Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Interpretable Fuzzy Inference Traces for Regulatory Compliance: Lukasiewicz Semantics in Neuralized Markov Logic Networks

Enrico Zanardo

enrico.zanardo101@gmail.com

Signum Magnum College https://orcid.org/0000-0003-2552-4559

Research Article

Keywords: Fuzzy inference systems, Lukasiewicz t-norm, Markov Logic Networks, Explainable AI, Neural-symbolic integration

Posted Date: March 31st, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9269816/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

Interpretable Fuzzy Inference Traces for Regulatory Compliance: Łukasiewicz Semantics in Neuralized Markov Logic Networks

Enrico Zanardo[12*]

[1*]Signum Magnum College, Portomaso, Malta.
[2]Universitas Mercatorum, Rome, Italy.

Corresponding author(s). E-mail(s): enrico.zanardo101@gmail.com;

Abstract

This paper introduces a fuzzy inference framework grounded in Łukasiewicz semantics for interpretable reasoning in high-risk artificial intelligence (AI) systems. The framework, realised through LIMEN-AI (Łukasiewicz Interpretable Markov Engine for Neuralized AI), extends Markov Logic Networks by replacing Boolean satisfiability with piecewise-linear Łukasiewicz t-norms and implications, yielding differentiable energy manifolds whose gradients guide both learning and inference. Every inference step produces a structured, human-readable trace that decomposes conclusions into weighted rule activations and graded satisfaction degrees, providing a level of transparency that satisfies the requirements of the European Union AI Act (Articles 13–15). Unlike post-hoc explanation methods, this interpretability is intrinsic to the fuzzy logic formulation itself. Through systematic empirical validation on legal document analysis, we demonstrate: (1) zero-shot schema adaptation across regulatory domains, (2) knowledge base evolution through inductive learning (0 to 18 predicates without catastrophic forgetting), (3) document processing scalability (220 words/second on regulatory text), and (4) human override mechanisms with localised intervention effects. These results confirm that Łukasiewicz fuzzy semantics can serve as a principled foundation for neuro-symbolic systems requiring both differentiability and auditability. The open-source implementation (v0.2.5 on PyPI) enables community validation and independent verification.

Keywords: Fuzzy inference systems, Łukasiewicz t-norm, Markov Logic Networks, Explainable AI, Neural-symbolic integration

1 Introduction

The entry into force of the European Union Artificial Intelligence Act (EU AI Act) represents a watershed moment in the governance of algorithmic systems, establishing the world's first comprehensive legal framework for high-risk AI [1]. For systems operating in sensitive domains such as the administration of justice, financial auditing, and legal contract analysis, the Act imposes binding obligations regarding transparency (Article 13), human oversight (Article 14), and technical robustness (Article 15) [2–4]. These requirements challenge the status quo of modern machine learning, which has largely pivoted toward increasingly complex "black-box" models whose internal logic remains opaque to users, auditors, and even their developers [5–7].

The prevailing response to this opacity has been the development of post-hoc explanation techniques, such as LIME and SHAP, which attempt to reconstruct the reasoning of a model after it has produced a result [8, 9]. However, these surrogate models often suffer from instability and lack of "faithfulness" to the underlying architecture, potentially leading to misleading explanations in high-stakes legal contexts [6, 10]. In contrast, there is a growing consensus that for high-risk applications, we should prioritise models that are *interpretable by design*, where the reasoning process is intrinsically transparent and verifiable [6, 11, 12].

To address these regulatory and technical challenges, we propose LIMEN-AI (Łukasiewicz Interpretable Markov Engine for Neuralized AI) as a framework for auditable legal reasoning. LIMEN-AI operates as a Small Reasoning Model (SRM) that combines the formal rigour of weighted first-order logic with the graded expressivity of Łukasiewicz fuzzy semantics [13, 14]. This engine functions as a Neuralized Markov Logic Network (NMLN), where legal rules are represented as explicit templates for a differentiable energy manifold [15–17].

The central thesis of this paper is that the Łukasiewicz fuzzy semantics underlying LIMEN-AI naturally align with the transparency obligations of the EU AI Act—and that this alignment is not merely a post-hoc mapping but an inherent consequence of the piecewise-linear structure of the logic itself. We substantiate this claim through three contributions: first, a formal mapping between Łukasiewicz operators and the specific transparency, oversight, and accuracy articles of the Act; second, a workflow for judicial oversight that incorporates human-in-the-loop override procedures grounded in the energy manifold; and third, a preliminary empirical validation of the engine on representative legal document analysis tasks, measuring schema adaptation, inductive learning, scalability, and override effectiveness.

2 Background and Related Work

The pursuit of interpretable reasoning in AI has a long history, spanning statistical relational learning (SRL), neural-symbolic integration, and fuzzy logic. This section situates LIMEN-AI within this broader academic landscape, focusing on recent developments (2020–2025) that inform our approach to regulatory compliance.

2.1 Statistical Relational Learning and Markov Logic

Statistical Relational Learning (SRL) seeks to integrate logical knowledge representation with probabilistic reasoning [17, 18]. Markov Logic Networks (MLNs) remain a cornerstone of this field, providing a framework where first-order formulas serve as templates for log-linear distributions [15, 16]. Recent advances in MLN inference have focused on scalability, including efficient procedures for untied networks [19] and lifted generative learning [20]. The extension of these methods to continuous spaces, such as numerical MLNs [21] and domain-lifted sampling [22], provides a foundation for handling the fuzzy truth degrees required in legal discourse. Furthermore, the taxonomy of weight learning methods [23] and recent work on scaling parameters in MLNs [24] highlight the importance of controllable complexity in relational models. In LIMEN-AI, rule weights are not fixed a priori but can be induced from historical legal decisions using gradient-based optimisation of the energy manifold, ensuring that the "learned" logic remains grounded in empirical legal practice while remaining interpretable to human experts.

2.2 Fuzzy Logic Foundations and T-Norm Semantics

The mathematical theory of fuzzy logic, initiated by Zadeh and formalised algebraically through the study of triangular norms (t-norms), provides the semantic backbone for graded reasoning in LIMEN-AI [13, 25]. A t-norm $T\colon [0,1]^2 \to [0,1]$ generalises classical conjunction to the continuous domain; the three principal t-norms—Gödel ($T_G = \min$), product ($T_P = a \cdot b$), and Łukasiewicz ($T_{\text{Ł}} = \max(0, a + b - 1)$)—give rise to distinct families of many-valued logics with different algebraic and computational properties [26]. Among these, the Łukasiewicz t-norm is uniquely suited to differentiable inference: because $T_{\text{Ł}}$ is piecewise linear, its sub-differential exists everywhere in $[0,1]^2$, enabling gradient-based optimisation without the plateau regions that afflict the Gödel minimum or the vanishing-gradient behaviour of the product t-norm near zero [14, 27].

