

Benchmarking Large Language Models as Automated Coders of AI Governance Regulatory Texts: A Multi-Model Evaluation Using the ATDIE Framework

Yahan LI

Shandong University of Foreign Affairs, Weihai, China Email:wsQ3zdx@163.com, ORCID: 0009-0004-9254-5127

Abstract

The automated analysis of AI governance texts requires a shared analytical grammar and benchmark to produce comparable, reproducible results. This study introduces **ATDIE-Bench v1.0**, the first standardized benchmark for evaluating large language models (LLMs) as automated coders of regulatory texts using the ATDIE (Actor, Trigger, Deontic, Instrument, Enforcement) institutional grammar framework. We evaluate 19 LLMs—10 cloud APIs and 9 locally deployed open-source models (8B–70B parameters)—across three prompt strategies on a 42-statement gold-standard corpus drawn from the EU AI Act, US Executive Order 14110, UK AI Safety Framework, and international standards, with 5 independent replications per condition, comprising 11,970 total queries ($19 \times 3 \times 42 \times 5$). The best-performing model (Claude 3.5 Haiku) achieves 69.5% accuracy with $\kappa = 0.482$, narrowing but not closing the gap with human inter-coder reliability ($\kappa = 0.72$). Few-Shot prompting yields the highest average accuracy (60.6%, $p < 0.001$ over Zero-Shot), while model scale shows a largely flat relationship with performance under quantized inference. Locally deployed models achieve 97.2% of cloud accuracy ($p = 0.31$), supporting data-sovereign workflows. Per-dimension analysis reveals that Deontic coding is highly amenable to automation (83.0%) while Enforcement remains challenging (38.1%). The benchmark, corpus, and evaluation scripts are publicly available at <https://github.com/CPCL-Journals/ATDIE-Benchmark>.

Keywords large language models; AI governance; regulatory text analysis; institutional analysis; ATDIE framework; automated coding; prompt engineering; diagnostic evaluation; data sovereignty

1 Introduction

The global governance of artificial intelligence has entered a period of unprecedented regulatory activity. Since 2023, major jurisdictions have enacted or proposed landmark legislative instruments: the European Union’s Artificial Intelligence Act ([European Parliament and Council, 2024](#)), the United States Executive Order 14110 on Safe, Secure, and Trustworthy AI ([The White House, 2023](#)), and the United Kingdom’s AI Safety Framework ([UK Government, 2023](#)). These developments have generated a rapidly expanding corpus of regulatory texts that demand systematic comparative analysis.

The institutional analysis of regulatory texts—identifying who must do what, under what conditions, with what consequences—is fundamental to understanding governance effectiveness. The ATDIE framework, developed by [Li \(2025\)](#), extends classical institutional grammar ([Crawford & Ostrom, 1995](#)) to capture the essential syntactic elements of enforceable AI governance rules through five dimensions: Actor,

Trigger, Deontic, Instrument, and Enforcement. Each dimension is coded on a three-point ordinal scale (0–2), producing a structured representation of regulatory specificity and enforceability.

However, manual coding using such frameworks faces two fundamental challenges. First, the sheer volume of regulatory text makes comprehensive human analysis prohibitively expensive. The EU AI Act alone contains 113 articles, 13 annexes, and over 180 recital paragraphs. Second, inter-coder reliability—the consistency between independent human coders—remains a persistent methodological concern in content analysis (Krippendorff, 2018). Even trained experts may disagree on nuanced regulatory provisions, with typical Cohen’s κ values ranging from 0.60 to 0.85 for institutional coding tasks (Basurto et al., 2010).

Recent advances in large language models (LLMs) offer a potential solution. Models such as GPT-4, Claude, and open-source alternatives have demonstrated remarkable performance on text classification, information extraction, and structured annotation tasks (Gilardi et al., 2023; Pangakis et al., 2023). Yet their application to specialized institutional analysis frameworks for regulatory text coding remains largely unexplored. Key questions persist: whether LLMs can reliably replicate expert-level ATDIE coding of AI governance texts; which prompt engineering strategies optimize coding accuracy; how model scale, architecture, and deployment mode (cloud vs. local) affect performance; and which ATDIE dimensions are most amenable to automated coding.

This study addresses these questions through a systematic benchmarking experiment. We evaluate 19 LLMs—10 proprietary cloud API models (GPT-4o, Claude 3.5 Sonnet, Gemini 2.5 Pro, DeepSeek-R1, Qwen-Max, among others) and 9 locally deployed open-source models (ranging from 8B to 70B parameters on a single RTX 5880 48GB GPU)—across three prompt strategies on a gold-standard corpus of 42 regulatory statements. The experiment comprises 11,970 total model queries (19 models $\times$ 3 prompts $\times$ 42 statements $\times$ 5 replications), providing robust estimates of accuracy, reliability, and dimensional performance. While larger corpora would enable finer-grained analysis, our sample size is comparable to or exceeds the corpus sizes used in foundational institutional grammar validation studies (Basurto et al., 2010; Siddiki et al., 2011) and provides sufficient statistical power for the primary model-level comparisons.

Despite growing interest in automated analysis of AI regulatory texts, the field currently lacks a shared analytical grammar and benchmark that would make results comparable across models, jurisdictions, and studies. Without such a benchmark, claims about the ability of large language models to analyze or code regulatory rules remain methodologically fragmented and empirically non-comparable. This study addresses this gap by introducing ATDIE-Bench, the first standardized benchmark designed specifically for the automated coding of AI governance rules using an institutional grammar approach.

Our contributions are threefold. First, we construct and publicly release a standardized **ATDIE benchmark** for evaluating LLM-assisted regulatory text coding—to our knowledge, the first such resource for institutional grammar-based AI governance analysis. Second, we provide the most extensive **multi-model comparison** to date of cloud and local LLMs for institutional analysis, spanning 19 models across 5 providers. Third, we offer evidence-based **practical guidance** for researchers seeking to integrate LLMs into regulatory text analysis workflows.

