⚖ARCCS: An Automated Regulatory Compliance Checking System

Giorgos Filandrianos[1,4], José Menezes[2], Chrysoula Zerva[2,4], Alessandro Gianola[3]
[1]Instituto de Telecomunicações
[2]Instituto de Telecomunicações/IST, Universidade de Lisboa
[3]INESC-ID/IST, Universidade de Lisboa
[4] AILS Lab; ECE, National Technical University of Athens
geofila@islab.ntua.gr,
{jose.sacadura.menezes,chrysoula.zerva,alessandro.gianola}@tecnico.ulisboa.pt

Abstract

Regulatory compliance checking—deciding whether a target document satisfies the obligations of a regulation—requires interpreting dense legal text, identifying which provisions apply, and grounding each decision in explicit evidence. We present ARCCS, an end-to-end, automated, agentic, and regulation-agnostic Legal NLP system for compliance checking. ARCCS decomposes raw regulatory text into atomic, traceable requirements and evaluates a target document against them using retrieved evidence, confidence scores, and human-interpretable justifications. This design decouples compliance assessment from any fixed regulatory template or predefined rule set, enabling the pipeline to operate over regulations of varying size and structure. We evaluate ARCCS in two complementary settings. First, in a GDPR policy-document evaluation, LLM-based judges find its decisions and justifications legally and evidentially consistent in up to 96.67% of the assessed cases. Second, on an EU public-procurement benchmark comprising more than 1,200 individual rule checks, the system attains 98.8% accuracy in violation detection. ARCCS is, to our knowledge, the first fully open-source system for end-to-end regulatory compliance checking and auditable report generation.

1 Introduction

Regulatory compliance checking—deciding whether a target document, such as a vendor proposal or a terms-of-use agreement, satisfies the obligations imposed by a body of regulation—is a high-stakes task that today rests almost entirely on human experts. Doing it well requires three competencies that are individually hard and jointly rare: *interpreting* dense, cross-referential legal text; *identifying* which provisions are actually applicable to a given document; and *grounding* every decision in explicit textual evidence so that it can be audited. This combination makes manual compliance review slow and expensive, and difficult to scale to the document volumes of modern procurement (Gianola and Zerva, 2025).

Legal NLP supplies strong components for these subtasks—domain-adapted encoders and benchmarks for legal language understanding (Chalkidis et al., 2020, 2022), expert-annotated contract review (Hendrycks et al., 2021), and broad legal-reasoning evaluation of large language models (LLMs) (Guha et al., 2023)—and a parallel line of work has shipped *deployed* tools for individual legal artefacts: unfair-clause detection in terms of service (Lippi et al., 2018), privacy-policy analysis (Harkous et al., 2018), and building-code compliance (Hettiarachchi et al., 2025). These systems, however, are each tied to a single artefact type or a single regulation, and do not offer an end-to-end path from a raw, previously unseen regulation to an auditable, requirement-level decision. Applying LLMs directly to close this gap is unsafe: models hallucinate legal content at high rates (Dahl et al., 2024), and even retrieval-augmented commercial legal tools remain unreliable (Magesh et al., 2024). A trustworthy system must therefore not only predict but also expose its evidence and *abstain* when support is insufficient—a behaviour studied as answering only when warranted (Rajpurkar et al., 2018), abstention in LLMs (Wen et al., 2025), and learning to defer to a human expert (Madras et al., 2018; Mozannar and Sontag, 2020).

We present ARCCS[1], an interactive system for end-to-end regulatory compliance checking. Given a regulation and a target document, ARCCS decomposes the regulation into atomic requirements, retrieves relevant evidence from the target document, and produces requirement-level compliance decisions with supporting excerpts and explanations. The system is designed to be regulation- and

[1]https://geofila.github.io/ARCCS/

Proceedings of the 2026 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 109–119
October 24-29, 2026 ©2026 Association for Computational Linguistics

document-agnostic: it is not tied to a fixed checklist, a single legal domain, or a specific artefact type, but provides a reusable pipeline for turning complex regulatory text into traceable compliance assessments.

This paper makes three contributions:

- a general agentic compliance-checking algorithm that converts unseen regulations into atomic, traceable requirements and evaluates target documents using grounded evidence, abstention, and human deferral;
- an evaluation in two complementary settings: GDPR policy-document assessment with multiple LLM judges and an EU public-procurement benchmark of over 1,200 rule-level checks with deterministic ground truth; and
- to our knowledge, the first free and fully open-source implementation of an end-to-end regulatory compliance workflow, provided through an easy-to-use application that requires no programming or pipeline configuration and generates structured, auditable reports.

2 System Architecture

This section describes what ARCCS computes and how. At a high level, ARCCS takes two inputs—a regulatory document D and a target document T (a proposal or terms-of-use agreement)—and returns a structured report that, for each obligation in D, states whether T satisfies it, with what confidence, and on the basis of which textual evidence. As illustrated in Figure 1, computation proceeds in two stages. The *Regulatory Processing and Extraction Module* (RPEM, §2.1) converts the regulation into a set of machine-checkable requirements; the *Compliance Classification Module* (CCM, §2.2) checks the target document against each requirement. We formalise the two modules below and then give precise definitions of the four compliance labels (§2.3).