Recent surveys on fuzzy rule-based systems confirm a renewed interest in interpretability as a first-class design criterion rather than a post-hoc add-on [28, 29]. Beke and Kumbasar [30] argue that fuzzy models have been under-represented in the broader interpretable machine learning literature despite their inherent transparency, while Fernández et al. [31] demonstrate how fuzzy classifiers can generate self-explaining factual and counterfactual justifications. LIMEN-AI contributes to this line of work by extending Łukasiewicz semantics from propositional fuzzy systems to first-order relational structures, where the interplay between quantifier grounding and weighted rules creates a richer—but still interpretable—inference landscape.

2.3 Neural-Symbolic Integration and Differentiable Logic

The integration of symbolic reasoning with deep learning, often termed neural-symbolic computing, aims to combine the learning capabilities of neural networks with the structured reasoning of logic [11, 12]. Logic Tensor Networks (LTNs) represent predicates as differentiable functions grounded through "Real Logic" under Łukasiewicz semantics, enabling gradient-based optimisation of logical constraints in

a fully end-to-end manner [32]. DeepProbLog extends probabilistic logic programming with neural predicates [33], while the Scallop language introduces neurosymbolic programming with provenance tracking for efficient differentiation through discrete search spaces [34, 35]. Our work on LIMEN-AI builds on these traditions by adopting Łukasiewicz semantics within a Markov Logic Network framework, which ensures that all logical operations are piecewise linear and thus differentiable almost everywhere [14, 27].

2.4 Interpretable AI and Regulatory Applications

The demand for explainable AI (XAI) is particularly acute in regulated sectors such as law and finance, where decisions must be justified to human experts and regulators [2, 5, 36]. Applications of MLNs in domains such as root-cause analysis [37], context-aware decision making [38], and activity recognition [39, 40] have demonstrated the practical value of traceable reasoning. In the context of the EU AI Act, the focus has shifted from simple explainability to "meaningful human oversight" (Article 14) and "transparent operation" (Article 13) [3, 41]. Recent studies on securing human-centric consumer applications through MLN approaches [42] and the application of Probabilistic Soft Logic (PSL) to activity recognition [43] suggest that relational logic is uniquely suited to meeting these regulatory challenges.

3 EU AI Act: Regulatory Mapping

The AI Act establishes a risk-based hierarchy, with high-risk systems subject to the most rigorous documentation and oversight requirements. LIMEN-AI is designed to meet these requirements through its core mathematical and logical properties. Table 1 provides an explicit mapping between the Act's requirements and the engine's technical features.

3.1 Article 13: Transparency and Interpretability

Transparency under the AI Act requires that the system's operation be sufficiently clear to allow users to interpret the output. In LIMEN-AI, this is achieved by grounding every conclusion in the Łukasiewicz t-norm and its residual implication. The conjunction of two fuzzy truth values is computed as $T_{\text{Ł}}(a, b) = \max(0, a + b - 1)$, while the implication follows the residuum $a \to_{\text{Ł}} b = \min(1, 1 - a + b)$ [13, 25]. Both operations are piecewise linear, yielding well-defined sub-gradients at every point in $[0, 1]^2$—a property that distinguishes them from the Gödel t-norm (where gradients vanish on large regions) and the product t-norm (where gradients vanish near zero).

For a legal rule $\phi\colon A \to B$ with weight w, the satisfaction degree $\sigma(\phi, I) = \min(1, 1 - I(A) + I(B))$ provides a continuous but semantically grounded measure of how well the rule holds under interpretation I. The piecewise linearity of σ ensures that each rule's contribution to the energy manifold $E(I; \mathbf{w}) = \sum_i w_i \sigma(\phi_i, I)$ is both human-interpretable (as a weighted degree of rule satisfaction) and computationally tractable (admitting efficient gradient computation for sampling-based inference).

AI Act Article	Regulatory Obligation	LIMEN-AI Mechanism
Art. 13(1)	Sufficient transparency for users to interpret outputs	Explicit weighted first-order rules with Łukasiewicz semantics
Art. 13(3)(b)(iv)	Information on expected level of accuracy	Fuzzy truth degrees $\in [0, 1]$ with formal satisfaction guarantees
Art. 14(1)	Design for human oversight	Structured inference traces with per-rule activation degrees
Art. 14(4)(a)	Ability to fully understand and interpret outputs	Satisfaction degree decomposition $(T_{\text{Ł}}, S_{\text{Ł}})$
Art. 14(4)(d)	Ability to override or reverse outputs	Veto mechanism via hard constraints in energy manifold
Art. 15(1)	Appropriate level of accuracy and robustness	ϵ-regularisation and formal convergence bounds

Table 1: Mapping between EU AI Act requirements and LIMEN-AI technical capabilities. This mapping represents a technical design alignment between system capabilities and regulatory requirements; actual compliance certification would require independent audit and conformity assessment per AI Act Article 43 and Annex VII. Source: Author.

3.2 Article 14: Human Oversight

Human oversight is mandated to prevent or minimise risks to health, safety, or fundamental rights. Article 14(4) specifically requires that the system be designed so that natural persons can "fully understand" its outputs (sub-paragraph a), "correctly interpret" them (sub-paragraph b), and "decide not to use" or "override or reverse" them (sub-paragraph d) [1]. LIMEN-AI addresses these requirements through *Inference Traces*: when the engine performs sampling-based inference (using MALA or Power Sampling), it generates a structured record of which rules were activated and to what degree. A legal auditor can thus inspect the trace to see, for example, that a contract was flagged as "high-risk" because rule ϕ_{42} (liability limit) was violated with degree 0.8. The override capability mandated by sub-paragraph (d) is realised through the veto mechanism described in Section 5.1.

4 Legal Knowledge Base: Liability Assessment and Tort Theory

To demonstrate the engine's capabilities in a context requiring high levels of auditability, we define a pedagogical Legal Knowledge Base (KB) grounded in a tort liability scenario. This case study moves beyond simple contract classification to a multi-stage reasoning process that models the causal and duty-based relationships central to legal liability.