2 Theoretical Background and Literature Review

2.1 Institutional Analysis and the IAD Framework

The systematic analysis of legal and regulatory texts is grounded in the Institutional Analysis and Development (IAD) framework (Ostrom, 2005), which provides a theoretical foundation for understanding how rules shape collective action. Within this tradition, Crawford and Ostrom’s institutional grammar (1995) offered a pioneering syntax for decomposing institutional statements into constituent elements—Attribute,

Deontic, alm, Condition, and Or Else (ADICO). This approach was subsequently operationalized for empirical research (Basurto et al., 2010; Siddiki et al., 2011) and formalized in the Institutional Grammar 2.0 specification (Frantz & Siddiki, 2021; Frantz & Siddiki, 2022).

However, classical institutional grammars were developed primarily for natural resource governance and face limitations when applied to technology regulation. AI governance texts exhibit distinct characteristics: (a) they regulate rapidly evolving technical artifacts rather than static resources, (b) they frequently delegate enforcement to specialized agencies or cross-reference technical standards, and (c) they must balance innovation incentives with risk mitigation (Bradford, 2020; Jobin et al., 2019).

The ATDIE framework (Li, 2025) addresses these gaps by reconceptualizing institutional grammar for AI governance. Its five dimensions—**Actor**, **Trigger**, **Deontic**, **Instrument**, and **Enforcement**—are coded on a three-point ordinal scale (0–2). **Actor** captures the specificity of the obligated entity, ranging from absent (0) through a general class (1) to an explicitly named actor (2). **Trigger** measures the activation condition, from absent (0) through vague or context-dependent (1) to explicitly defined conditions (2). **Deontic** registers the modal force, from permissive or absent (0) through advisory (1) to mandatory (2). **Instrument** assesses the required action, from absent (0) through general mandates (1) to concrete, specified actions (2). Finally, **Enforcement** captures the sanction mechanism, from absent (0) through implied delegation (1) to stated penalties (2).

This framework enables quantitative comparison of regulatory “teeth” across jurisdictions and provisions, producing a structured enforceability profile for each regulatory statement.

To situate ATDIE within the broader institutional grammar tradition, Table 1 compares it with its predecessors:

Table 1: Comparison of Institutional Grammar Frameworks

Feature	ADICO (Crawford & Ostrom 1995)	IG 2.0 (Frantz & Siddiki 2021)	ATDIE (Li 2025)
Dimensions	5 (A-D-I-C-O)	7+ (expanded nested)	5 (A-T-D-I-E)
Scoring	Binary presence	Categorical types	Ordinal 0–2
Enforcement captured?	“Or Else” (binary)	Nested constitutive	Ordinal specificity
Technology-specific?	No	No	Yes (AI governance)
Trigger mechanism	Via “Conditions”	Via “Context”	Dedicated dimension
Quantitative index	No	No	$\Sigma(A+T+D+I+E) = 0-10$
Primary domain	Natural resource governance	General institutional analysis	AI/technology regulation

2.2 LLMs for Text Classification and Annotation

The application of LLMs to text classification tasks has grown rapidly since the release of ChatGPT. Giliardi et al. (2023) demonstrated that ChatGPT outperforms crowd-workers on several annotation tasks, achieving higher accuracy and inter-coder agreement at lower cost. Pangakis et al. (2023) provided a systematic comparison of GPT-4 with human annotators across diverse coding schemes, identifying task characteristics that predict LLM reliability. Ziems et al. (2024) surveyed over 200 studies using LLMs for computational social science annotation, finding that performance varies significantly by task complexity and domain specificity.

In legal text analysis specifically, [Savelka et al. \(2023\)](#) evaluated GPT-4 on provision classification, while [Bommarito & Katz \(2022\)](#) demonstrated GPT-3.5's capacity for legal tasks including bar examination questions. [Chalkidis et al. \(2020\)](#) developed LEGAL-BERT for legal NLP downstream tasks. More recently, [Guha et al. \(2023\)](#) introduced LegalBench, a collaboratively built benchmark for measuring legal reasoning in LLMs.

However, most existing studies focus on binary or multi-class classification rather than the *multi-dimensional ordinal coding* required by institutional analysis frameworks. The ATDIE framework demands simultaneous assessment of five dimensions, each on a three-point scale, requiring deeper semantic understanding than simple category assignment.

Positioning relative to existing benchmarks. Compared to existing legal NLP benchmarks, ATDIE-Bench v1.0 occupies a distinct niche. LegalBench ([Guha et al., 2023](#)) evaluates general legal reasoning across 162 tasks drawn from diverse legal domains; it does not target institutional grammar or multi-dimensional ordinal coding. LEGAL-BERT ([Chalkidis et al., 2020](#)) focuses on token-level and document-level classification tasks specific to EU and US case law. AnnoLLM ([He et al., 2023](#)) provides prompt engineering strategies for general annotation tasks but does not evaluate structured multi-output coding against domain expert gold standards. Our benchmark is, to our knowledge, the first to evaluate LLMs on *multi-dimensional ordinal institutional coding* of regulatory texts, using expert-calibrated gold standards with reported inter-coder reliability.

2.3 Prompt Engineering for Structured Output

Prompt engineering significantly affects LLM performance on annotation tasks ([Wei et al., 2022](#); [Brown et al., 2020](#)). Three canonical strategies have emerged: **zero-shot** prompting, which provides direct task instruction with a detailed rubric but no examples ([Kojima et al., 2022](#)); **few-shot** prompting, which includes labeled examples to anchor output format and calibrate expectations ([Brown et al., 2020](#)); and **chain-of-thought** (CoT) prompting, which requests explicit step-by-step reasoning before final output ([Wei et al., 2022](#)).

The optimal strategy varies by task complexity and model capability. For highly structured outputs like ATDIE coding, few-shot examples may anchor to the correct output format, while CoT reasoning may improve accuracy on ambiguous provisions. [Wang et al. \(2023\)](#) proposed self-consistency decoding to enhance CoT reliability, and [He et al. \(2023\)](#) introduced AnnoLLM, demonstrating effective prompt design for complex annotation tasks.