2.1 Regulatory Processing and Extraction Module

Hierarchical segmentation. RPEM first segments the regulation D into an ordered set of chunks $\mathcal{C} = \{c_1, \dots, c_m\}$ that respects its legal structure (articles, paragraphs, sub-paragraphs). Each chunk c_j retains its structural metadata (e.g. article identifier and position), so that local legal context is preserved for fine-grained analysis, following structured legal-document modelling (Chalkidis et al., 2020).

Atomic requirement extraction. RPEM then applies an extraction mapping

$$\phi : \mathcal{C} \to \mathcal{R}, \qquad \mathcal{R} = \{r_1, \dots, r_n\}, \tag{1}$$

that converts chunks into *atomic* requirement objects. Each requirement is represented as a tuple

$$r_i = \big(f_i,\ \alpha_i,\ \sigma_i,\ M_i,\ \mathrm{src}_i\big), \tag{2}$$

where f_i is the legal function of the provision (obligation, prohibition, right, or condition), α_i the regulated actor, σ_i the jurisdictional and domain scope, $M_i = \{\mu_{i1}, \mu_{i2}, \dots\}$ the set of mandatory conditions that must hold for the requirement to be satisfied, and src_i an explicit pointer to the originating regulatory text. The mandatory-condition set M_i is what renders a requirement *checkable*, and src_i makes every subsequent decision traceable to a specific provision —a deliberate response to the evidence and auditability gaps behind legal hallucination (Dahl et al., 2024; Magesh et al., 2024).

Filtering and consolidation. Not all chunks yield enforceable constraints. RPEM removes provisions that are purely definitional, descriptive, or contextual, and then deduplicates requirements that are semantically equivalent but restated across cross-references, merging them while preserving all source pointers. The result is a compact, non-redundant requirement set $\mathcal{R}$ that serves as the normative knowledge layer for the CCM.

2.2 Compliance Classification Module

The CCM segments the target document into passages $\mathcal{P} = \{p_1, \dots, p_k\}$ and evaluates each requirement $r_i \in \mathcal{R}$ in three steps.

(1) Applicability. An applicability predicate

$$a(r_i \mid T) \in \{0, 1\} \tag{3}$$

decides, from the scope σ_i and the properties of T (e.g. system type, domain, deployment jurisdiction), whether r_i governs T at all. Requirements with $a(r_i \mid T) = 0$ are excluded from compliance reasoning, reducing analytical noise and keeping the report focused on relevant obligations.

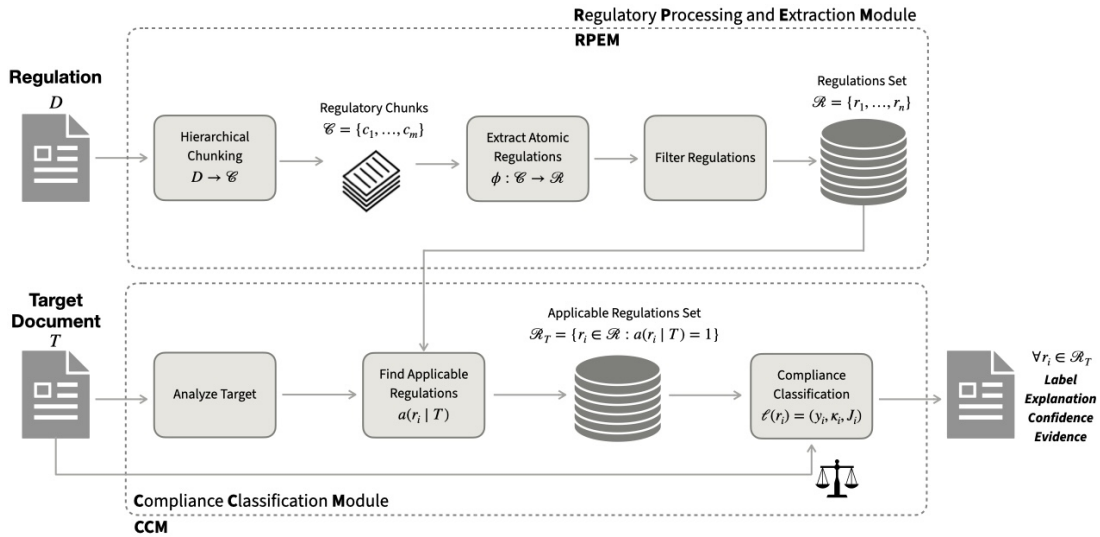

Figure 1: Overview of the proposed regulatory compliance assessment pipeline.

(2) Evidence retrieval. For each applicable requirement, the CCM scores the relevance of every passage with a function

$$\rho(r_i, p_j) \in [0, 1], \tag{4}$$

and selects the supporting evidence set

$$E_i = \{\, p_j \in \mathcal{P} : \rho(r_i, p_j) \geq \tau \,\}, \tag{5}$$

where τ is a relevance threshold. The compliance decision for r_i is then conditioned on E_i rather than on the model's parametric memory—a retrieval-augmented design (Lewis et al., 2020) that grounds judgments in the document text and is known to curb, though not eliminate, hallucinations (Magesh et al., 2024).