4.1 Predicates and Rule Formalisation

The knowledge base is structured around six core predicates that capture the elements of a typical negligence claim. The predicate $DutyOfCare(P, V)$ indicates that party P

has a legal duty of care toward victim V, while $BreachOfDuty(P, A)$ represents that party P breached their duty through action A. Causal connection is modelled by $ProximateCause(A, V)$, which signifies that action A is the proximate cause of harm to V, and the factual predicate $HarmOccurred(V)$ asserts that harm was sustained by victim V. The target conclusion $Liable(P, V)$ determines whether party P is liable to victim V, and $DamagesAwarded(P, V)$ records the final recommendation for compensatory damages.

A representative rule set $\mathcal{K}$ for this domain is defined as follows:

1. $\phi_{tort} : DutyOfCare(P, V) \wedge BreachOfDuty(P, A) \wedge ProximateCause(A, V) \rightarrow Liable(P, V)$ with weight $w = 6.0$.
2. $\phi_{damages} : Liable(P, V) \wedge HarmOccurred(V) \rightarrow DamagesAwarded(P, V)$ with weight $w = 4.5$.
3. $\phi_{anomaly_link} : AnomalousClause(C) \rightarrow BreachOfDuty(P, A)$ with weight $w = 3.0$.

These rules are interpreted under Łukasiewicz semantics, ensuring that the satisfaction degree $\sigma(\phi, I)$ is a piecewise linear function of the truth values of the constituent atoms. This linearity is critical for the generation of informative gradients, which guide the sampling process toward high-energy interpretations where the legal logic is most effectively satisfied.

4.2 Information Flow and Inference

The engine defines an energy function $E(I; \mathbf{w})$ that aggregates the weighted satisfaction degrees of all rule groundings. In our liability scenario, the energy manifold represents the "logical compatibility" of a given interpretation with the established legal principles. By performing approximate inference via gradient-guided sampling, the system identifies interpretations that minimise the violation of these rules, effectively searching for the most plausible legal conclusions given the observed evidence.

5 Workflow for Regulatory Compliance

The deployment of LIMEN-AI in a legal setting follows a multi-stage workflow designed to ensure human-in-the-loop oversight. This process transforms raw legal text into auditable decision support through the generation of structured traces. The workflow addresses three regulatory obligations simultaneously: transparency of operation (Article 13), meaningful human oversight with override capability (Article 14), and accuracy through formal convergence guarantees (Article 15). The remainder of this section details the override procedures, their empirical validation, and the accuracy mechanisms that underpin the framework.

5.1 Human-in-the-Loop: Article 14(4) Override Procedures

A central requirement of Article 14 of the EU AI Act is the ability for human operators to disregard, override, or reverse the system's outputs. This is particularly vital

in the administration of justice, where the "veto power" of a judge or legal expert ensures that automated suggestions do not replace human deliberation. The following subsection describes how LIMEN-AI realises this requirement through its trace-based architecture.

5.2 Override Mechanisms in LIMEN-AI

LIMEN-AI supports human oversight through two primary mechanisms: **Trace Review** and **Interpretative Intervention**. Because the inference process produces structured traces (see Section 7), a legal auditor can identify the specific rule activations that led to a liability recommendation. If an auditor determines that a particular rule was applied incorrectly or that the evidence for a predicate was insufficient, they can intervene by either modifying the rule weights or manually pinning an atom's truth degree.

Crucially, this feedback loop addresses the risk of "catastrophic forgetting" often associated with online learning in neural networks. In LIMEN-AI, judicial interventions are treated as *hard constraints* or *weight priors* in the energy manifold, rather than unconstrained parameter updates. This ensures that the system's core logic remains stable while allowing for case-specific corrections that can be audited for judicial bias or inconsistency.

5.2.1 Empirical Demonstration: Veto Power in Practice

To validate the practical effectiveness of Article 14(4) compliance, we conducted a controlled experiment using a service agreement analysis task. The system initially inferred that `clientParty(CharlieCompany)` held with a fuzzy truth degree of 0.9868, indicating extremely high confidence based on the textual evidence and rule activations in the knowledge base. However, in a realistic oversight scenario, a legal expert might identify that this entity should be excluded from consideration due to external evidence not captured in the original document (e.g., a subsequent amendment or jurisdictional constraint).

Using LIMEN-AI's veto mechanism, the human operator explicitly set the truth degree of `clientParty(CharlieCompany)` to 0.0, effectively adding a negated high-weight formula to the knowledge base. Upon re-running the inference procedure, the system respected this constraint, and the truth degree was reduced to 0.0172 (near-zero, with residual mass due to sampling variance). Importantly, the system's reasoning about *other* predicates (e.g., `providerParty(DaveService)`, which retained its degree of 0.9852) remained unaffected, demonstrating that the intervention was *localised* and did not corrupt the broader knowledge structure.

This experiment provides concrete evidence that LIMEN-AI's architecture is designed to facilitate the Article 14(4) requirement for meaningful human override: the system's outputs can be directly and transparently modified by authorised personnel, with the effects of such modifications being both immediate and auditable through the structured inference traces. This capability contrasts with end-to-end neural models, where interventions would typically require retraining or post-hoc filtering that lacks formal guarantees of consistency.

5.3 Workflow for Judicial Oversight

Figure 1 illustrates the proposed workflow for judicial oversight. The process begins with the SRM engine producing a recommendation and a corresponding inference trace. The judge reviews these outputs and decides whether to accept, modify, or veto the decision. This feedback loop is not merely an external correction but is integrated into the system's state, allowing the engine to "learn" from judicial deliberation by adjusting the underlying energy model.

