poskitt, Jun Sun, Yuqi Chen, and Zijiang Yang. Lawbreaker: An approach for specifying traffic laws and fuzzing autonomous vehicles. In *37th IEEE/ACM International Conference on Automated Software Engineering, ASE 2022, Rochester, MI, USA, October 10-14, 2022*, pp. 62:1–62:12. ACM, 2022. doi: 10.1145/3551349.3556897.

Yang Sun, Christopher M. Poskitt, Kun Wang, and Jun Sun. FIXDRIVE: automatically repairing autonomous vehicle driving behaviour for $0.08 per violation. In *47th IEEE/ACM International Conference on Software Engineering, ICSE 2025, Ottawa, ON, Canada, April 26 - May 6, 2025*, pp. 1921–1933. IEEE, 2025. doi: 10.1109/ICSE55347.2025.00216.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *CoRR*, abs/2305.16291, 2023. doi: 10.48550/ARXIV.2305.16291.

Zora Zhiruo Wang, Jiayuan Mao, Daniel Fried, and Graham Neubig. Agent workflow memory. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *CoRR*, abs/2210.03629, 2022. doi: 10.48550/ARXIV.2210.03629.

Wenhao Yu, Chengxiang Zhao, Hong Wang, Jiaxin Liu, Xiaohan Ma, Yingkai Yang, Jun Li, Weida Wang, Xiaosong Hu, and Ding Zhao. Online legal driving behavior monitoring for self-driving vehicles. *Nature Communications*, 15(1):408, 2024. doi: 10.1038/s41467-024-44694-5.

Mert Yüksekgönül, Federico Bianchi, Joseph Boen, Sheng Liu, Pan Lu, Zhi Huang, Carlos Guestrin, and James Zou. Optimizing generative AI by backpropagating language model feedback. *Nat.*, 639(8055):609–616, 2025. doi: 10.1038/S41586-025-08661-4.

Genrui Zhang, Erle Zhu, Jinfeng Zhou, Caiyan Jia, and Hongning Wang. Skillevolver: Skill learning as a meta-skill. *CoRR*, abs/2605.10500, 2026. doi: 10.48550/ARXIV.2605.10500.

Jiayi Zhang, Jinyu Xiang, Zhaoyang Yu, Fengwei Teng, Xionghui Chen, Jiaqi Chen, Mingchen Zhuge, Xin Cheng, Sirui Hong, Jinlin Wang, Bingnan Zheng, Bang Liu, Yuyu Luo, and Chenglin Wu. Aflow: Automating agentic workflow generation. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. Expel: LLM agents are experiential learners. In Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan (eds.), *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2024, February 20-27, 2024, Vancouver, Canada*, pp. 19632–19642. AAAI Press, 2024. doi: 10.1609/AAAI.V38I17.29936.

A Evaluation Protocols

A.1 Dataset Construction and Annotation

Data collection and sampling. The source logs were collected from vehicles operating on real roads. Logs were randomly sampled before annotation. Human reviewers could inspect the full original records, including perception outputs produced by perception algorithms. TRACE instead accessed factual evidence through the atomic proposition runtime and its query interfaces.

Case definition and temporal access. Each case corresponds to a regulation-related interval whose target regulation and boundaries were determined manually. For a violation case, the evaluation time is the midpoint of the annotated violation interval. For a compliant case, it is the midpoint of the interval during which the regulation is subject to monitoring. At evaluation time t, TRACE accesses current and historical driving facts up to t through atomic propositions and evidence-query interfaces.

Annotation and adjudication. A rule-based system provided preliminary screening. Three graduate researchers in autonomous driving independently reviewed every case, using Chinese traffic regulations as the legal basis. Conditions requiring quantitative operationalization were assessed using the corresponding thresholds adopted from Yu et al. (2024). The reviewers discussed disagreements to reach a consensus, and cases whose compliance status could not be determined were excluded.

A.2 Data Splits and Evaluation Protocols

Benchmark and data split. Tables 7 and 5 summarize the protocols and case distribution. The evaluation benchmark comprises 751 provision-level cases from 749 driving logs. An additional 49 cases used for implementation development and debugging are excluded from the reported main evaluation and ablations. Main evaluation uses frozen hand-authored skills, with component removals applied in the ablation study. Skill evolution uses 104 development cases, 104 validation cases, and 543 held-out test cases within the benchmark. Within each regulation, source logs are disjoint across splits.

Table 5: Case distribution by regulation and review domain. C/V denotes compliant/violation cases. Development and validation columns give the final case counts used for evolution; the test labels are binary. Corresponding regulation texts appear in Table 6.