(3) Labelling. Finally, a decision function

$$\ell(r_i) = \big(y_i,\ \kappa_i,\ J_i\big) \tag{6}$$

assigns a compliance label $y_i \in \mathcal{Y}$ (§2.3), a confidence score $\kappa_i \in [0, 1]$ derived from the consistency and completeness of the evidence in E_i with respect to the mandatory conditions M_i, and a natural-language justification J_i that cites the passages in E_i and the provision src_i.

2.3 Compliance Labels

ARCCS does not force a verdict on every requirement. Instead, the label space is $\mathcal{Y} = \{\text{C}, \text{NC}, \text{II}, \text{HR}\}$, defined as follows. Let θ_{hi} and θ_{lo} be confidence thresholds with $\theta_{\text{lo}} < \theta_{\text{hi}}$.

COMPLIANT (C). Every mandatory condition $\mu \in M_i$ is explicitly supported by the evidence E_i, no passage in E_i contradicts r_i, and $\kappa_i \geq \theta_{\text{hi}}$. Formally, $\big(\forall \mu \in M_i : \text{sat}(\mu, E_i)\big) \wedge \neg\text{conflict}(r_i, E_i) \wedge \kappa_i \geq \theta_{\text{hi}}$.

NON-COMPLIANT (NC). The evidence E_i either directly contradicts a mandatory condition or omits a required condition while addressing its topic, with high confidence: $\big(\exists \mu \in M_i : \text{violate}(\mu, E_i)\big) \wedge \kappa_i \geq \theta_{\text{hi}}$.

INSUFFICIENT INFORMATION (II). The evidence references the regulatory topic but does not determine satisfaction of the mandatory conditions; that is, some $\mu \in M_i$ is neither confirmed nor refuted by E_i, yielding low-to-moderate confidence ($\kappa_i < \theta_{\text{hi}}$). This is principled abstention: the system answers only when the evidence warrants it (Rajpurkar et al., 2018; Wen et al., 2025).

HUMAN REQUIRED (HR). A reliable automated decision cannot be made due to ambiguity, mutually contradictory evidence, missing external legal/technical context, or confidence below the deferral threshold ($\kappa_i < \theta_{\text{lo}}$). The case is thus routed to a human expert (Madras et al., 2018; Mozannar and Sontag, 2020).

The distinction between II and HR concerns the source of uncertainty: II indicates missing documentary evidence, whereas HR indicates that relevant evidence exists but requires expert interpretation because it is ambiguous, conflicting, or dependent on external context. An illustrative example follows.

II: The regulation requires an internal record of processing activities, but the public terms provide no information about whether such a record exists.
HR: The terms contain conflicting retention clauses, or the decision depends on whether a legal exemption applies based on facts outside the document.

Compliance report. The output of ARCCS is a regulation-level report listing, for every applicable requirement, the tuple $(y_i, \kappa_i, J_i, \text{src}_i)$. Since each entry pairs a decision with its confidence, its supporting evidence, and the governing provision, the report is auditable end-to-end and directly supports human-in-the-loop review in real compliance workflows.

3 Experimental Setup

We evaluate ARCCS in two complementary settings. The GDPR setting tests evidence-grounded assessment of policy documents without deterministic gold labels, while the procurement setting tests rule-level violation detection against independently implemented deterministic checks.

3.1 Setting 1: GDPR Policy-Document Compliance

Inputs. We use the GDPR as regulatory input and evaluate the terms of use of three widely used applications: WhatsApp, Netflix, and ChatGPT. The documents vary substantially in length and specificity, allowing us to examine how available textual evidence affects compliance labels.

Evaluation protocol. We report the predicted-label distribution over the 100 GDPR-derived requirements for each document. Since no deterministic gold standard is available, we additionally assess decision quality using an LLM-as-a-judge protocol (Gu et al., 2026; Zheng et al., 2023). For each sampled decision, the judge receives the regulation text, the extracted requirement representation, the relevant terms-of-use evidence, and the full ARCCS output, including the predicted label and explanation. We sample 30 decisions per document, yielding 90 judged instances, and use three independent evaluator models to assess consistency with the provided regulation and evidence as well as inter-evaluator agreement.

3.2 Setting 2: Public-Procurement Event-Log Compliance

Legal scope and benchmark. We evaluate public-procurement compliance under Directive 2014/24/EU (CELEX: 32014L0024), focusing on twelve rules derived from Articles 4, 48, 56, and 73 rather than the full Directive. The complete rule list and the operational mapping from legal provisions to executable checks are included with the released system.

Data and ground truth. We construct a controlled benchmark of 100 procurement traces following the Tenders Electronic Daily (TED) event-log schema.[2] Because several relevant violations are sparse in naturally occurring data, the benchmark includes 60 traces with at least one targeted violation and 40 fully compliant controls. Expected labels are derived independently through deterministic logic over each trace and its attributes, ensuring that ARCCS is evaluated against externally defined legal-process ground truth rather than model-generated judgments. This setting is therefore intended as a controlled sanity check of rule-level compliance reasoning, not as evidence of full real-world procurement coverage.