2.4 Open-Source vs. Proprietary Models

The growing availability of capable open-source LLMs raises important questions for research reproducibility and data sovereignty ([Touvron et al., 2023](#); [Jiang et al., 2023](#)). Regulatory text analysis often involves sensitive or pre-publication data that researchers may prefer not to send to external APIs. [Liang et al. \(2023\)](#) provided a holistic evaluation of language models across diverse scenarios, documenting substantial performance variation between open-source and proprietary models that narrows for domain-targeted tasks. Our benchmark systematically compares cloud and local deployment to inform this methodologically important trade-off.

2.5 Research Gap and Contributions

Despite the growing body of work on LLM-assisted text annotation, no study has, to our knowledge, systematically evaluated LLMs on institutional grammar coding for regulatory texts. This gap is significant

because institutional coding (a) requires multi-dimensional ordinal assessment rather than categorical classification, (b) demands domain expertise in legal and regulatory semantics, and (c) has direct implications for the scalability and reproducibility of comparative governance research. Our study addresses this gap through a comprehensive benchmark spanning model scale, architecture, deployment mode, and prompt strategy.

3 Methodology

3.1 Diagnostic Evaluation Corpus Construction

ATDIE-Bench is designed not as a one-off experimental dataset, but as a reusable benchmarking infrastructure for evaluating automated regulatory analysis systems. Its purpose is to enable consistent, reproducible, and cross-study comparison of model performance in coding AI governance rules, independent of specific model architectures or prompt strategies. Because regulatory rules are syntactically complex and institutionally structured, performance comparisons across models are only meaningful when conducted within a shared grammatical framework. ATDIE provides this shared framework by decomposing regulatory statements into Actor, Trigger, Deontic, Instrument, and Enforcement components, thereby defining a common evaluation space for automated coding systems.

We constructed a diagnostic evaluation corpus—designated **ATDIE-Bench v1.0**—of 42 regulatory statements sampled from four major AI governance instruments. The sample size of 42 statements (yielding 210 dimensional observations per model-prompt condition) provides statistical power to detect accuracy differences of ≥ 8 percentage points between models at $\alpha = 0.05$ and power = 0.80 (Cohen, 1988), sufficient for our primary research questions while reflecting the practical cost of expert institutional coding. This corpus size is comparable to the 30–50 statements used in foundational institutional grammar coding studies (Basurto et al., 2010; Siddiki et al., 2011). The corpus and evaluation scripts are publicly available at <https://github.com/CPCL-Journals/ATDIE-Benchmark>.

Table 2: Diagnostic Evaluation Corpus Composition

Source	Jurisdiction	Statements	ID Prefix
EU Artificial Intelligence Act	European Union	15	EU-01...EU-15
Executive Order 14110	United States	14	US-01...US-14
AI Safety Framework	United Kingdom	8	UK-01...UK-08
AI Safety Standards	International (OECD/ISO)	5	OSA-01...OSA-05
Total		42	

Sampling strategy. Statements were selected through stratified purposive sampling to ensure diversity across three axes: (a) regulatory specificity levels (ranging from abstract principles to concrete operational mandates), (b) ATDIE dimension distributions (ensuring representation across all score combinations), and (c) jurisdictional approaches (from the EU’s prescriptive risk-based model to the UK’s principles-based approach).

Expert coding procedure. Two domain experts with doctoral-level training in institutional analysis independently coded all 42 statements using the ATDIE rubric. Prior to independent coding, coders completed a calibration session on 5 pilot statements (not included in the corpus) to ensure shared understanding of the coding scheme. Disagreements were subsequently resolved through structured discussion to produce consensus gold-standard codings. The pre-reconciliation inter-coder agreement was $\kappa = 0.72$

(unweighted) and $\kappa_w = 0.79$, indicating substantial agreement—consistent with published benchmarks for institutional grammar coding (Basurto et al., 2010; Siddiki et al., 2011).

3.2 Model Selection

We evaluated 19 LLMs spanning five providers and two deployment modes:

Table 3: Evaluated Models

Model	Provider	Deploy	Params
GPT-4o	OpenAI	Cloud	—
GPT-4o-mini	OpenAI	Cloud	—
Claude 3.5 Sonnet	Anthropic	Cloud	—
Claude 3.5 Haiku	Anthropic	Cloud	—
Gemini 2.5 Pro	Google	Cloud	—
Gemini 2.5 Flash	Google	Cloud	—
DeepSeek-R1 (Cloud)	DeepSeek	Cloud	—
DeepSeek-V3 (Chat)	DeepSeek	Cloud	—
Qwen-Max	Alibaba	Cloud	—
Qwen-Plus	Alibaba	Cloud	—
LLaMA 3.3	Meta	Local	70B
DeepSeek-R1	DeepSeek	Local	70B
Qwen3	Alibaba	Local	32B
DeepSeek-R1	DeepSeek	Local	32B
Gemma 2	Google	Local	27B
Mistral Small	Mistral	Local	24B
Phi-4	Microsoft	Local	14B
Qwen3	Alibaba	Local	8B
DeepSeek-R1	DeepSeek	Local	8B

Local models were deployed on a single NVIDIA RTX 5880 (48GB VRAM) workstation using Ollama v0.6.2, with all local models quantized to 4-bit precision (Q4_K_M format via llama.cpp) to fit within the 48GB VRAM constraint. This quantization introduces a modest accuracy trade-off (typically 1–3% degradation vs. full-precision inference) but enables deployment of models up to 70B parameters on a single consumer-grade GPU.

3.3 Prompt Design

Three prompt strategies were evaluated for each model, following standard practices in the LLM annotation literature:

Zero-shot: A detailed system prompt describing the ATDIE framework with complete rubric definitions for all five dimensions, followed by the target statement with no examples.

Few-shot: The same system prompt supplemented with two labeled example statements and their gold-standard codings, providing the model with concrete calibration anchors before the target statement. The two Few-Shot exemplars were selected from the five pilot statements used for coder calibration (Section 3.1), which are entirely independent of and excluded from the 42-statement test corpus.