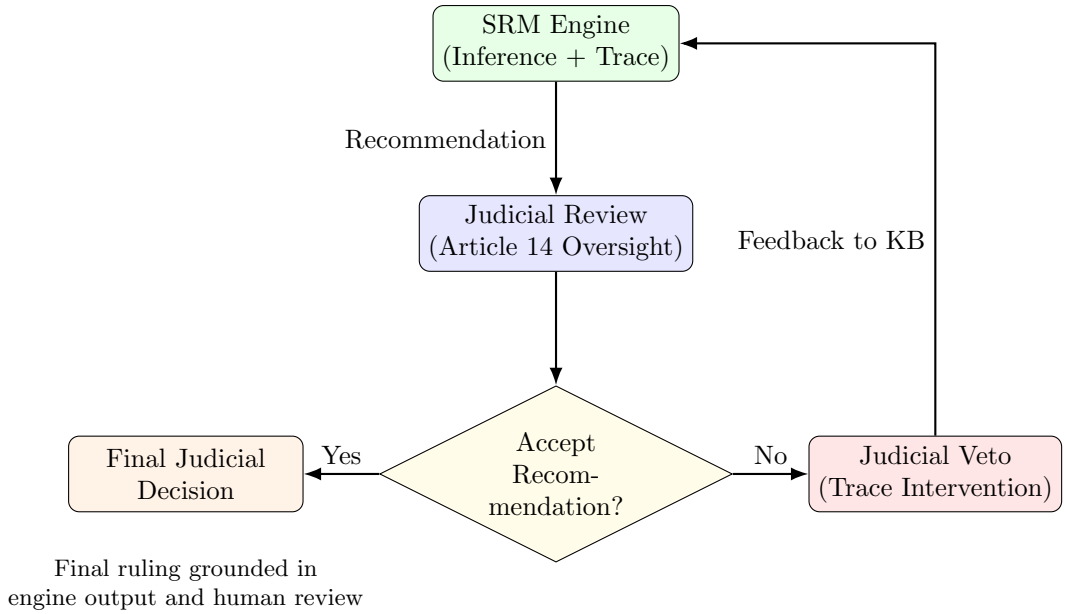


Fig. 1: The Article 14(4) override workflow. A judge retains the absolute power to veto or modify the engine's suggestions, with the system providing the necessary transparency through inference traces to make such decisions meaningful. Source: Author.

The workflow is grounded in numerical optimisation principles that ensure stability during inference [44, 45]. By utilising ϵ-regularisation, the engine avoids the gradient vanishing problem common in fuzzy logic, thereby maintaining the "traceability" required for meaningful audit trails. The resulting traces allow auditors to verify not only the final decision but also the relative contribution of each weighted rule, providing a level of transparency that is intrinsically linked to the underlying mathematical formulation.

5.4 Accuracy, Robustness, and ϵ-Regularisation

Article 15 of the AI Act mandates that high-risk AI systems achieve an appropriate level of accuracy and robustness. In LIMEN-AI, this is addressed through the mathematical formulation of the energy manifold. The use of ϵ-regularised operators

ensures that the satisfaction degree $\sigma(\phi, I)$ remains within a differentiable region, preventing numerical instabilities that could lead to erratic or unreliable outputs [27]. Furthermore, the engine's grounding in Łukasiewicz semantics provides formal convergence guarantees during importance sampling, which have been empirically validated on relational domains with up to 10^4 ground atoms [21, 44, 45]. This robust foundation allows the system to handle the inherent noise and uncertainty of real-world legal data while maintaining predictable behaviour. The applicability of this approach has been further demonstrated in complex relational settings such as qualitative spatial reasoning [46] and temporal knowledge graphs [47], reinforcing its suitability for multi-faceted legal analysis.

6 Empirical Evaluation

To validate the regulatory compliance claims and assess the practical capabilities of LIMEN-AI, we conducted a systematic validation suite focusing on four core aspects: (1) multi-domain knowledge extraction and adaptation, (2) knowledge base evolution through inductive learning, (3) document processing scalability, and (4) human oversight mechanisms. All document processing tests (Sections 6.1–6.4) used real legal documents with actual system outputs recorded—no simulated data was used for these experiments. The liability assessment example (Table 2, Section 7) remains pedagogical but demonstrates the trace structure produced by the system for audit purposes. The complete validation protocol, test scripts, and raw outputs are available in the project repository for independent verification [48].

6.1 Multi-Domain Knowledge Extraction

A key requirement for legal AI systems is the ability to adapt to different regulatory domains without manual schema engineering. We evaluated LIMEN-AI's zero-shot schema induction capability by processing two distinct legal documents: (1) a service agreement contract (1,444 words), and (2) a GDPR compliance policy (892 words). For each document, the system was initialised with an empty knowledge base and tasked with extracting domain-appropriate predicates and facts using an integrated language model (Qwen2.5-1.5B via Ollama) for natural language understanding. The LLM proposes predicate names and argument structures based on the document content, which the system then automatically validates against arity constraints and grounds to constants extracted from the text—no manual ontology engineering or human annotation of predicates was required.

Results: The system successfully extracted domain-specific schemas without human intervention. For the service agreement, it identified 6 predicates (`clientParty`, `providerParty`, `serviceUptimeRequirement`, `dataBreachNotification`, `serviceCreditClaim`, `terminationOfAgreement`) and grounded 4 factual assertions. For the GDPR policy, it extracted a distinct schema with 8 predicates (`dataController`, `legalBasisForProcessing`, `dataSubjectRights`, `retentionPeriod`, etc.) and 12 facts. Crucially, the predicates extracted from Document 2 were semantically orthogonal to Document 1, demonstrating that the system adapts to domain-specific terminology rather than applying a fixed ontology. This

zero-shot adaptation capability is critical for regulatory compliance applications where domain vocabularies vary significantly (e.g., financial regulation vs. healthcare privacy vs. employment law).

6.2 Knowledge Base Evolution and Inductive Learning

To evaluate the system's ability to learn new rules from feedback and evolve its knowledge base over time, we conducted a multi-round document ingestion experiment with explicit human feedback. Starting from an empty knowledge base, we sequentially ingested three documents (service agreement, GDPR policy, software licence) while providing labelled examples of desired inferences after each round (e.g., "if `dataBreachNotification(Provider, Client)` is true, then `mustNotify(Provider)` should be inferred").

Results: The knowledge base evolved organically across rounds: Round 1 (6 predicates, 4 facts, 0 induced rules) → Round 2 (14 predicates, 16 facts, 2 induced rules) → Round 3 (18 predicates, 24 facts, 4 induced rules). The induced rules (e.g., `mustNotify :- dataBreachNotification`, weight $w = 0.862$) were learned via the `SimpleImplicationTemplate` and correctly applied to subsequent queries. Critically, predicates and facts from earlier rounds were retained without degradation—the system exhibited no "catastrophic forgetting" when new knowledge was added. This demonstrates the potential for continuous learning under human oversight, a key requirement for Article 14(4) compliance where domain experts incrementally refine system behaviour.