Regulation	Domain	Full C/V	Dev.	Val.	Test C/V
IMR 38.1	Traffic signal	52/11	8	8	44/3
IMR 38.4	Right of way	44/19	8	8	36/11
IMR 38.6	Right of way	42/18	8	8	34/10
IMR 44.1	Lane change	15/12	8	8	7/4
IMR 45.1	Speed	38/41	8	8	30/33
IMR 46.9	Speed	54/14	8	8	46/6
IMR 51.1	Lane usage	35/24	8	8	27/16
IMR 52.2	Right of way	58/10	8	8	48/4
IMR 79.4	Speed	18/4	8	8	4/2
OSP 8.2.1	Distance	38/36	8	8	30/28
TSL 47.1	Speed	35/36	8	8	27/28
TSM 3.6.3	Lane usage	59/8	8	8	47/4
TSM 3.6.4	Lane usage	0/30	8	8	0/14
Total		488/263	104	104	380/163

Learning-curve budgets use the same cumulative fraction of development cases for every regulation. The not-triggered targets belong to lane-change applicability review. Full-benchmark and held-out test results use their binary reference labels. The solid-yellow-line regulation has only violation cases, so its results do not measure compliant-case specificity.

Baseline inputs. All static LLM baselines receive the target regulation identifier and title. LLM-Base, LLM-Law, and DriveReg share a frozen scene summary containing current states and limited

Table 6: English translations of the evaluated clauses in the frozen regulation corpus. Identifiers follow the corpus indexing.

Regulation ID	Regulation text
IMR 38.1	Vehicles are permitted to proceed when the green light is on.
IMR 38.4	Left-turning vehicles must not obstruct straight-moving vehicles that have been permitted to proceed.
IMR 38.6	Right-turning vehicles must not obstruct straight-moving vehicles that have been permitted to proceed.
IMR 44.1	On roads with two or more motor-vehicle lanes in the same direction, a vehicle changing lanes must not affect the normal travel of motor vehicles in the relevant lanes.
IMR 45.1	Motor vehicles must not exceed the speed indicated by speed-limit signs.
IMR 46.9	Vehicles must not exceed 30 km/h while turning.
IMR 51.1	At intersections with designated directional lanes, vehicles must enter the lane corresponding to their intended direction of travel.
IMR 52.2	Where there are no traffic signs or markings controlling the intersection, vehicles must stop and look before entering, and yield to vehicles approaching from the road on the right.
IMR 79.4	Vehicles leaving an expressway must enter the deceleration lane and reduce their speed before exiting.
OSP 8.2.1	While driving, vehicles should maintain a lateral safety clearance greater than 1.5 m from vehicles on either side.
TSL 47.1	Motor vehicles must slow down when approaching a pedestrian crossing.
TSM 3.6.3	A yellow dashed line permits vehicles to cross it briefly when it is safe on the left, and vehicles must return when conditions permit.
TSM 3.6.4	A solid yellow centerline indicates that vehicles are prohibited from crossing it.

IMR: implementing regulations of the Road Traffic Safety Law; TSL: Road Traffic Safety Law; TSM: traffic-marking rules; OSP: supplementary clauses in the corpus.

Table 7: Evaluation protocols. Counts denote provision-level cases; "–" indicates an unused stage. Shared model settings appear in Table 9.

	Main evaluation and ablations	**Skill evolution**
Skill source	Frozen hand-authored skills, modified by ablation; none for static baselines	Automatically initialized skills; learned procedures or memory
Dev./val./test	– / – / 751	104 / 104 / 543
Execution	Router and domain agents, modified by ablation; static and rule baselines	One target skill and regulation per execution
Selection	Fixed hand-authored skills; no checkpoint selection	Validation for edits; current state for cumulative memory

historical event cues. LLM-Law additionally receives the complete target regulation text; DriveReg receives retrieved snippets. Fixed calib. in Table 2 denotes fixed calibration definitions.

Evolution updates and selection. Table 8 specifies the baseline adaptations. All learning methods receive the same regulation-specific development cases and recorded experience packs, including execution evidence and feedback. The comparison matches exposed-case budgets, not total update calls or computation. Our method permits up to three edit steps per two-case increment, with $M = 3$ revisions and the parent retained at each step. For each regulation and budget, the checkpoint with the highest validation Strict Accuracy is selected, breaking ties in favor of the earliest checkpoint. Frozen skills or memory are then executed by DeepSeek on the test cases.

Table 8: Evolution method implementations. Editing is restricted to the procedure and query-plan resources in Appendix B; legal text and runtime semantics remain fixed.

Method	Update from development experience	Deployed state and selection
Static	No update	Automatically initialized skill
Memory	Append full experience; discard oldest entries beyond 12,000 serialized characters	Seed plus current history; no candidate selection
Reflexion	One reflection call per increment; retain at most eight reflections	Seed plus current reflection memory
TextGrad	One critique call and one editing call produce a local revision	Validation comparison with parent; reject lower accuracy, prefer smaller edits among tied revisions
EvoPrompt	Three mutation/crossover proposals per increment	Retain a population of three by validation accuracy; deploy its best member
TRACE evolution	Revise event anchoring, binding, interval selection, evidence composition, or stopping from execution feedback	Parent and three revisions per edit step; validation-selected checkpoint within budget

Information access. Test executors cannot access test labels, which are read only by the offline deterministic evaluator. Evolution updates may use development labels and feedback, while candidate selection uses validation metrics. Test outcomes never enter updating or selection.