Metrics. Each of the 100 traces is evaluated against all twelve rules, yielding 1,200 rule-level decisions. We report rule-level accuracy, precision, recall, and F_1, treating true legal violations as the positive class, together with case-level performance obtained by aggregating rule outputs into a binary compliant/non-compliant trace label. We also report exact match, defined as the proportion of cases for which all twelve rule decisions are correct. The benchmark is evaluated with `gpt-5.4`, `gpt-5-mini`, and `gpt-5.2` under the same legal scope and protocol.

3.3 Models

ARCCS is model- and backend-agnostic: RPEM and CCM use structured prompts and typed outputs, without model-specific fine-tuning or local GPU requirements. In our experiments, we use OpenAI API models from the GPT-5 family, with `gpt-5.2` as the default backbone unless otherwise stated. All main runs use medium reasoning effort, and the confidence threshold for the *Human Required* label is fixed at 0.60. Exact model identifiers and model cards are reported in App. D.

4 Evaluation

4.1 Policy-Document Compliance

Decisions are evidence-grounded. Table 1 reports the accuracy of the three evaluator models on the sampled GDPR set. The judges find ARCCS

[2] https://ted.europa.eu

Evaluator (LLM-as-a-Judge)	Accuracy (%)
gpt-5.1	96.67
gpt-5.2	90.00
gpt-5.2-pro	90.00

Table 1: Accuracy of model-based evaluators on the sampled set (30 checks per terms-of-use doc.; 90 total).

Agreement	κ	p-value
Cohen's κ (5.1 vs 5.2)	0.47	0.19
Cohen's κ (5.1 vs 5.2-pro)	0.47	0.19
Cohen's κ (5.2 vs 5.2-pro)	1.00	< 0.001
Fleiss' κ (3 raters)	0.69	< 0.001

Table 2: Inter-evaluator agreement on the binary verdict (YES/NO) for the sampled set.

decisions consistent with the provided regulation and evidence in 90.00–96.67% of cases. Because the judge conditions its verdict jointly on the regulation, the retrieved evidence, and the ARCCS explanation, these scores indicate that the system produces evidence-grounded rationales rather than unsupported label assignments. Table 2 reports inter-evaluator agreement: we observe substantial overall agreement (Fleiss' $\kappa = 0.69$, $p < 0.001$), with gpt-5.2 and gpt-5.2-pro in perfect agreement ($\kappa = 1.00$, $p < 0.001$), indicating that the favourable verdicts are stable rather than an artefact of a single judge.

Abstention is evidence-faithful, not evasive. Across the GDPR-derived requirements, *Insufficient Information* is the dominant outcome, as expected for public terms-of-use documents. Most GDPR obligations depend on internal governance artefacts, such as records of processing, DPIAs, and security measures, which are not normally disclosed in public-facing terms. A faithful compliance system should therefore abstain when the document does not provide enough evidence, rather than infer internal practices from silence. Importantly, abstention does not prevent the system from issuing determinate verdicts when the retrieved evidence supports them: ARCCS reserves *Compliant*/*Non-Compliant* labels for evidence-backed cases and routes borderline cases to *Human Required*.

Surfacing confirmed conflicts. Beyond missing information, ARCCS surfaces *candidate* normative conflicts for human review—system-generated flags, not legal determinations. Some of these flags correspond to conflicts that have independently been confirmed by regulators. For example, ARCCS flags WhatsApp terms as problematic, and the Irish Data Protection Commission later found WhatsApp non-compliant and imposed a 225M fine under Arts. 5(1)(a) and 12–14 GDPR.[3] Similarly, for the *ChatGPT Terms of Use*, ARCCS flags GDPR Article 79 as *Non-Compliant*, because mandatory-arbitration and exclusive San Francisco forum clauses conflict with the Article 79 guarantee that EU/EEA data subjects may sue in their Member State of residence. These cases show that the system identifies concrete evidence-backed conflicts, not only gaps, while still leaving final legal interpretation to experts.

Model	Acc.	Prec.	Rec.	F_1
gpt-5.4	98.5	83.2	98.8	90.3
gpt-5.2	98.8	85.9	100.0	92.4
gpt-5-mini	98.8	85.9	100.0	92.4

Table 3: Rule-level performance on the procurement benchmark (100 traces, 1,200 rule decisions per model).

4.2 Public-Procurement Logs

Rule-level performance. At rule level (Table 3) all three models perform strongly and remain tightly clustered: accuracy ranges from 98.5% to 98.8% and recall is near-saturated (98.8–100.0%), indicating robust detection of true violations. gpt-5.2 and gpt-5-mini jointly provide the strongest profile (precision 85.9, recall 100.0, F_1 92.4), outperforming gpt-5.4 chiefly through fewer false alarms. Crucially, the residual error is false-positive-driven rather than miss-driven: since recall is near 100%, ARCCS follows the conservative error profile that compliance screening demands.

Case-level performance. Aggregated to a per-trace violation/no-violation decision (Table 4), performance is strong: gpt-5.2 and gpt-5-mini reach 100.0% on all case-level metrics, while gpt-5.4 attains 98.0% accuracy and 98.4% F_1, correctly flagging all violating traces. As a first-stage screening mechanism that aims to separate problematic from compliant cases ARCCS is therefore highly reliable.