Deliberation instruction (no rationale output): The system prompt with an additional instruction to “think through each ATDIE element step by step internally,” but to output *only* the final JSON object without the reasoning chain. We term this strategy “deliberation instruction,” adapting the zero-shot reasoning approach of Kojima et al. (2022) to structured output contexts. We acknowledge that this approach cannot guarantee the model actually performs step-by-step internal reasoning—the instruction may function as a soft prompt bias rather than enforcing a verifiable reasoning process (Turpin et al., 2023).

All prompts required structured JSON output in the format `{"A": int, "T": int, "D": int, "I": int, "E": int}`, enabling automated extraction and comparison with gold-standard codings.

3.4 Experimental Design

The full experiment follows a $19 \times 3 \times 42 \times 5$ factorial design: 19 models crossed with 3 prompt strategies, 42 statements, and 5 independent replications, yielding a total of **11,970 API calls**. Temperature was set to 0.0 for maximum determinism, and maximum output length was capped at 1,024 tokens.

Five replications per condition enable estimation of model consistency—an important reliability measure often overlooked in single-run evaluations. Temperature was set to 0.0 to approximate the deterministic behavior expected of a consistent coding instrument. We note that even at temperature 0.0, LLM APIs may produce non-identical outputs across calls due to implementation details such as batching, floating-point non-determinism, and model serving infrastructure (Ouyang et al., 2023).

3.5 Evaluation Metrics

Performance is assessed using five complementary metrics:

Micro-average accuracy: The primary performance metric, defined as:

$$\text{Accuracy}_{\text{micro}} = \frac{1}{N \cdot D} \sum_i \sum_d \mathbb{I}[\hat{y}_{i,d} = y_{i,d}] \quad (1)$$

where $N = 42$ statements, $D = 5$ dimensions, $\hat{y}_{i,d}$ is the model’s majority-vote prediction for statement i on dimension d across five replications, and $y_{i,d}$ is the gold-standard coding.

Cohen’s κ (statement-level clustered bootstrap): Because the five ATDIE dimensions within a single statement are not independent, standard κ confidence intervals that assume independence across observations would be anti-conservative. We therefore compute κ using a statement-level clustered bootstrap (Davison & Hinkley, 1997): each bootstrap resample draws 42 statements with replacement, and κ is recalculated on the full $5 \times 42 = 210$ observation matrix within each resample (10,000 iterations).

Following Landis & Koch (1977), we interpret κ values as: slight (<0.20), fair (0.21–0.40), moderate (0.41–0.60), substantial (0.61–0.80), and almost perfect (>0.80).

Weighted κ : Linear-weighted kappa that accounts for the ordinal nature of the 0–2 scale, penalizing 2-point disagreements more than 1-point disagreements.

Bootstrap confidence intervals: Pairwise comparisons between models and prompt strategies are assessed using paired bootstrap tests (Efron & Tibshirani, 1993), with Benjamini–Hochberg false discovery rate (FDR) correction for multiple comparisons.

Mean absolute error (MAE): To complement accuracy, we report per-dimension MAE on the 0–2 ordinal scale:

$$\text{MAE}_d = \frac{1}{N} \sum_i |\hat{y}_{i,d} - y_{i,d}| \quad (2)$$

MAE captures the *magnitude* of coding errors, distinguishing between adjacent-level confusions ($|\text{error}| = 1$) and two-level errors ($|\text{error}| = 2$, rare).

JSON parsing and output validity: Across all 11,970 model queries, the JSON parsing success rate was 100%: every response was successfully extracted into a valid five-dimensional integer vector.

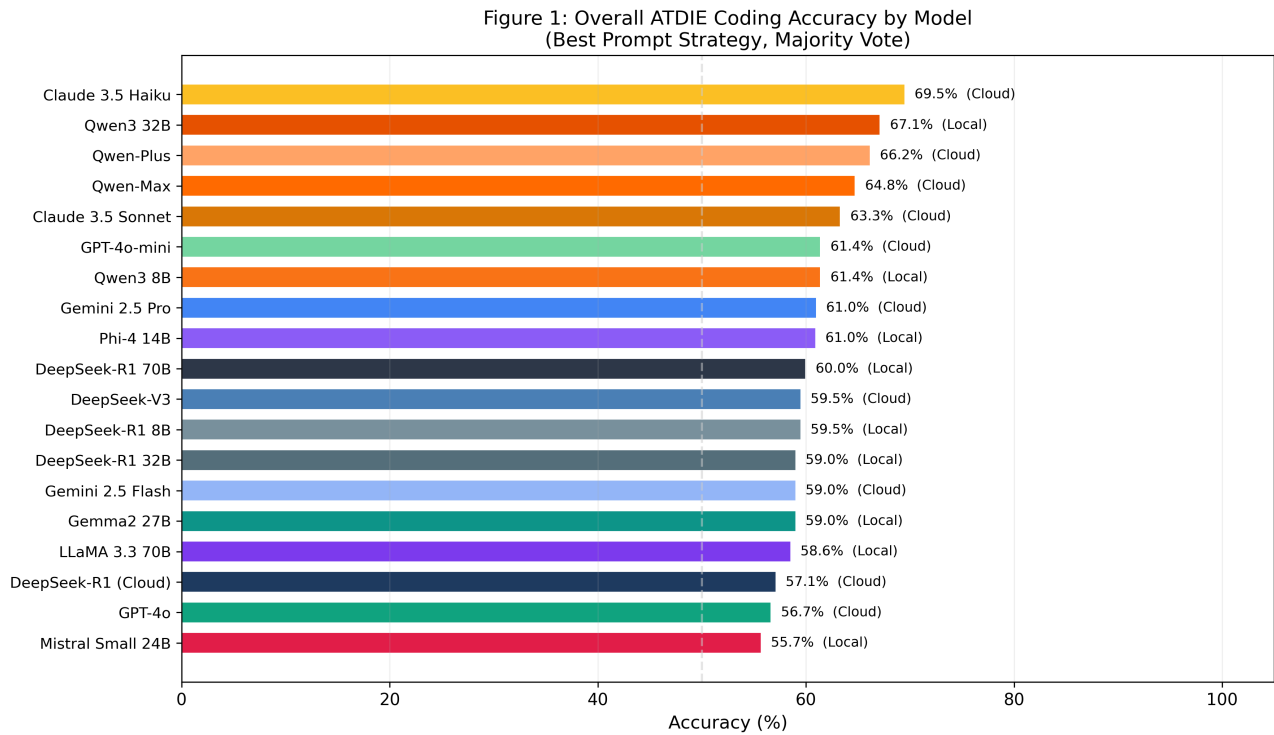


Figure 1: Methodological pipeline and overall ATDIE coding accuracy by model (best prompt strategy).