Table 9: Execution settings. DeepSeek denotes DeepSeek-V4-Flash and Luna denotes GPT-5.6-Luna.

Setting	Configuration
Main evaluation	Luna with medium reasoning; DeepSeek with thinking enabled or disabled; deterministic Rule Engine
Static LLM baselines	DeepSeek, thinking enabled, temperature 0, JSON output
Component ablations	DeepSeek, thinking enabled, temperature 0, JSON output
Evolution	Luna with high reasoning for development, updates, and validation; DeepSeek with thinking enabled and temperature 0 for testing
DeepSeek agent limits	32 turns per stage; at most two complete stage attempts; 1,200 s request timeout
Token and query limits	No explicit output-token cap in the DeepSeek requests; agent execution is bounded by turns, with no separate evidence-query-count cap
Reported runs	One retained run per configuration/checkpoint; no averaging across independent repetitions

Final invalid outputs and missing reviews remain in the scoring denominator. Stage retries address execution or output-contract failures and do not use reference labels. The non-thinking configuration changes DeepSeek's thinking setting while retaining the same agent interface and limits.

A.3 Component Ablations

Table 10 defines each intervention. All variants use the full benchmark, its binary labels, and the frozen DeepSeek settings. The full-system reference reuses the main-experiment predictions.

Table 10: What each component ablation changes and retains.

Variant	Removed or changed	Retained
w/o Law Skill	Procedural instructions, evidence requirements, recommended query order, and stopping guidance	Raw regulations, fixed calibrations, neutral query interface, and adaptive querying
w/o Temporal Evidence	Historical/window queries and cross-frame state in domain review; historical router cues are stripped	Current-frame facts and frozen domain selection; the original router may have used history
w/o Adaptive Querying	Additional evidence queries and repeated opening to refresh evidence	Skills, initial domain dossier, and a predetermined atomic evidence package prepared without labels or model outputs
w/o Router	Domain selection and router cues; execute all six domains	Domain skills, adaptive queries, and domain-specific contexts

The first three variants share routing decisions frozen from a separate full-system run. Final ablation outputs receive the main scorer's state-contract and citation normalization, preserving model-produced statuses, reasoning, and evidence references. This processing does not replay corrected online stopping or retries, so the displayed differences also include variation between runs. Query counts cover retained final stages, excluding failed retries and acquisition used to freeze routing; they measure recorded query use rather than total deployment cost.

A.4 Fixed Thresholds and Operational Parameters

Quantitative operationalization follows the monitoring approach of Yu et al. (2024). To support fair comparison, TRACE uses the same numerical thresholds as the Rule Engine for the shared quantitative criteria listed in Table 11. These include limits specified in the frozen corpus and operational calibrations of qualitative requirements. These parameters, proposition semantics, and missing-data handling remain fixed during evolution; learning changes how evidence is collected and combined.

Table 11: Numerical thresholds shared by TRACE and the Rule Engine.

Regulation / use	Value or criterion	Alignment
IMR 45.1 / speed	Applicable posted speed limit	Shared with Rule Engine
IMR 46.9 / turning	Maximum speed 30 km/h while turning	Shared with Rule Engine
OSP 8.2.1 / distance	Lateral body-edge clearance > 1.5 m	Shared with Rule Engine
IMR 44.1 / source front	Time to longitudinal body-edge alignment ≥ 1.8 s at event entry	Shared with Rule Engine
IMR 44.1 / target rear	Required gap $\mathrm{clip}(13.6 - 3.4(v_e - v_r), 0, 50)$ m, with speeds in m/s; observed gap must meet it	Shared with Rule Engine
TSL 47.1 / slowdown	Forward crosswalk within 50 m; 2 s response allowance; 30 km/h gate; thereafter acceleration compared with $(v_{30}^2 - v^2)/(2 \cdot 50\,\mathrm{m})$	Shared with Rule Engine
IMR 79.4 / exit lane	2 s response allowance; applicable exit limit, with 60 km/h fallback; current deceleration while above the gate	Shared with Rule Engine
Right-of-way conflict	Signed arrival-time window $0 < \tau < 3.4$ s, combined with participant and path filters	Shared with Rule Engine

A.5 DriveReg Adaptation and Retrieval Protocol

DriveReg adapts retrieval-augmented regulation reasoning to compliance assessment: the frozen scene summary replaces image-based scene analysis, and judgment replaces candidate-action selection. A scene-only query retrieves from 91 raw regulation paragraphs using Qwen/Qwen3-Embedding-0.6B for paragraph and sentence retrieval, Top-5 at each stage, and a paragraph co-

sine threshold of 0.28. The target regulation identifier and title are supplied only at the reasoning stage. The encoder and task-specific prompts are adaptations; no Law Skill procedures or adaptive evidence tools are provided.