Exact-match performance. Under the strictest criterion—a trace counts as correct only if all twelve rule decisions are simultaneously correct—scores are necessarily lower (Table 5), since a single rule-level slip invalidates the trace. gpt-5-mini is best (87.0%) and also exhibits

[3] Irish Data Protection Commission WhatsApp decision.

Model	Acc.	Prec.	Rec.	F_1
gpt-5.4	98.0	96.8	100.0	98.4
gpt-5.2	100.0	100.0	100.0	100.0
gpt-5-mini	100.0	100.0	100.0	100.0

Table 4: Case-level (per-trace) performance on the procurement benchmark.

Model	Exact Match (%)
gpt-5.4	83.0
gpt-5.2	86.0
gpt-5-mini	87.0

Table 5: Document-level exact-match performance: the percentage of traces for which all 12 rule decisions are simultaneously correct.

the lowest *Insufficient Information* rate (2.3%, vs 7.3% for gpt-5.2 and 8.8% for gpt-5.4), indicating slightly better fine-grained calibration when all rules must hold at once. The gap between near-perfect case-level scores and exact-match scores stems from isolated extra false alarms on individual rules rather than broad misunderstanding of entire traces, and should be read as a fine-grained calibration challenge rather than a failure of core violation detection.

5 Demonstration

ARCCS is delivered as a self-contained web application that runs locally and requires no programming, model configuration, or pipeline orchestration from the user: after supplying their own API key, the user starts the server and reaches the interface in a browser, where the full ARCCS pipeline is driven through a three-step wizard. *Step 1*, the user uploads a regulatory document or selects one of the preloaded regulations (e.g., the GDPR), which skips extraction and reuses a cached, pre-extracted requirement set. *Step 2*, the user uploads the target policy or terms-of-use document via drag-and-drop. *Step 3*, the user launches the analysis and the system checks the document against the extracted requirements. Throughout extraction and checking, the interface streams real-time logs that expose pipeline progress and intermediate stages, making the reasoning process transparent to non-technical stakeholders.

The user inspects the outcome through a structured compliance report. A summary panel reports the counts of the four labels together with an aggregate compliance-rate indicator; below it, a results table lists one row per requirement. Selecting any row opens a detail view exposing the governing provision and its identifier, the assigned label, the confidence score, the verbatim evidence quoted from the target document, and the natural-language justification. This layout makes the outputs directly interpretable: a non-expert can read a verdict, see the exact clause that supports it, and immediately recognise the cases the system has deferred—those labelled *Human Required* (confidence below the 0.60 threshold) or *Insufficient Information*—as the ones warranting manual review. Because each decision is linked to its source provision and evidence, the report is auditable clause by clause. The user can export the complete report as a structured JSON file, and the system persists a browsable history of prior runs, enabling iterative auditing and comparison across documents and over time. The application itself runs locally: documents are never uploaded to a third-party web service, and the only external call is to the language-model provider chosen by the user, which can be a self-hosted open-weight model for fully air-gapped deployments. A step-by-step visual walkthrough, worked example outputs, and implementation details are provided in App. A, B, and C. Additionally, the released system includes the full rule definitions, implementation details, prompts, example inputs and outputs, and materials needed to reproduce the demo.

6 Conclusion

We presented ARCCS, an end-to-end interactive system for regulatory compliance checking that turns a regulation and a target document into an auditable compliance report. By decomposing the regulation into atomic, traceable requirements, ARCCS can analyse long legal and policy documents requirement by requirement, grounding each decision in retrieved evidence, confidence scores, and explanations rather than relying on a fixed checklist or hand-crafted rules. Our evaluation demonstrates strong performance in both policy-document compliance and deterministic public-procurement checking, showing that the same architecture supports evidence-grounded assessment and accurate rule-level violation detection. Together with the openly available, on-premise demo, this positions ARCCS as a transparent tool for human-in-the-loop regulatory review.

Limitations

Several limitations qualify our findings and outline directions for future work.

Absence of a human gold standard in the GDPR setting. For the terms-of-use experiments no deterministic ground truth exists, so we rely on an LLM-as-a-judge protocol. While inter-judge agreement is substantial overall (Fleiss' $\kappa = 0.69$), agreement between some evaluator pairs is only moderate and not statistically significant (Cohen's $\kappa = 0.47$, $p = 0.19$). Moreover, all evaluators belong to the GPT-5 model family, and GPT-5.2 is also used as the default ARCCS backbone. This limits the independence of the evaluation. A rigorous human evaluation by qualified legal experts is needed to validate these results and is left to future work.

Controlled data and limited real-world validation. Because violations are sparse in real procurement logs, the procurement benchmark consists of controlled traces built on the TED schema with deliberately balanced violations. This design enables reproducible, deterministic evaluation, but performance on naturally occurring, noisier logs may differ, and the externally defined ground truth reflects our operational formalisation of the Directive rather than adjudicated legal outcomes.

Component-level evaluation. We therefore do not include dedicated comparisons against simpler direct-prompting or vanilla RAG configurations, nor systematic ablations of the individual stages of the pipeline. In particular, the fidelity of the requirements produced by RPEM is not independently evaluated. Errors introduced during requirement extraction may propagate to downstream compliance decisions even when the subsequent CCM reasoning is internally consistent.