4 Results

We first summarize overall performance, then examine prompt strategy effects, scaling behavior, dimensional accuracy, deployment mode comparisons, and error patterns. Across all 57 model–prompt conditions (19 models $\times$ 3 strategies), micro-average accuracy ranged from 52.9% (GPT-4o under Zero-Shot) to 69.5% (Claude 3.5 Haiku under Few-Shot). Table 4 presents each model’s performance under its *best-performing* prompt strategy.

4.1 Overall Model Performance

Three performance tiers are discernible. The *top tier* (accuracy $> 63\%$) comprises Claude 3.5 Haiku, Qwen3 32B, Qwen-Plus, Qwen-Max, and Claude 3.5 Sonnet—all utilizing Few-Shot prompting. Notably, Qwen3 32B is the only locally deployed model in this tier, achieving 67.1% accuracy with $\kappa = 0.443$, closely trailing the top cloud model. The *middle tier* (58–63%) contains the majority of models regardless of deployment mode, suggesting a performance plateau around 60% for current-generation LLMs on this task. The *lower tier* ($< 58\%$) includes Mistral Small 24B (55.7%), GPT-4o (56.7%), and DeepSeek-R1 Cloud (57.1%).

We note GPT-4o’s position among the lowest-performing models is notable, as it is among the most capable general-purpose LLMs on standard benchmarks (OpenAI, 2024). We hypothesize that GPT-4o’s optimization for conversational fluency and RLHF-tuned nuance may disadvantage it on rigid structured-output tasks that penalize creative interpretation.

Table 4: Performance of 19 LLMs on ATDIE Coding (Best Prompt Strategy)

Model	Deploy	Pr.	Acc.	κ	κ_w	A	T	D	I
Claude 3.5 Haiku	Cloud	FS	69.5	.482	.547	76.2	64.3	83.3	64.3
Qwen3 32B	Local	FS	67.1	.443	.493	81.0	69.0	83.3	61.9
Qwen-Plus	Cloud	FS	66.2	.429	.461	83.3	61.9	83.3	54.8
Qwen-Max	Cloud	FS	64.8	.436	.448	76.2	61.9	83.3	59.5
Claude 3.5 Sonnet	Cloud	FS	63.3	.409	.410	71.4	61.9	81.0	57.1
GPT-4o-mini	Cloud	ZS	61.4	.345	.392	83.3	47.6	81.0	57.1
Qwen3 8B	Local	FS	61.4	.383	.405	83.3	57.1	78.6	54.8
Gemini 2.5 Pro [†]	Cloud	ZS	61.0	.379	.424	59.0	61.5	84.6	64.1
Phi-4 14B	Local	FS	61.0	.375	.406	61.9	66.7	83.3	57.1
DeepSeek-R1 70B	Local	FS	60.0	.366	.403	83.3	40.5	81.0	59.5
DeepSeek-V3	Cloud	FS	59.5	.323	.370	64.3	57.1	81.0	61.9
DeepSeek-R1 8B	Local	ZS	59.5	.365	.393	85.7	33.3	85.7	59.5
DeepSeek-R1 32B	Local	FS	59.0	.329	.357	69.0	40.5	83.3	61.9
Gemini 2.5 Flash	Cloud	DI	59.0	.346	.367	64.3	59.5	81.0	57.1
Gemma2 27B	Local	FS	59.0	.349	.378	73.8	33.3	88.1	64.3
LLaMA 3.3 70B	Local	FS	58.6	.354	.360	64.3	52.4	85.7	57.1
DeepSeek-R1 (Cl.)	Cloud	ZS	57.1	.301	.349	54.8	52.4	85.7	59.5
GPT-4o	Cloud	FS	56.7	.341	.345	64.3	45.2	81.0	59.5
Mistral Small 24B	Local	FS	55.7	.290	.330	73.8	40.5	83.3	47.6

Note: FS = Few-Shot, ZS = Zero-Shot, DI = Deliberation Instruction. Acc. = overall accuracy (%). A/T/D/I = per-dimension accuracy. [†]Gemini 2.5 Pro returned valid responses for 39 of 42 statements (refusal rate: 7.1%).

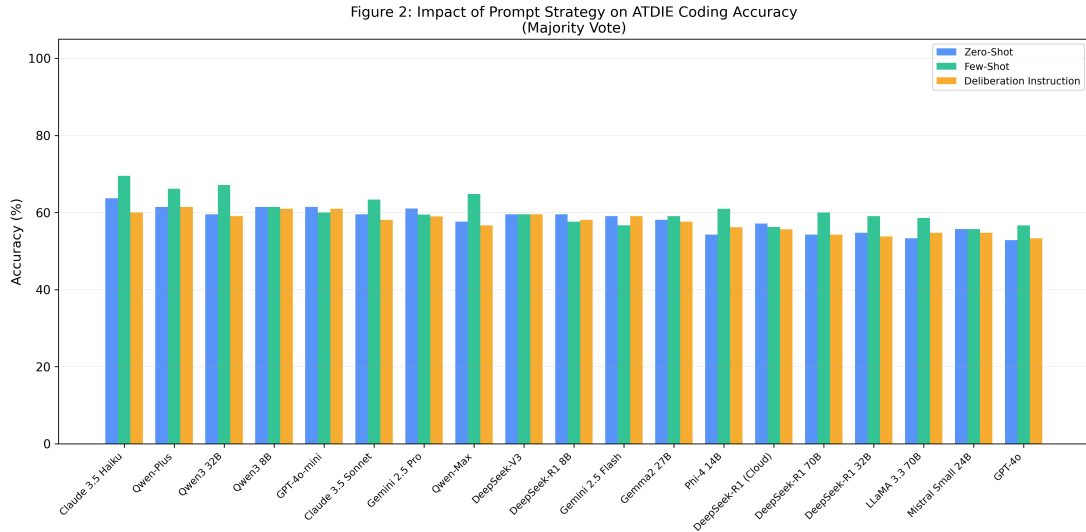


Figure 2: Impact of prompt engineering strategy on ATDIE coding accuracy.