The 751 benchmark cases were executed with settings frozen after the pilot. Three invalid outputs remain errors, yielding 425/751 correct decisions. Retrieved snippets include the target regulation family in 106/751 cases (14.11%). This retrieval diagnostic uses snippet identities, since canonical legal references are attached after prediction; it does not measure recall of all applicable regulations.

A.6 Evidence-matched Comparison with TRACE's Output Space

Evidence construction. For each case, initial observations and factual tool outputs from the retained router and domain traces of the frozen DeepSeek TRACE run are combined into one prompt without summarization or truncation. Exact duplicates are removed; case identifiers, reference labels, model-generated reasoning and decisions, and procedural skill content are excluded. Target regulation text and fixed calibration definitions use the same frozen resource version. Factual prompt identity is verified across the binary and four-state runs. Baseline inference cost excludes the upstream TRACE execution that collected the evidence.

Output states. The four-state output space distinguishes *compliant* (the regulation applies and its requirements are satisfied), *violation* (the regulation applies and a requirement is violated), *not-triggered* (the applicability conditions are not met), and *unknown* (available evidence is insufficient to resolve applicability or compliance). Under the shared strict scoring rule, unknown and not-triggered predictions count as errors against binary compliant/violation reference labels.

Matched output space. Table 12 compares both methods under this four-state output space. LLM-Evidence achieves 52.20% Strict Accuracy, compared with 96.67% for TRACE. The gap therefore widens from 11.45 percentage points with the binary baseline reported in the main table to 44.47 percentage points with the four-state baseline. Both baseline runs use identical frozen factual inputs and the same binary reference labels. The four-state baseline returns unknown or not-triggered for 321 cases. TRACE thus retains its advantage when the baseline is required to make a binary judgment, and exhibits a larger advantage when both systems have the same output options.

Table 12: Evidence-matched comparison on the full benchmark. All configurations use DeepSeek-V4-Flash with thinking enabled. Binary and four-state baseline runs share the same frozen factual inputs. All metrics are percentages.

Method	Output space	Strict Acc. ↑	Viol. Recall ↑	FPR ↓
LLM-Evidence	Binary	85.22	71.48	7.38
	Four-state	52.20	53.61	5.12
TRACE	Four-state	**96.67**	**92.78**	**0.20**

A.7 Detailed Results on the Full Benchmark

Table 13: Detailed results on the full benchmark under the shared binary scoring convention. Model abbreviations follow Table 2. All metrics are percentages.

Method	Model	Strict Acc.	Macro-F1	Comp. Recall	Viol. Recall
Rule Engine	Det.	86.55	91.12	88.32	83.27
DriveReg (adapted)	DS-T	56.59	54.21	61.48	47.53
LLM-Base	DS-T	57.12	56.64	52.05	66.54
LLM-Law	DS-T	59.12	57.18	61.89	53.99
LLM-Evidence	DS-T	85.22	83.14	92.62	71.48
TRACE	Luna	94.14	95.89	94.88	**92.78**
TRACE	DS-NT	88.28	91.77	87.70	89.35
TRACE	DS-T	**96.67**	**97.32**	**98.77**	**92.78**

B LAW SKILL REPRESENTATION AND EDITABLE RESOURCES

A Law Skill is a versioned resource bundle interpreted by the domain agent. It combines natural-language evidence-acquisition and composition instructions in `SKILL.md` with JSON resources for legal text, evidence profiles, query plans, and output constraints. The tuple S_d in Section 3.3 describes these semantic roles rather than a literal serialization schema. The agent reads the instructions, selects factual queries, and composes findings.

B.1 STORAGE AND EDITING BOUNDARY

Table 14 maps the stored resources to their roles. In the illustrated formal evolution configuration, only `SKILL.md` and `references/tool_catalog.json` are editable. Editable content covers event anchoring, participant binding, observation intervals, evidence acquisition and composition, and stopping conditions. This file-level permission does not allow changes to legal text or thresholds quoted in procedural instructions. Statutory thresholds, calibration values, proposition semantics, query implementations, and missing-data definitions remain fixed. Query plans may change how existing capabilities are used but cannot create new runtime capabilities.

Table 14: Resources in the illustrated Law Skill bundle. Paths other than `SKILL.md` are relative to `references/`. The final column describes the formal experiment configuration.