Limited scale and scope of evaluation. The judged GDPR sample comprises 90 instances, and the quantitative benchmark covers 100 traces under a single Directive. Our study also spans only two regulatory regimes (the GDPR and Directive 2014/24/EU) in English; generalisation to other legal domains, jurisdictions, and languages remains to be demonstrated.

Model coverage and configuration. Although ARCCS is designed to be model-agnostic, the reported experiments use only the GPT-5 family through a commercial API. We have not yet quantified performance with open-weight, fully on-premise backbones. Likewise, the decision thresholds (the *Human Required* confidence threshold of 0.60 and the retrieval threshold τ) were fixed rather than systematically calibrated. Finally, although violation recall is near-perfect to perfect in the controlled procurement benchmark, this evaluation does not establish the prevalence of the more consequential false-Compliant failure mode—that is, cases in which a true violation is not surfaced as *Non-Compliant*—in naturally occurring regulatory settings.

Scope of intended use. ARCCS is a decision-support tool, not a substitute for professional legal judgment. Its outputs are intended to assist expert review through transparent, evidence-linked reports, and consequential compliance determinations should remain subject to human oversight.

Ethical Considerations

ARCCS is a decision-support tool and is not a substitute for professional legal judgment; it must not be used to make final, consequential compliance decisions without expert oversight. Several risks follow directly from the task. (i) **False negatives**—a missed violation—could create a false sense of safety; we therefore tune the system conservatively (recall at or near 1.0 in our procurement experiments) and surface confidence and evidence so users can prioritise review. (ii) **False positives and hallucinated legal interpretations** could mislead non-expert users; to mitigate this, every output is grounded in retrieved source text, every NC flag is presented as a candidate conflict for human verification, and the *Insufficient Information* and *Human Required* labels make abstention explicit rather than forcing a verdict. (iii) **Over-reliance and accountability**: the interface presents verdicts as reviewable flags with citations, and we document that legal responsibility remains with the human user. (iv) **Privacy**: the application runs locally and processes documents on the user's own machine; when a hosted model API is used, only the user's chosen provider receives the text, and a self-hosted open-weight model can be used for fully air-gapped operation. We make the code and data public to support scrutiny and reproducibility, and we caution that deployment in a new jurisdiction or legal domain requires validation by qualified legal experts.

Acknowledgments

This work was partially supported by the 'OptiGov' project, with ref. n. 2024.07385.IACDC (DOI:

10.54499/2024.07385.IACDC), fully funded by the 'Plano de Recuperação e Resiliência' (PRR) under the investment 'RE-C05-i08 - Ciência Mais Digital' (measure 'RE-C05-i08.m04'), framed within the financing agreement signed between the 'Estrutura de Missão Recuperar Portugal' (EMRP) and Fundação para a Ciência e a Tecnologia, I.P. (FCT) as an intermediary beneficiary. A. Gianola was also partly supported by Portuguese national funds through Fundação para a Ciência e a Tecnologia, I.P. (FCT) under projects UID/50021/2025 (DOI: https://doi.org/10.54499/UID/50021/2025) and UID/PRR/50021/2025 (DOI: https://doi.org/10.54499/UID/PRR/50021/2025). G. Filandrianos and C. Zerva were also partly supported within the framework of the Pharos AI Factory project, funded by the European High-Performance Computing Joint Undertaking (EuroHPC JU) under Grant Agreement No. 101234269 as part of the Horizon Europe and by the Greek Public Investments Program programme.

References

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: The muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904, Online. Association for Computational Linguistics.

Ilias Chalkidis, Abhik Jana, Dirk Hartung, Michael Bommarito, Ion Androutsopoulos, Daniel Katz, and Nikolaos Aletras. 2022. LexGLUE: A benchmark dataset for legal language understanding in English. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4310–4330, Dublin, Ireland. Association for Computational Linguistics.

Matthew Dahl, Varun Magesh, Mirac Suzgun, and Daniel E Ho. 2024. Large legal fictions: Profiling legal hallucinations in large language models. *Journal of Legal Analysis*, 16(1):64–93.

Alessandro Gianola and Chrysoula Zerva. 2025. Compliance checking for public administration processes using retrieval-augmented generation in llms: Novel directions and challenges. In *Short Paper Proceedings of the 7th International Workshop on Artificial Intelligence and Formal Verification, Logic, Automata, and Synthesis, OVERLAY 2025, Bologna, Italy, October 26, 2025*, volume 4142 of *CEUR Workshop Proceedings*, pages 199–208. CEUR-WS.org.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, and 1 others. 2026. A survey on llm-as-a-judge. *The Innovation*, 7(6).

Neel Guha, Julian Nyarko, Daniel E. Ho, Christopher Ré, Adam Chilton, Aditya Narayana, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel N. Rockmore, Diego Zambrano, Dmitry Talisman, Enam Hoque, Faiz Surani, Frank Fagan, Galit Sarfaty, Gregory M. Dickinson, Haggai Porat, Jason Hegland, and 21 others. 2023. Legalbench: a collaboratively built benchmark for measuring legal reasoning in large language models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Hamza Harkous, Kassem Fawaz, Rémi Lebret, Florian Schaub, Kang G Shin, and Karl Aberer. 2018. Polisis: Automated analysis and presentation of privacy policies using deep learning. In *27th USENIX Security Symposium (USENIX Security 18)*, pages 531–548.

Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. 2021. Cuad: An expert-annotated nlp dataset for legal contract review. *NeurIPS*.

Hansi Hettiarachchi, Amna Dridi, Mohamed Medhat Gaber, Pouyan Parsafard, Nicoleta Bocaneala, Katja Breitenfelder, Gonçal Costa, Maria Hedblom, Mihaela Juganaru-Mathieu, Thamer Mecharnia, and 1 others. 2025. Code-accord: A corpus of building regulatory data for rule generation towards automatic compliance checking. *Scientific data*, 12(1):170.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Marco Lippi, Przemyslaw Palka, Giuseppe Contissa, Francesca Lagioia, Hans-Wolfgang Micklitz, Giovanni Sartor, and Paolo Torroni. 2018. Claudette: an automated detector of potentially unfair clauses in online terms of service. *arXiv preprint arXiv:1805.01217*.

David Madras, Toni Pitassi, and Richard Zemel. 2018. Predict responsibly: improving fairness and accuracy by learning to defer. *Advances in neural information processing systems*, 31.

V Magesh, F Surani, M Dahl, M Suzgun, CD Manning, and DE Ho. 2024. Hallucination-free? assessing the reliability of leading ai legal research tools. arxiv.

Hussein Mozannar and David Sontag. 2020. Consistent estimators for learning to defer to an expert. In *International conference on machine learning*, pages 7076–7087. PMLR.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th Annual*

Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 784–789.

Bingbing Wen, Jihan Yao, Shangbin Feng, Chenjun Xu, Yulia Tsvetkov, Bill Howe, and Lucy Lu Wang. 2025. Know your limits: A survey of abstention in large language models. *Transactions of the Association for Computational Linguistics*, 13:529–556.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

A Demonstration Walkthrough

This appendix provides a visual walkthrough of the ARCCS web interface (Figure 2), which is designed for users without programming experience. The interface abstracts away model configuration and pipeline orchestration, exposing the end-to-end compliance workflow through three sequential steps (Figure 2).

Step 1 – Upload the regulation. The user uploads a regulatory document or selects one of the preloaded regulations, such as the GDPR. Internally, this input is passed to the RPEM, which performs hierarchical segmentation and atomic requirement extraction.

Step 2 – Upload the target document. The user uploads the policy or terms-of-use document to be assessed. This document is segmented into passages that serve as the evidence pool for the retrieval-augmented compliance reasoning performed by the CCM.

Step 3 – Inspect the compliance report. After execution, the interface renders a structured compliance report. A summary panel reports the counts of the four compliance labels, followed by a per-requirement breakdown in which each entry shows the governing provision, the assigned label, the confidence score, the retrieved evidence, and a natural-language justification. During execution, the interface additionally streams real-time logs that expose pipeline progress, and it preserves a history of prior runs to support iterative auditing and cross-document comparison.

B Qualitative Example

This appendix gives representative ARCCS outputs for two labels, using GDPR checks against WhatsApp terms of use. Detailed examples for all labels and all evaluation inputs used in the paper are included in the released demo.

Example 1 – Compliant (GDPR Article 2)

Compliance Status: COMPLIANT

Confidence: 0.78
Regulation: GDPR Article 2 – Material scope
Target: WhatsApp Terms of Service

Evidence from Document.

> "WhatsApp's Privacy Policy describes our data (including message) practices. . . and your rights in relation to the processing of information about you."

Explanation. GDPR Article 2 defines the material scope of the Regulation. The document is a Terms of Service that references a separate Privacy Policy for data-processing practices; it neither claims that the GDPR is inapplicable (e.g., as purely personal/household activity or competent-authority processing) nor states that it applies where it should not. No direct conflict with the material-scope provision is present.

Example 2 – Non-Compliant (GDPR Article 5(1)(d))

Compliance Status: NON-COMPLIANT
Confidence: 0.62
Regulation: GDPR Article 5(1)(d) – accuracy principle
Target: WhatsApp Terms of Service

Contradiction Details. Article 5(1)(d) requires personal data to be accurate and kept up to date, and obliges controllers to erase or rectify inaccurate data without delay. The document includes a broad disclaimer of accuracy, which conflicts with this principle.
Evidence from Document.

> "We do not warrant that any information provided by us is accurate, complete, or useful. . . "

Explanation. The accuracy principle is a mandatory obligation on controllers. The Terms explicitly disclaim any commitment to accuracy, and no other clause re-establishes a commitment to maintain or rectify accuracy; this directly conflicts with the regulatory requirement.