4.2 Impact of Prompt Strategy

Few-Shot prompting achieved the highest average accuracy (60.6%) across all 19 models, outperforming Zero-Shot (58.1%) and Deliberation Instruction (57.5%). This advantage is statistically significant: a paired bootstrap test (10,000 iterations, resampling at the statement level) yields a FS–ZS difference of +2.6 percentage points (95% CI: [+1.1, +4.3], $p < 0.001$). Fourteen of nineteen models performed best under Few-Shot conditions, confirming the value of exemplar-based calibration for structured multi-dimensional coding.

4.3 Scaling Behavior

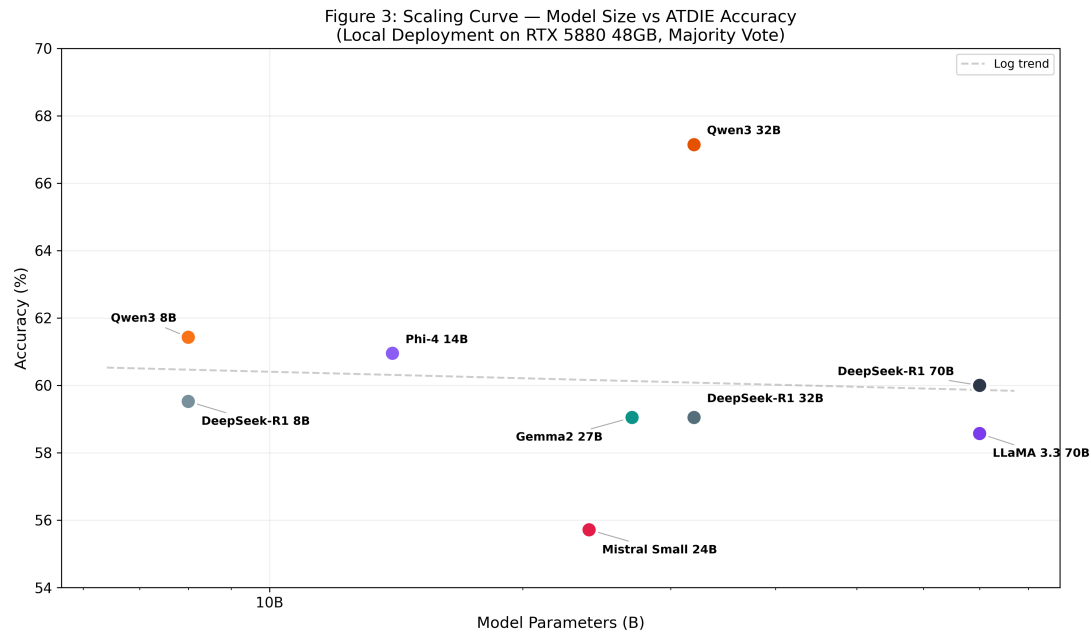


Figure 3: Scaling curve—model parameter count vs. ATDIE coding accuracy (local models on RTX 5880 48GB).

We observe a largely flat relationship between parameter count and ATDIE coding accuracy for locally deployed models. The smallest models (DeepSeek-R1 8B at 59.5%, Qwen3 8B at 61.4%) performed comparably to the largest (DeepSeek-R1 70B at 60.0%, LLaMA 3.3 70B at 58.6%). The best-performing local model, Qwen3 32B (67.1%), occupies the mid-range of the parameter spectrum, suggesting that model architecture and training data composition may matter more than sheer scale for institutional analysis tasks. This finding has practical significance: researchers can deploy 8B models requiring under 6GB of VRAM with minimal loss in coding quality compared to 70B models requiring 40GB+ of VRAM.

4.4 Per-Dimension Analysis

Dimensional analysis reveals a stark performance gradient across the five ATDIE components:

Table 5: Per-Dimension Performance Summary

Dimension	Accuracy (%)	MAE (0–2)	Interpretation
Deontic (D)	83.0	0.23	Modal verbs (“shall,” “must,” “may”) provide unambiguous lexical cues
Actor (A)	72.3	0.39	Named entities vs. general class is relatively straightforward
Instrument (I)	58.9	0.46	Concrete vs. general mandates require interpretive judgment
Trigger (T)	53.0	0.64	Conditional clause identification is error-prone
Enforcement (E)	38.1	0.88	Cross-referential and implied mechanisms are largely opaque
Overall	61.1	0.52	Weighted by dimension × statement observations