Resource	Role	Editable
`SKILL.md`	Metadata and natural-language evidence procedure	Procedure only
`tool_catalog.json`	Query order and structured evidence plans	Plans only
`legal_articles.json`	Source regulation text and legal identity	No
`clause_evidence_profiles.json`	Clause-specific evidence requirements	No
`domain_activation_profile.json`	Domain activation information	No
`decision_schema.json`	TRACE finding format and allowed states	No

B.2 WORKFLOW AND RESOURCE SKELETONS

The following skeletons show how the five-step evidence procedure is expressed in `SKILL.md`, mirrored in a structured query plan, and linked to the finding schema. Ellipses omit regulation-specific instructions, query arguments, and schema constraints. These are structural illustrations, not complete executable files.

B.2.1 NATURAL-LANGUAGE PROCEDURE: SKILL.MD

```
---
name: <domain>-compliance
description: ...
---
# Scope and fixed resources
Regulations, evidence profiles, and fixed calibrations: ...

# Evidence procedure
## 1. Event anchoring
Applicability conditions and event-defining observations: ...
## 2. Participant binding
Required legal roles and persistent participant identities: ...
## 3. Observation-window selection
Event-relative interval, bounded by the assessment time: ...
## 4. Evidence acquisition and composition
Queries, validity checks, thresholds, and exceptions: ...
## 5. Stopping conditions
Adverse / benign / non-applicable / unresolved evidence: ...

# Output contract
Legal basis, evidence references, and findings: ...
```

B.2.2 STRUCTURED QUERY PLAN: TOOL_CATALOG.JSON

The plan names its required outputs and allowed factual queries. Its `decisive_evidence` stage implements evidence acquisition and composition.

```
{
  "skill_id": "<domain>-compliance",
  "composition_order": [...],
  "evidence_protocol": {
    "schema_version": "evidence_protocol_v2",
    "event_anchor": {
      "required_outputs": [...],
      "plan": {"op": "query", "query": "...", "args": {...}},
      "query_order": [...]
    },
    "participant_binding": {"plan": {"op": "bind", ...}, ...},
    "observation_window": {"plan": {...}, ...},
    "decisive_evidence": {"plan": {...}, ...},
    "stopping_conditions": {"plan": {...}, ...}
  }, ...
}
```

B.2.3 FINDING CONTRACT: DECISION_SCHEMA.JSON

The schema links conclusions to regulations and evidence references. Its omitted conditional constraints couple applicability to the permitted judgment states.

```
{
  "type": "object",
  "required": ["skill_id", "summary", "findings"],
  "properties": {...},
  "$defs": {
    "finding": {
      "properties": {
        "article_key": {...}, "applicability": {...},
        "status": {"enum": ["compliant", "non_compliant",
                            "not_triggered", "unknown"]},
        "legal_basis": {...}, "query_refs": {...},
        "supporting_atoms": {...}, "missing_evidence": {...}, ...
      },
      "allOf": [...], ...
    }, ...
  }, ...
}
```

C Scoring Conventions and Statistical Analysis

Targets and exact matching. Each case is scored against its designated review target. The main tables use binary targets: compliant or violation. Strict Accuracy is the fraction of all cases whose predicted target-review status exactly matches the target. Predictions of `unknown`, `not_triggered`, or a missing target review are errors; these cases remain in the evaluation denominator.

Class metrics. For each class $c \in \{C, V\}$ (compliant or violation), let TP_c count correct predictions of c, FP_c count predictions of c for other targets, and FN_c count targets of c assigned any other status or no review. Recall is $TP_c/(TP_c + FN_c)$, and $F1_c = 2TP_c/(2TP_c + FP_c + FN_c)$. Macro-F1 is the unweighted mean of the two class F1 scores, not an average across regulations or across four output states. A zero denominator yields zero. Unknown and not-triggered predictions are therefore penalized as false negatives for their target class without being treated as additional classes in the macro average.

False positive rate. With violations as the positive class, let N_C be the number of compliant reference cases and $N_{C\to V}$ the number predicted as violations. Then

$$\mathrm{FPR} = \frac{N_{C\to V}}{N_C}, \qquad 1 - \mathrm{Recall}_C = \mathrm{FPR} + \frac{N_{C\to U}}{N_C},$$

where $N_{C\to U}$ counts compliant cases assigned unknown, not-triggered, or no review. Thus FPR equals $1 - \mathrm{Recall}_C$ only when all compliant cases receive a binary prediction. Retaining abstentions in the denominator makes Strict Accuracy and coverage necessary complements to FPR.

Coverage and query counts. Review-target recall is the fraction of cases with a finding matching the designated review identity, regardless of its predicted status. Unknown rate and not-triggered rate divide their respective target-review prediction counts by all evaluated cases; a missing review is a separate outcome, not automatically an unknown prediction. Queries per case divide recorded agent-issued runtime evidence queries by the case count. They exclude preparation of the initial evidence package and do not count every MCP call. Thus, the fixed-evidence ablation records zero additional runtime queries while still receiving initial evidence. The counts describe the saved runs rather than a complete measure of preprocessing or deployment cost.