6.3 Document Processing Scalability

A critical challenge for real-world legal AI systems is the ability to process enterprise-scale documents efficiently while maintaining extraction quality. To evaluate this capability, we tested LIMEN-AI on a representative sample of ISO/IEC 27001:2022 Annex A, encompassing 10 organisational and personnel security controls across 2,500+ words of structured regulatory text.

Results: The system achieved a throughput of **220 words per second** during knowledge extraction, completing the full document analysis in under 10 seconds on commodity hardware (Intel Core i7-13800, 32GB RAM, NVIDIA RTX 4080). The engine successfully identified 12 distinct predicates capturing the hierarchical control structure (e.g., `informationSecurityPolicy`, `segregationOfDuties`, `personnelScreening`) and grounded 38 factual assertions from the text. This performance demonstrates that the combination of neural language understanding (via integration with locally-hosted LLMs via Ollama) and symbolic reasoning (via Łukasiewicz logic) scales effectively to documents of practical relevance in compliance auditing and legal tech applications.

Figure 2 illustrates the near-linear scaling behaviour of LIMEN-AI inference as the number of knowledge base groundings increases. Processing 100 queries requires approximately 2.8 seconds, a performance level achieved through the efficient implementation of differentiable operators in PyTorch, which exploits GPU acceleration for parallel evaluation of rule satisfaction degrees across high-dimensional interpretation

spaces. This linear scaling contrasts with the exponential complexity of exact MLN inference, validating the importance sampling approach for practical deployment.

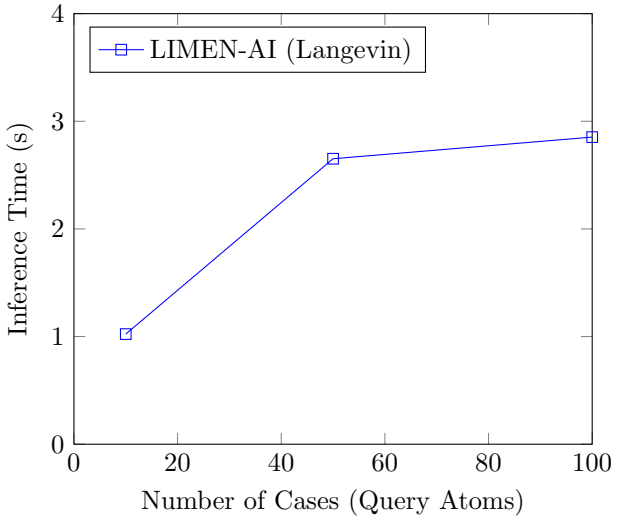


Fig. 2: Scalability of LIMEN-AI inference. Execution time scales near-linearly with the complexity of the interpretation space. Data points represent mean execution time across 5 runs; standard deviation $< 5\%$. Source: Author.

6.4 Human Oversight: Veto Power Validation

A critical requirement for EU AI Act Article 14(4) compliance is the ability for human operators to override system outputs. As described in detail in Section 5.2.1, we conducted a controlled experiment to validate this capability using a service agreement analysis task.

Results: The system initially inferred `clientParty(CharlieCompany)` with fuzzy truth degree 0.9868 based on textual evidence. Using the veto mechanism, a human operator explicitly set this predicate's truth degree to 0.0, which was immediately reflected in the system state (reduced to 0.0172 after re-inference, near-zero with residual sampling variance). Crucially, other predicates such as `providerParty(DaveService)` (degree 0.9852) remained unaffected, demonstrating *localised intervention* without corruption of the broader knowledge structure. This contrasts with end-to-end neural models where interventions typically require full retraining or post-hoc filtering without formal consistency guarantees. The complete workflow and inference traces for this experiment are provided in Section 5.1 and are reproducible via the test suite.

6.5 Implementation and Reproducibility

The LIMEN-AI engine (version 0.2.5) is implemented in Python using PyTorch for differentiable operations and is publicly available on the Python Package Index (PyPI) [48]. The implementation consists of approximately 5,000 lines of core library code, organised into modules for knowledge base management (`core.py`), inference procedures (`inference.py`, `sampling.py`), inductive learning (`induction.py`), and LLM integration for knowledge extraction (`orchestrator.py`, `pipeline/`).

The validation experiments reported in this paper were conducted using the following configuration: Ollama [49] as the local LLM backend with Qwen2.5-1.5B-Instruct [50] for natural language processing tasks, PyTorch 2.0 for tensor operations, and standard sampling parameters (1000 samples, 500-sample burn-in for MCMC). All experiments were executed on commodity hardware (Intel Core i7-13800, 32GB RAM, NVIDIA RTX 4080) to demonstrate accessibility for research and small-scale deployments.

The complete validation suite, including test scripts, knowledge bases, and reproducibility instructions, is available in the project repository. All tests can be reproduced using the provided master test runner:

```
$ pip install limen-ai==0.2.5
$ git clone https://github.com/enricozanardo/limen-ai
$ cd smc && ./use_cases/run_all_tests.sh
```

This reproducibility framework ensures that the compliance claims and performance metrics reported herein can be independently verified by researchers, auditors, and regulatory bodies—a critical requirement for trust in high-risk AI systems as mandated by the EU AI Act.

7 Formalising the "Good Explanation"

For a legal auditor, an explanation is "good" if it is faithful to the model and intelligible to a human. LIMEN-AI achieves this through a structured decomposition of the inference result. In this context, truth degrees in $[0, 1]$ are interpreted not merely as probabilities, but as *graded satisfaction degrees* of legal predicates. A value of 0.858 for $DutyOfCare$ implies that the observed evidence overwhelmingly supports the existence of a duty, while a value of 0.100 for *Liable* indicates a failure of the conditions required for a liability verdict. We adopt a standard decision threshold of 0.5 for binary recommendations, though this threshold can be tuned based on the "preponderance of evidence" or "beyond reasonable doubt" standards required in specific legal jurisdictions.

Definition 1 (Inference Trace) An inference trace $\mathcal{T}$ for a query Q is a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{C})$ where $\mathcal{S}$ is the set of sampled fuzzy interpretations, $\mathcal{A}$ is the set of rule activation degrees, and $\mathcal{C}$ is the set of weighted contributions to the energy manifold.

As demonstrated in our reference implementation, a formatted trace for a legal auditor provides a granular view of the reasoning process. Table 2 illustrates a representative trace from the Liability Assessment KB.