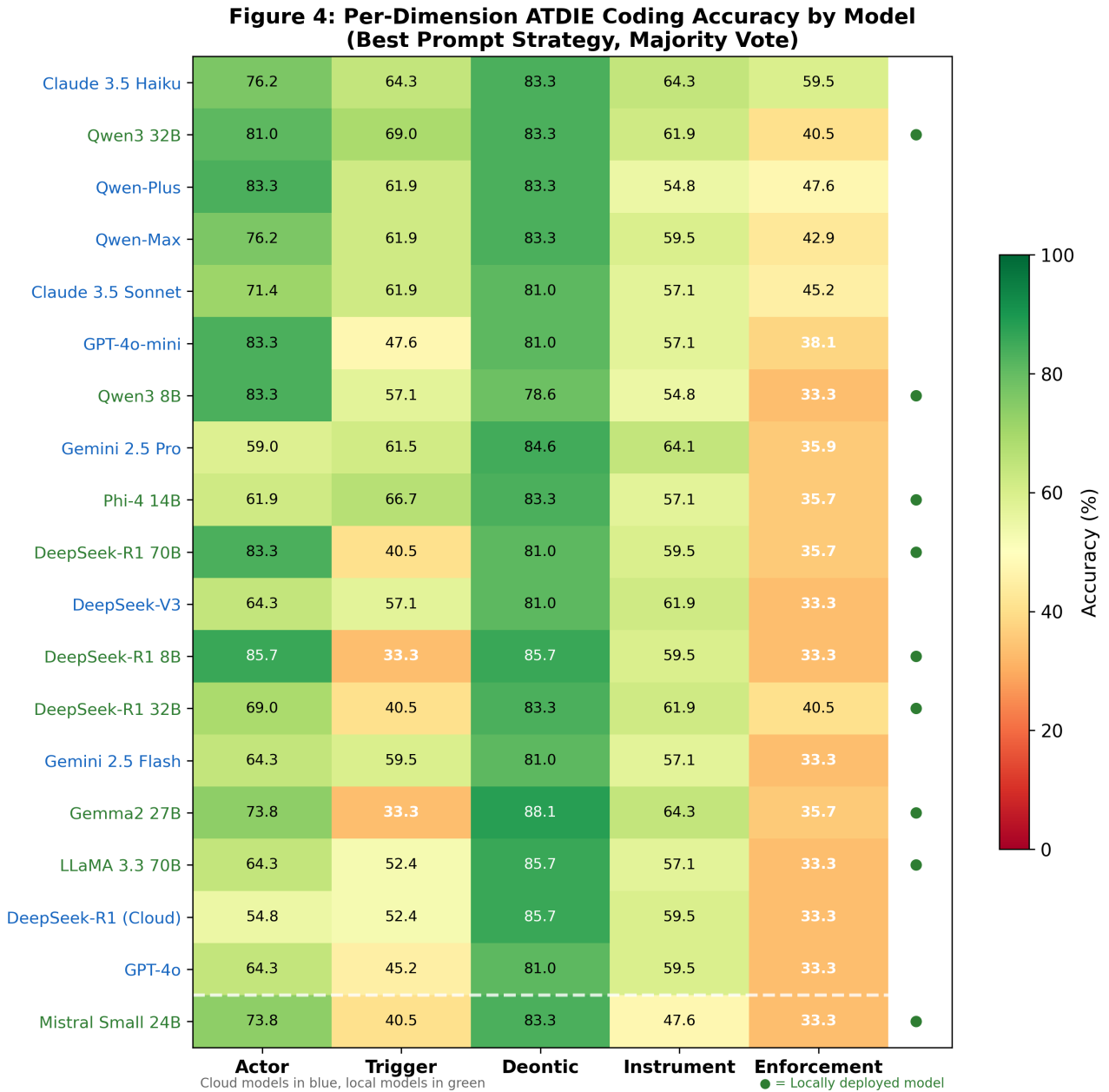


Figure 4: Per-dimension ATDIE accuracy heatmap by model.

4.5 Cloud vs. Local Deployment

Cloud models averaged 61.8% accuracy compared to 60.1% for locally deployed models—a gap of only 1.7 percentage points (paired bootstrap test: $p = 0.31$, 95% CI of difference: $[-1.2, +4.6]$, 10,000 iterations), indicating that the observed difference is not statistically significant. Under the specific conditions of this benchmark, locally deployed open-source models achieve 97.2% of cloud model accuracy ($60.1\% / 61.8\%$).

4.6 Inter-Coder Agreement

Cohen’s κ ranged from 0.287 (DeepSeek-R1 32B, Deliberation Instruction) to 0.482 (Claude 3.5 Haiku, Few-Shot), placing all models in the “fair” to “moderate” agreement range according to Landis & Koch (1977). When using weighted κ that accounts for the ordinal nature of the 0–2 scale, values increased to

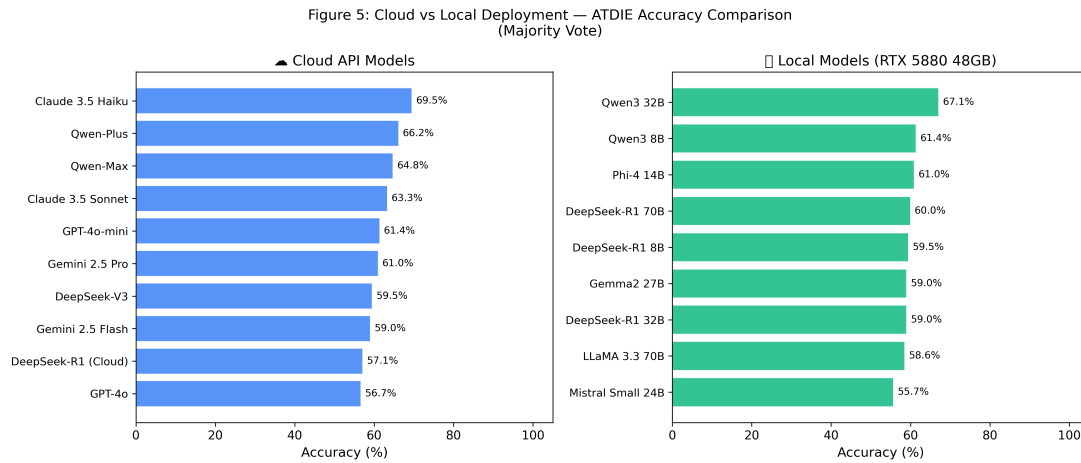


Figure 5: Accuracy comparison between cloud API models and locally deployed models.