Learning efficiency and paired statistics. Normalized AULC integrates Strict Accuracy across cumulative development-set fractions $[0, 1/4, 1/2, 3/4, 1]$ using the trapezoidal rule and divides by the budget span. The same fraction is applied to each regulation, and the integration includes the initial seed at zero development budget. Test checkpoint results are used only for offline reporting, never for skill selection. For the comparison between the seed and validation-selected skills with the full development budget on the same 543 cases, the Strict Accuracy gain is 34.8 percentage points. A paired bootstrap with 20,000 resamples (random seed 20260911) gives a 95% confidence interval of $[+29.5, +40.0]$ percentage points. A two-sided exact McNemar test gives $p = 1.03 \times 10^{-31}$. Tool traces, evidence chains, skill versions, and validation records are retained with the experimental artifacts.

D CASE STUDIES

The following cases show domain-agent inputs, tool evidence, and final findings. Times are relative to the start of each driving segment; camera overlays were removed using generative inpainting.

D.1 CASE 1: LANE CHANGE WITH PERSISTENT PARTICIPANT TRACKING



$t = 7.89$ s



$t = 10.29$ s



$t = 12.69$ s



BEV

Selected calls from an independent case-study rerun; tool returns are excerpted.

Initial input

Task setup: lane-change-compliance; analysis_frame=220; replay from frame 0.
System context: domain skill instructions, output JSON schema, and tool interfaces.

Task instruction (recorded excerpt): Review exactly the lane-change-compliance domain in this isolated context. Call compliance_open_domain_review exactly once with the supplied CSV, frame, and skill ID. [...]

Router handoff (recorded): Retained-history lane-position transition observed at frame 216 and active right boundary contact at frame 220; the episode boundary for these cues must be resolved by the domain Skill.

```
initial_query = "lane.transition_atoms"
```

Historical snapshot interface: arguments.frame_index. The recorded input includes a generic temporal-argument check instruction.

Evidence-call defaults shown once: session_id="S", detail="compact". S is returned by the opening call.

Step 1 — 1 tool call

Agent → Tool: `compliance_open_domain_review`

```
{"csv_path": "<driving-record>", "analysis_frame": 220, "skill_id": "lane-change-compliance"}
```

Tool → Agent

```
analysis_frame = 220; frames_replayed = 221; warning_count = 0
scene_dossier.scene_summary.ego = {"speed_mps": 24.043404, "yaw_rate_radps": -0.004349,
"turn_signal": "left", "turn_intention": "unknown", "navigation_instruction": "straight"}
lane_observations = {"lane_position": "same_direction_leftmost",
"latest_lane_position_transition_frame_index": 216, "right_boundary_contact": true}
... (remaining dossier, data_quality, skill_contract, clause_index)
```

Step 2 — 1 tool call

Agent → Tool: `compliance_query_evidence`

```
query_name="lane.boundary_context_window"
arguments={"frame_index": 220, "lookback_frames": 60, "outside_clearance_threshold_m": -0.25}
```

Tool → Agent

```
query_ref = "query_001"; frame_index = 220
object_evidence.spatial_role_segments = {"same_lane.nearest_ahead": [{"start_frame_index": 161,
"end_frame_index": 215, "track_id": 33014}], "left_adjacent_lane.nearest_behind":
[{"start_frame_index": 161, "end_frame_index": 215, "track_id": 32761}],
"same_lane.nearest_behind": [{"start_frame_index": 216, "end_frame_index": 220, "track_id":
32761}]}
geometry_evidence.boundary_contact_sequences[0] = {"boundary_track_id": "lane_boundary_0001",
"first_footprint_intersection_frame_index": 190, "first_outside_threshold_frame_index": 195,
"slot_change_events": [{"frame_index": 216, "from_observation_slot": "left",
"to_observation_slot": "right"}]}
state_evidence.lane_position_transition_events = [{"frame_index": 216, "previous": null,
"current":
null, "direction": "left"}]
...
```

Step 3 — 1 tool call

Agent → Tool: `compliance_query_evidence`

```
query_name="distance.object_longitudinal_relation"
arguments={"frame_index": 194, "track_id": 33014}
```

Tool → Agent

```
query_ref = "query_002"; frame_index = 194
object_evidence.selected_object_identity.track_id = 33014
object_evidence.selected_object_lane_assignment.geometry.label = "same_lane"
numeric_evidence.constant_velocity_time_to_longitudinal_body_edge_alignment_s = 4.876419650400851
...
```

Case 1 — Lane change: tracking the same rear vehicle

Selected evidence-acquisition steps — Follow-up calls and final answer

Step 4 — 1 tool call

Agent → Tool: `compliance_query_evidence`

```
query_name="object.observation_window"
arguments={"track_id": 32761, "start_frame_index": 190, "end_frame_index": 220}
```

Tool → Agent

```
query_ref = "query_003"; frame_index = 220
semantic_view.observed_lane_assignment_sequence = ["left_adjacent_lane", "same_lane"]
state_evidence.window.observation_coverage_ratio = 1.0
...
```