C Implementation Details

ARCCS is implemented in Python as two modules, RPEM and CCM, exposed through both a lightweight web application and a research notebook. The RPEM parses regulatory PDFs with the `unstructured` library (high-resolution strategy, backed by `poppler`), converts the parsed elements to markdown, and segments the text into sections by headers, articles, and chapters. Each section is passed to the language model, which returns *typed, schema-constrained* regulatory objects; a quality-scoring stage then filters out incomplete or non-actionable items below a configurable threshold. The CCM performs contradiction-based compliance reasoning: for each requirement it retrieves the relevant document evidence, conditions the decision on that evidence, and emits a label, a confidence score in $[0, 1]$, the supporting evidence quotes, and a justification. The four-way decision follows a fixed priority—a detected contradiction yields *Non-Compliant*; absence of rel-

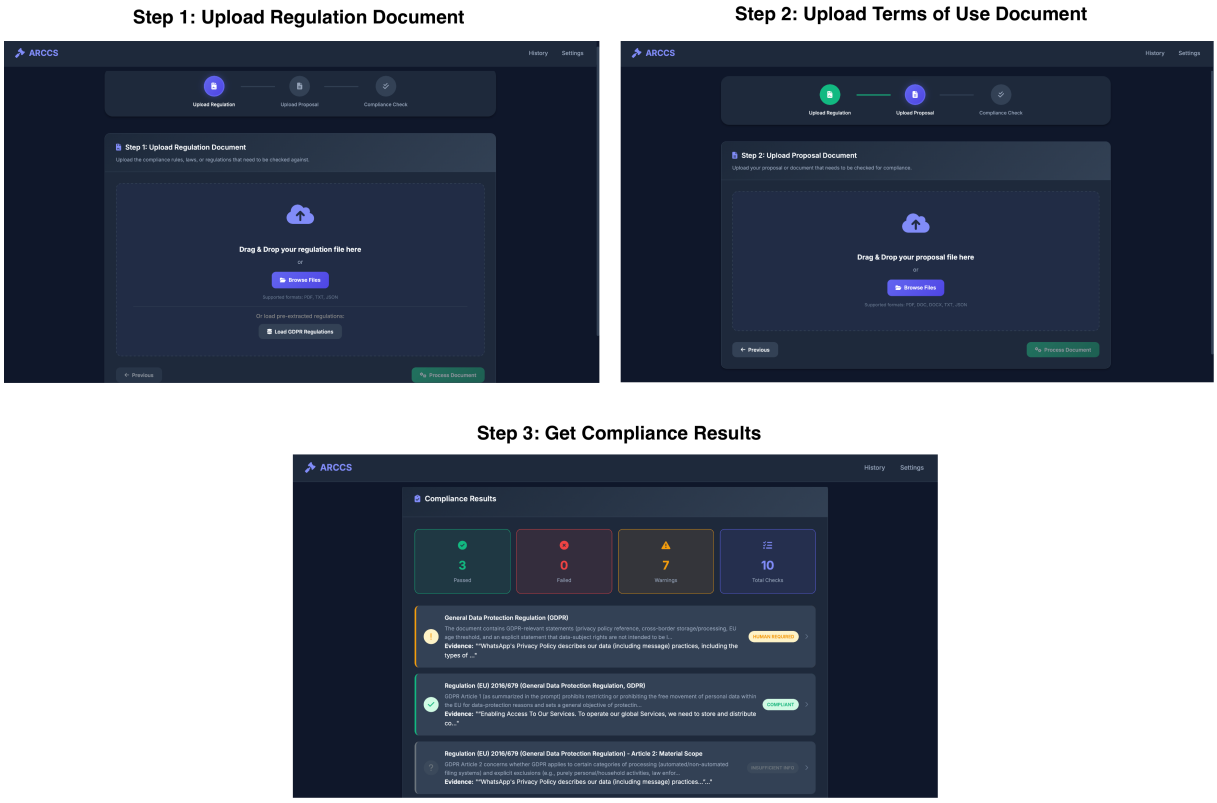

Figure 2: The ARCCS web demonstration interface, designed for users without programming experience. The workflow takes a regulation and a target document as input and returns an auditable, evidence-linked compliance report.

evant information yields *Insufficient Information*; a confidence below 0.60 yields *Human Required*; otherwise *Compliant*.

The web layer is a Flask application with a browser front-end (a dark-themed three-step wizard with drag-and-drop upload, a results table, and a per-row detail view). Pipeline progress is streamed to the client in real time via Server-Sent Events. Reports are stored as structured JSON and can be exported by the user; in addition, the system maintains a per-run history (timestamped snapshots plus a `latest` pointer per case) that supports auditing and cross-run comparison. The application runs entirely on the user's own machine (`python app.py`, served locally); the user supplies their own model API key, which is read from the local environment and never bundled with the code, so that confidential regulatory and proposal documents remain on-premise. End-to-end latency is dominated by language-model calls and is reduced in practice by caching pre-extracted regulation sets (e.g., the preloaded GDPR), which makes repeated checks against the same regulation near-instant on the extraction side.

D Model Cards

For full reproducibility, Table 6 lists the exact model snapshots used in our experiments, together with the role each model plays in the study. Unless otherwise stated, the ARCCS backbone is `gpt-5.2` with reasoning effort set to *medium*, and the confidence threshold for emitting the *Human Required* label is fixed at 0.60. All models are accessed through the OpenAI API; no model-specific fine-tuning is performed.

Model	Snapshot
`gpt-5.2`	`gpt-5.2-2025-12-11`
`gpt-5.1`	`gpt-5.1-2025-11-13`
`gpt-5.2-pro`	`gpt-5.2-pro-2025-12-11`
`gpt-5.4`	`gpt-5.4-2026-03-05`
`gpt-5-mini`	`gpt-5-mini-2025-08-07`

Table 6: Exact model snapshots used in the experiments.