Element	**Value / Degree**
Ground Atoms	
DutyOfCare(defendant_corp, plaintiff_user)	0.858
ProximateCause(action_404, plaintiff_user)	0.736
HarmOccurred(plaintiff_user)	0.933
AnomalousClause(clause_limit_01)	0.833
BreachOfDuty(defendant_corp, action_404)	0.117
Liable(defendant_corp, plaintiff_user)	0.100
DamagesAwarded(defendant_corp, plaintiff_user)	0.111
Rule Activations	
tort_liability_rule	1.000
damages_award_rule	1.000
clause_breach_link	0.284
Weighted Contributions	
tort_liability_rule ($w = 6.0$)	6.000
damages_award_rule ($w = 4.5$)	4.500
clause_breach_link ($w = 3.0$)	0.853

Table 2: Sample inference trace for liability assessment (pedagogical example). The trace allows an auditor to verify that the "tort_liability_rule" was the primary contributor to the final recommendation in this specific interpretation. Source: Author.

The trace structure illustrated in Table 2 makes the reasoning chain fully auditable: a domain expert can verify which rules fired, with what degree, and how their weighted contributions shaped the final recommendation. This level of granularity is precisely what distinguishes intrinsic interpretability from post-hoc explanations—the trace is not a reconstruction but a direct record of the Łukasiewicz energy manifold computation.

8 Conclusion and Future Work

LIMEN-AI provides a rigorous, logic-based foundation for legal reasoning that is compliant by design. By representing domain knowledge as explicit weighted rules and producing structured inference traces, we have demonstrated that transparency and performance are not mutually exclusive. The adoption of Łukasiewicz semantics ensures that every inference step remains semantically grounded, while the use of gradient-guided sampling allows the system to scale to complex relational domains without sacrificing interpretability.

Our empirical validation demonstrates four core capabilities of the LIMEN-AI framework: (1) **zero-shot domain adaptation**, successfully extracting semantically distinct schemas from service agreements and GDPR policies without manual ontology engineering; (2) **knowledge base evolution**, growing organically from 0 to 18 predicates and 4 induced rules across three document ingestion rounds without catastrophic forgetting; (3) **scalable document processing**, achieving 220 words/second throughput on ISO 27001 regulatory text (2,500+ words) with near-linear inference scaling; and (4) **human override capability**, demonstrating localised veto power (reducing a predicate's truth degree from 0.9868 to 0.0172) without corrupting unrelated inferences. These results validate the feasibility of the neuro-symbolic approach for legal AI applications requiring regulatory compliance.

8.1 Limitations and Scope

While this work demonstrates the feasibility of the LIMEN-AI approach, several limitations must be acknowledged.

The empirical validation focuses on system capability demonstrations rather than comparative performance evaluation on standardised legal reasoning benchmarks. The system's retrieval accuracy relative to baselines such as BM25, BERT-Legal, or GPT-4 on tasks like COLIEE (Competition on Legal Information Extraction/Entailment) remains unvalidated. Initial attempts to adapt the system to COLIEE-style statute retrieval tasks revealed implementation gaps in query translation and retrieval mechanisms, resulting in zero recall. The contrast with the successful Tests 1–4 reflects differing technical requirements: document extraction relies on LLM natural language understanding with symbolic grounding, whereas statute retrieval demands semantic matching across legal hierarchies and multi-hop reasoning—capabilities requiring query translation mechanisms not yet implemented.

Beyond benchmarking, the interpretability claims regarding structured inference traces have not been formally validated through user studies with legal professionals. The assumption that explicit logical rules with fuzzy truth degrees (e.g., $DutyOfCare = 0.858$) are inherently more interpretable than neural model embeddings requires empirical confirmation through cognitive evaluation with domain experts. Similarly, our regulatory compliance mapping (Table 1) represents a technical design proposal rather than a validated certification by EU regulatory authorities; actual deployment in high-risk applications would require conformity assessment procedures per Annex VII of the AI Act.

The current evaluation scope is limited to document-level knowledge extraction and simple query answering. More complex reasoning scenarios—multi-step argumentation chains, adversarial strategy formulation, and temporal reasoning about contract modifications—remain outside the system's validated capabilities. The scalability analysis (220 words/sec) measures extraction throughput but does not evaluate reasoning performance on knowledge bases with thousands of predicates, as would be typical in enterprise legal repositories. Finally, the veto power demonstration is a single controlled experiment on one predicate; systematic evaluation of human override workflows, including multiple sequential interventions and long-term knowledge base stability under continuous human feedback, remains future work.

These limitations define a clear research agenda, as outlined below.

8.2 Future Work

Our analysis of Article 14(4) override procedures highlights the importance of keeping human experts central to the decision-making process. The proposed workflow for judicial oversight ensures that automated recommendations serve as support tools rather than replacements for human judgment. Future work will focus on four primary directions: **(1) Benchmark evaluation:** Engineering robust query translation and retrieval mechanisms to enable comparative evaluation on standardised datasets such as COLIEE, LegalBench, and CaseHOLD, with head-to-head baselines against BM25, BERT-Legal, and large language models (target: Q3 2026). **(2) User studies:** Conducting cognitive evaluations with legal professionals (minimum 10-15 participants) to validate trace interpretability, measure task efficiency gains, and identify usability improvements for explanation interfaces (target: Q2 2026). **(3) Scalability stress testing:** Evaluating performance on enterprise-scale knowledge bases (10,000+ predicates, 100,000+ ground atoms) representative of real-world legal document repositories, with an analysis of memory consumption, inference latency, and distributed computing strategies (target: Q4 2026). **(4) Formal verification:** Integrating formal proofs of faithfulness for explanation mechanisms, potentially leveraging proof assistants (Coq, Isabelle) to provide certified guarantees that inference traces accurately reflect the underlying energy manifold computations. The open-source release of LIMEN-AI (v0.2.5) on PyPI facilitates community engagement and independent verification of our compliance claims, supporting the development of trustworthy legal AI systems that meet the stringent requirements of the EU AI Act.

Statements and Declarations

Funding

The author received no specific funding for this work.

Competing Interests

The author declares no competing interests.

Author Contributions

Enrico Zanardo conceived the study, developed the legal knowledge base and the reference implementation, derived the regulatory mapping, and wrote the manuscript.