Step 5 — 1 tool call

Agent → Tool: `compliance_query_evidence`

```
query_name="distance.object_calibrated_clearance_window"
arguments={"track_id": 32761, "start_frame_index": 190, "end_frame_index": 220,
"base_clearance_m":
13.6, "relative_speed_coefficient_s": 3.4, "minimum_clearance_floor_m": 0,
"maximum_clearance_cap_m": 50}
```

Tool → Agent

```
query_ref = "query_004"; frame_index = 220
numeric_evidence.minimum_margin_observation = {"frame_index": 192,
"measured_longitudinal_clearance_m": 27.085359500000003, "calculated_clearance_m": 27.9848798,
"clearance_margin_m": -0.8995202999999989}
...
```

Step 6 — 1 tool call

Agent → Tool: `compliance_query_evidence`

```
query_name="lane.temporal_window"
arguments={"frame_index": 190, "lookback_frames": 30}
```

Tool → Agent

```
query_ref = "query_006"; frame_index = 190
state_evidence.observations (selected fields) = [{"frame_index": 190, "turn_signal": {"state":
"left", "observation_status": "known", "elapsed_s": 2.0500000000000007}}]
...
```

Step 7 — 3 tool calls

Agent → Tool

```
compliance_get_clause_material(skill_id="lane-change-compliance", article_key=...)
```

One call per row; clause text is translated into English.

article_key	Tool: legal_basis.original_text
IMR_44_1	On roads with two or more motor-vehicle lanes in the same direction, a vehicle changing lanes must not impede the normal travel of motor vehicles in the relevant lanes.
IMR_57_2	When changing lanes to the left, the left turn signal must be activated in advance.
IMR_57_7	When changing lanes to the right, the right turn signal must be activated in advance.

...

Step 8 — 1 tool call

Agent → Tool: `compliance_resolve_legal_citations`

```
{"citations": [{"skill_id": "lane-change-compliance", "article_key": "IMR_44_1"}, {"skill_id":
"lane-change-compliance", "article_key": "IMR_57_2"}, {"skill_id": "lane-change-compliance",
"article_key": "IMR_57_7"}]}
```

Tool → Agent

```
all_resolved = true; unresolved_citations = []
...
```

Final answer

article_key	status	query_refs
IMR_44_1	Violation	query_001, query_002, query_003, query_004
IMR_57_2	compliant	query_001, query_006
IMR_57_7	not_triggered	query_001, query_006

reason (excerpt; status label normalized):

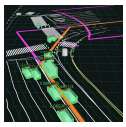
... Decisive minimum-margin observation at frame 192: measured longitudinal clearance 27.085 m, ego speed 23.967 m/s, target speed 28.198 m/s, required minimum clamp(13.6-3.4*(23.967-28.198),0,50)=27.985 m, signed margin $-0.900\,\mathrm{m} < 0$, i.e. below the project calibration during the lane change. One confirmed unsafe target-rear interaction is sufficient, so the lane change affected the relevant vehicle's normal travel: violation.

D.2 CASE 2: RIGHT TURN WITH CONFLICTING MANEUVER CUES


$t = 9.26$ s


$t = 10.26$ s


$t = 11.66$ s


BEV

All domain tool calls are retained; returns are excerpted. Images show the scene. Intermediate reasoning text was not archived.

Initial input

Task setup: traffic-light-compliance; analysis_frame=274; replay from frame 0.
System context: domain skill instructions, output JSON schema, and tool interfaces.

Task instruction (runner excerpt): Review exactly the traffic-light-compliance domain in this isolated context. Call compliance_open_domain_review exactly once with the supplied CSV, frame, and skill ID. [...]

Router handoff (recorded): A preserved stopline crossing is observed (latest_stopline_crossing_observed=true, frame 213) together with retained stopline contact (latest_stopline_contact_observed=true, frame 199) and clearance (frame 216). Although the signal channel itself is unknown and stopline_exists=false, the retained stopline crossing event is a positive activation cue for review.

```
initial_query = "traffic.signal_atoms"
```

Setup reconstructed from the job and runner; the initial API prompt was not archived.

Evidence-call defaults shown once: session_id="S", detail="compact". S is returned by the opening call.

Round 1 — 1 tool call

Agent → Tool: `compliance_open_domain_review`

```
{"csv_path": "<driving-record>", "analysis_frame": 274, "skill_id":
"traffic-light-compliance"}
```

Tool → Agent

```
analysis_frame = 274; frames_replayed = 275; warning_count = 0
scene_dossier.scene_summary.ego = {"speed_mps": 7.199653, "yaw_rate_radps": -0.270924,
"turn_signal": "left", "turn_intention": "unknown", "navigation_instruction": "straight"}
signal_observations = {"stopline_exists": false, "latest_stopline_contact_frame_index": 199,
"latest_stopline_crossing_frame_index": 213, "latest_stopline_clearance_frame_index": 216}
... (remaining dossier, data_quality, skill_contract, clause_index)
```

Round 2 — 1 tool call

Agent → Tool: `compliance_query_evidence`

```
query_name="traffic.signal_atoms"
arguments={"frame_index": 274}
```

Tool → Agent

```
query_ref = "query_001"; frame_index = 274
semantic_view.traffic_light.observation_status = "unknown"
...
```

Archived event snapshots inside this one return

Event	Frame	Traffic lights (left/straight)	Indicator	Heading (rad)	Yaw (rad/s)
contact	199	red light / red light	right	2.193835	-0.209079
crossing	213	red light / red light	right	2.05596	-0.18562
clearance	216	red light / red light	off	2.028869	-0.172825

```
stopline_contact.latest_snapshot.ego_navigation_instruction.movement = "straight"
stopline_contact.latest_snapshot.ego_turn_signal.elapsed_s = 10.000000000000007
stopline_crossing.latest_snapshot.ego_turn_signal.elapsed_s = 10.700000000000017
stopline_crossing.latest_snapshot.channels.right.observation_status =
"no_active_directional_signal_observed"
```

Case 2 — Right turn: resolving conflicting maneuver cues

Recorded domain-agent execution — Follow-up calls and final answer

Round 3 — 2 tool calls

Agent → Tool: compliance_query_evidence

```
query_name="traffic.signal_atoms"
arguments={"frame_index": 186}
```

Tool → Agent

```
query_ref = "query_002"; frame_index = 186
semantic_view.traffic_light.channels = {"left": {"state": "red", "observation_status":
"known_signal_state"}, "straight": {"state": "red", "observation_status":
"known_signal_state"},
"right": {"state": "unknown", "observation_status": "no_active_directional_signal_observed"}}
semantic_view.stopline_contact.latest_snapshot = null
...
```

Agent → Tool: compliance_query_evidence

```
query_name="traffic.signal_atoms"
arguments={"frame_index": 250}
```

Tool → Agent

```
query_ref = "query_003"; frame_index = 250
traffic_light.observation_status = "unknown"
stopline_contact.latest_snapshot.frame_index = 199
stopline_crossing.latest_snapshot.frame_index = 213
stopline_contact.latest_end_snapshot.frame_index = 216
...
```

Round 4 — 1 tool call

Agent → Tool: compliance_get_clause_material

```
{"skill_id": "traffic-light-compliance", "article_key": "IMR_38_3"}
```

Tool → Agent

```
article_key = "IMR_38_3"
legal_basis.original_text = "When the red light is on, vehicles must not proceed. Vehicles
turning
right may proceed." (English translation)
... (evidence_contract)
```

Round 5 — 1 tool call

Agent → Tool: compliance_resolve_legal_citations

```
{"citations": [{"skill_id": "traffic-light-compliance", "article_key": "IMR_38_3"}]}
```

Tool → Agent

```
all_resolved = true; unresolved_citations = []
...
```

Round 6 — Final answer

findings (selected fields)

article_key	status	query_refs
IMR_38_3	compliant	query_001, query_002, query_003

reason (Chinese quotation translated):

Contact ordering is resolved: first stopline footprint contact at frame 199, reference-point crossing at frame 213, and full-footprint clearance at frame 216. At all three anchors the observed left and straight channels were red (known_signal_state), so a straight or left passage would be prohibited. The event-aligned movement is resolved as a right turn: turn intention is unknown and the navigation snapshot reads straight, but the vehicle carried a sustained right turn signal (elapsed 10.0 s at contact, 10.7 s at crossing, only cancelled just after clearance) together with a consistent heading decrease ($2.1938 \rightarrow 2.0560 \rightarrow 2.0289$ rad) and consistent negative yaw (-0.209/-0.186/-0.173 rad/s) through contact, crossing and clearance, which under the loaded evidence contract establishes the executed turn and overrides the conflicting navigation snapshot. The applicable dedicated right-turn channel reported no_active_directional_signal_observed, i.e. no active dedicated right-turn signal was observed, and the circular/straight channel is not required as the controlling right-turn channel. Accordingly the article's right-turn exception ('Vehicles turning right may proceed') governs: the red-phase stopline passage is consistent with the applicable red vehicle signal, and the review is resolved as compliant without requiring a retained later clearance snapshot. The separate question of whether the right turn obstructed released vehicles or pedestrians is not treated as satisfied here.

```
missing_evidence = ["Event-aligned general_vehicle_signal color was not directly observed (right
channel and general channel both no_active_directional_signal_observed); red phase is inferred
from the observed left/straight channels at contact, crossing and clearance.", "Whether the
right
turn obstructed released vehicles or pedestrians is not established here and belongs to a
separate
right-of-way review (e.g. IMR 38.6)."]
```