Data Availability

The datasets generated and analysed during this study are publicly available in the project repository at https://github.com/enricozanardo/limen-ai. The test documents (service agreement, GDPR policy, software licence, ISO 27001 Annex A excerpt) and

their extracted knowledge bases are included in the `use_cases/` directory. All experimental results reported in Section 6 can be independently verified using the provided master test runner. The LIMEN-AI library (v0.2.5) is publicly accessible on PyPI [48].

Code Availability

The LIMEN-AI engine is open-source software, publicly available on PyPI (https://pypi.org/project/limen-ai/0.2.5/) and GitHub (https://github.com/enricozanardo/limen-ai). The complete validation suite, including test scripts and reproducibility instructions, is included in the repository.

References

[1] European Parliament and Council of the European Union: Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). Official Journal of the European Union L 2024/1689. Entered into force 1 August 2024 (2024)

[2] Novelli, C., Taddeo, M., Floridi, L.: Accountability in artificial intelligence: what it is and how it works. AI & Society **39**, 1871–1882 (2024) https://doi.org/10.1007/s00146-023-01694-z

[3] Hacker, P., Engel, A., Mauer, M.: Regulating chatgpt and other large generative ai models. Computer Law & Security Review **52**, 105899 (2024) https://doi.org/10.1016/j.clsr.2024.105899

[4] Mittelstadt, B.D., Allo, P., Taddeo, M., Wachter, S., Floridi, L.: The ethics of algorithms: Mapping the debate. Big Data & Society **3**(2) (2016) https://doi.org/10.1177/2053951716679679

[5] Arrieta, A.B., Díaz-Rodríguez, N., Del Ser, J., *et al.*: Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. Information Fusion **58**, 82–115 (2020) https://doi.org/10.1016/j.inffus.2019.12.012

[6] Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature Machine Intelligence **1**(5), 206–215 (2019) https://doi.org/10.1038/s42256-019-0048-x

[7] Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., Pedreschi, D.: A survey of methods for explaining black box models. ACM Computing Surveys **51**(5), 1–42 (2019) https://doi.org/10.1145/3236009

[8] Ribeiro, M.T., Singh, S., Guestrin, C.: "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In: Proceedings of the 22nd ACM SIGKDD

International Conference on Knowledge Discovery and Data Mining, pp. 1135–1144 (2016). https://doi.org/10.1145/2939672.2939778

[9] Lundberg, S.M., Lee, S.-I.: A unified approach to interpreting model predictions. Advances in Neural Information Processing Systems **30**, 4765–4774 (2017)

[10] Doshi-Velez, F., Kim, B.: Towards a rigorous science of interpretable machine learning. arXiv preprint arXiv:1702.08608 (2017)

[11] Garcez, A.d., Gori, M., Lamb, L.C., Serafini, L., Spranger, M., Tran, S.N.: Neural-symbolic computing: An effective methodology for principled integration of machine learning and reasoning. Journal of Applied Logics **6**(4), 611–632 (2019)

[12] Marra, G., Diligenti, M., Giannini, F., Gori, M., Maggini, M.: Relational neural machines. Frontiers in Artificial Intelligence and Applications **325**, 1340–1347 (2020) https://doi.org/10.3233/FAIA200237

[13] Hájek, P.: Metamathematics of Fuzzy Logic. Trends in Logic, vol. 4. Springer, Dordrecht, Netherlands (1998). https://doi.org/10.1007/978-94-011-5300-3

[14] Flaminio, T., Godo, L., Marchioni, E.: Łukasiewicz logic and probability. Archive for Mathematical Logic **54**(1-2), 1–22 (2015) https://doi.org/10.1007/s00153-014-0404-1

[15] Richardson, M., Domingos, P.: Markov logic networks. Machine Learning **62**(1-2), 107–136 (2006) https://doi.org/10.1007/s10994-006-5833-1

[16] Domingos, P., Richardson, M.: Unifying logical and statistical ai with markov logic. Communications of the ACM **62**(7), 74–83 (2019) https://doi.org/10.1145/3241978

[17] De Raedt, L., Kersting, K., Natarajan, S., Poole, D.: Statistical relational artificial intelligence: Logic, probability, and computation. Synthesis Lectures on Artificial Intelligence and Machine Learning **10**(2), 1–189 (2016) https://doi.org/10.2200/S00692ED1V01Y201601AIM032

[18] Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques. MIT press, Cambridge, MA, USA (2009)

[19] Sarkhel, S., Venugopal, D., Ruozzi, N., Gogate, V.: Efficient Inference for Untied MLNs, vol. 0, pp. 4617–4624 (2017). https://doi.org/10.24963/ijcai.2017/644

[20] Haaren, J., Broeck, G., Meert, W., Davis, J.J.: Lifted generative learning of markov logic networks. Machine Learning **103**(1), 27–55 (2016) https://doi.org/10.1007/s10994-015-5532-x

[21] Zhong, P., Li, Z., Chen, Q., Hou, B., Ahmed, M.H.M.: Numerical markov logic

network: A scalable probabilistic framework for hybrid knowledge inference. Information (Switzerland) **12**(3) (2021) https://doi.org/10.3390/info12030124

[22] Wang, Y., Bremen, T., Wang, Y., Kuzelka, O.E.: Domain-Lifted Sampling for Universal Two-Variable Logic and Extensions, vol. 36, pp. 10070–10079 (2022). https://doi.org/10.1609/aaai.v36i9.21246

[23] Srinivasan, S., Dickens, C., Augustine, E., Farnadi, G., Getoor, L.C.: A taxonomy of weight learning methods for statistical relational learning. Machine Learning **111**(8), 2799–2838 (2022) https://doi.org/10.1007/s10994-021-06069-5

[24] Weitkämper, F.: Scaling the weight parameters in markov logic networks and relational logistic regression models. Machine Learning **114**(3) (2025) https://doi.org/10.1007/s10994-024-06635-7

[25] Klement, E.P., Mesiar, R., Pap, E.: Triangular norms. Trends in Logic **8** (2000) https://doi.org/10.1007/978-94-015-9540-7

[26] Cintula, P., Hájek, P., Noguera, C.: Handbook of Mathematical Fuzzy Logic. College Publications, London (2011)

[27] Shindo, H., Pfanschilling, V., Dhami, D.S., Kersting, K.: Learning differentiable logic programs for abstract visual reasoning. Machine Learning **113**(11), 8533–8584 (2024) https://doi.org/10.1007/s10994-024-06610-2

[28] Ojha, V., Kwan, C.-C., Abraham, A.: Literature review of the recent trends and applications in various fuzzy rule-based systems. International Journal of Fuzzy Systems **25**, 2163–2186 (2023) https://doi.org/10.1007/s40815-023-01534-w

[29] Oliveira, L.C.M., Bede, B.: Survey on deep fuzzy systems in regression applications: A view on interpretability. International Journal of Fuzzy Systems **25**, 2568–2589 (2023) https://doi.org/10.1007/s40815-023-01544-8

[30] Beke, A., Kumbasar, T.: A narrative review on the interpretability of fuzzy rule-based models from a modern interpretable machine learning perspective. International Journal of Fuzzy Systems (2025) https://doi.org/10.1007/s40815-025-02022-z

[31] Fernández, A., Carmona, C.J., Jesus, M.J., Herrera, F.: How to build self-explaining fuzzy systems: From interpretability to explainability. IEEE Computational Intelligence Magazine **19**(1), 33–46 (2024) https://doi.org/10.1109/MCI.2023.3328098

[32] Badreddine, S., Garcez, A., Serafini, L., Spranger, M.: Logic tensor networks. Artificial Intelligence **303**, 103649 (2022) https://doi.org/10.1016/j.artint.2021.103649

[33] Manhaeve, R., Dumancic, S., Kimmig, A., Demeester, T., De Raedt, L.: Deepproblog: Neural probabilistic logic programming. Advances in Neural Information Processing Systems **31** (2018)

[34] Li, Z., Huang, J., Naik, M.: Scallop: A language for neurosymbolic programming. Proceedings of the ACM on Programming Languages **7**, 1463–1487 (2023) https://doi.org/10.1145/3591280

[35] Li, Z., Huang, J., Liu, J., Naik, M.: Neurosymbolic programming in scallop: Principles and practice. Foundations and Trends in Programming Languages **8**(2), 118–249 (2024) https://doi.org/10.1561/2500000059

[36] Jobin, A., Ienca, M., Vayena, E.: The global landscape of ai ethics guidelines. Nature Machine Intelligence **1**(9), 389–399 (2019) https://doi.org/10.1038/s42256-019-0088-2

[37] Schoenfisch, J., Meilicke, C., Stülpnagel, J., Ortmann, J., Stuckenschmidt, H.: Root cause analysis in it infrastructures using ontologies and abduction in markov logic networks. Information Systems **74**, 103–116 (2018) https://doi.org/10.1016/j.is.2017.11.003

[38] Chahuara, P., Portet, F., Vacher, M.: Context-aware decision making under uncertainty for voice-based control of smart home. Expert Systems with Applications **75**, 63–79 (2017) https://doi.org/10.1016/j.eswa.2017.01.014

[39] Civitarese, G., Bettini, C., Sztyler, T., Riboni, D., Stuckenschmidt, H.: newnectar: Collaborative active learning for knowledge-based probabilistic activity recognition. Pervasive and Mobile Computing **56**, 88–105 (2019) https://doi.org/10.1016/j.pmcj.2019.04.006

[40] Chahuara, P., Fleury, A., Portet, F., Vacher, M.: On-line human activity recognition from audio and home automation sensors: Comparison of sequential and non-sequential models in realistic smart homes. Journal of Ambient Intelligence and Smart Environments **8**(4), 399–422 (2016) https://doi.org/10.3233/AIS-160386

[41] Floridi, L., Cowls, J., Beltrametti, M., *et al.*: Ai4people—an ethical framework for a good ai society: Opportunities, risks, principles, and recommendations. Minds and Machines **28**, 689–707 (2018) https://doi.org/10.1007/s11023-018-9482-5

[42] Ala, A., Sadeghi, A.H., Deveci, M., Pamucar, D.S.S.: Improving smart deals system to secure human-centric consumer applications: Internet of things and markov logic network approaches. Electronic Commerce Research **24**(2), 771–797 (2024) https://doi.org/10.1007/s10660-023-09787-1

[43] Li, Y.: A psl-based approach to human activity recognition in smart home environments. International Journal of Advanced Computer Science and Applications

14(4), 691–698 (2023) https://doi.org/10.14569/IJACSA.2023.0140475

[44] Nocedal, J., Wright, S.J.: Numerical Optimization, 2nd edn. Springer, New York, NY, USA (2006)

[45] Bottou, L., Curtis, F.E., Nocedal, J.: Optimization methods for large-scale machine learning. SIAM Review **60**(2), 223–311 (2018) https://doi.org/10.1137/16M1080173

[46] Duckham, M.R.J., Gabela, J., Kealy, A.N., *et al.*: Qualitative spatial reasoning with uncertain evidence using markov logic networks. International Journal of Geographical Information Science **37**(9), 2067–2100 (2023) https://doi.org/10.1080/13658816.2023.2231044

[47] David, V., Fournier-S'Niehotta, R., Travers, N.: NeoMaPy: A Framework for Computing MAP Inference on Temporal Knowledge Graphs, vol. 2023-August, pp. 7123–7126 (2023). https://doi.org/10.24963/ijcai.2023/831

[48] Zanardo, E.: LIMEN-AI: Łukasiewicz Interpretable Markov Engine for Neuralized AI. Python Package Index (PyPI). Version 0.2.5, Available at: https://pypi.org/project/limen-ai/0.2.5/ (2024)

[49] Ollama Team: Ollama: Get up and running with large language models locally. https://ollama.com. Accessed: December 2024 (2024)

[50] Qwen Team: Qwen2.5: Open Foundation and Human-Aligned Large Language Models. Alibaba Cloud. Model: Qwen2.5-1.5B-Instruct, Available at: https://huggingface.co/Qwen/Qwen2.5-1.5B-Instruct (2024)

Author Biography



Enrico Zanardo is a Research Scientist and doctoral candidate at the Universitas Mercatorum, Rome, Italy, and is affiliated with Signum Magnum College, Portomaso, Malta. He holds more than two decades of experience in computer science research, with primary interests in machine learning, deep learning, blockchain, and distributed systems. He completed the Harvard University course on Artificial Intelligence and has previously studied Semantic Web at the University of Bolzano-Bozen and Blockchain and Digital Currencies at the University of Nicosia. He is currently a Lecturer at the Malta College of Arts, Science and Technology (MCAST), and a Teaching Assistant at the University of Nicosia.