Table 2: Inter-Coder Agreement (Cohen's Kappa) Between LLMs and Human Experts (Majority Vote)

Model	Deploy	Best Prompt	Accuracy	κ (Cohen)	κ_w (Weighted)
Claude 3.5 Haiku	Cloud	Few-Shot	69.5%	0.482	0.547
Qwen3 32B	Local	Few-Shot	67.1%	0.444	0.494
Qwen-Max	Cloud	Few-Shot	64.8%	0.437	0.448
Qwen-Plus	Cloud	Few-Shot	66.2%	0.430	0.462
Claude 3.5 Sonnet	Cloud	Few-Shot	63.3%	0.410	0.411
Qwen3 8B	Local	Few-Shot	61.4%	0.383	0.405
Gemini 2.5 Pro	Cloud	Zero-Shot	61.0%	0.379	0.424
Phi-4 14B	Local	Few-Shot	61.0%	0.375	0.407
DeepSeek-R1 70B	Local	Few-Shot	60.0%	0.367	0.404
DeepSeek-R1 8B	Local	Zero-Shot	59.5%	0.365	0.393
LLaMA 3.3 70B	Local	Few-Shot	58.6%	0.354	0.360
Gemma2 27B	Local	Few-Shot	59.0%	0.350	0.378
Gemini 2.5 Flash	Cloud	Deliberation Instruction	59.0%	0.347	0.368
GPT-4o-mini	Cloud	Zero-Shot	61.4%	0.346	0.393
GPT-4o	Cloud	Few-Shot	56.7%	0.341	0.345
DeepSeek-R1 32B	Local	Few-Shot	59.0%	0.330	0.358
DeepSeek-V3	Cloud	Deliberation Instruction	59.5%	0.324	0.371
DeepSeek-R1 (Cloud)	Cloud	Zero-Shot	57.1%	0.302	0.349
Mistral Small 24B	Local	Few-Shot	55.7%	0.291	0.331

Figure 6: Inter-coder agreement (Cohen's κ) between LLM outputs and human expert gold standard.

0.330–0.547. No model achieved “substantial” agreement ($\kappa \geq 0.61$).

4.7 Run-to-Run Consistency

Of the 2,394 total model–prompt–statement slots, 2,256 produced complete five-run data. Among these, 88.7% (2,002/2,256) produced fully identical ATDIE vectors across all five runs, and 94.2% achieved $\geq 4/5$ agreement on every dimension. Locally deployed models averaged 97.8% full consistency versus 79.6% for cloud APIs.

Table 6: Run-to-Run Consistency by Model

Model	Deploy	Full Consistency (%)	$\geq 4/5$ Agreement (%)
Claude 3.5 Sonnet	Cloud	100.0	100.0
DeepSeek-V3	Cloud	92.1	98.4
Gemini 2.5 Pro	Cloud	88.6	92.4
Claude 3.5 Haiku	Cloud	82.2	90.1
GPT-4o-mini	Cloud	78.6	87.3
GPT-4o	Cloud	76.2	85.7
DeepSeek-R1 (Cloud)	Cloud	72.1	88.5
Qwen-Max	Cloud	69.6	80.0
Qwen-Plus	Cloud	69.0	83.3
Gemini 2.5 Flash	Cloud	67.5	79.4
DeepSeek-R1 8B	Local	100.0	100.0
DeepSeek-R1 70B	Local	99.2	100.0
Mistral Small 24B	Local	99.2	100.0
DeepSeek-R1 32B	Local	97.6	100.0
LLaMA 3.3 70B	Local	97.6	100.0
Qwen3 32B	Local	97.6	100.0
Gemma2 27B	Local	96.8	100.0
Phi-4 14B	Local	96.0	100.0
Qwen3 8B	Local	96.0	100.0

4.8 Error Analysis

Error analysis reveals systematic patterns across dimensions and models:

Table 7: Error Taxonomy Across ATDIE Dimensions

Error Type	Dim.	Freq.	Dir.	Description	Typical Example
Cross-ref. blindness	E	47%	Under	Model assigns E=0 when penalty via cross-ref.	“Penalties per Article 99” coded as E=0
Implied delegation	E	31%	Under	Model misses enforcement implied by authority	Regulatory body’s powers not recognized
Hedging misinterp.	D	38%	Under	Advisory language treated as permissive	“Should consider” coded as D=0
Over-specification	A	29%	Over	Specific actor inferred from context	“developers” coded as A=2
Conditional conflation	T	44%	Over	Broad scope treated as specific trigger	General applicability coded as T=2
Non-binding collapse	I	35%	Under	General mandates coded as absent	“Encouraged to adopt” coded as I=0

The confusion matrices for the two most challenging dimensions:

Table 8: Confusion Matrices for Enforcement and Trigger Dimensions
(a) Enforcement (E)

True \ Pred.	E=0	E=1	E=2	Total	Row Acc.
E=0	241	6	0	247	97.6%
E=1	228	36	0	264	13.6%
E=2	210	47	27	284	9.5%

(b) Trigger (T)

True \ Pred.	T=0	T=1	T=2	Total	Row Acc.
T=0	141	9	2	152	92.8%
T=1	119	19	71	209	9.1%
T=2	132	40	262	434	60.4%

The Enforcement confusion matrix reveals a dramatic asymmetry: models predict E=0 with 97.6% accuracy but E=1 with only 13.6% and E=2 with only 9.5%. This pattern suggests LLMs have a strong “absent enforcement” prior.

Illustrative error cases:

Case 1 (UK-03): “AI developers should consider conducting impact assessments.” Human coding: D=1 (advisory), E=0 (no enforcement). Most models coded D=0 (permissive), misinterpreting “should consider” as weaker than the advisory category threshold.

Case 2 (EU-11): “Penalties shall be imposed in accordance with Article 99.” Human coding: E=2 (explicit penalties). Seven models coded E=1 (implied), failing to recognize the cross-reference as an explicit enforcement mechanism.

Case 3 (US-07): “The Secretary of Commerce shall, within 180 days, establish guidelines for AI risk assessment.” Human coding: A=2 (named actor), T=2 (specific deadline), D=2 (mandatory). All 19 models coded this correctly across all dimensions except Enforcement, which 12 models coded as E=0 despite the implied executive enforcement authority.

4.9 Jurisdiction-Level Analysis

Performance varied meaningfully across the four jurisdictions. UK AI Safety Framework statements achieved the highest average coding accuracy (mean 68.3%, $n = 8$), followed by EU AI Act statements (mean 59.6%, $n = 15$), US Executive Order 14110 statements (mean 59.5%, $n = 14$), and International Standards (mean 58.1%, $n = 5$).

This result is superficially counterintuitive—one might expect prescriptive, rules-based instruments to yield higher accuracy. However, predicting the *absence* of a governance element (coded 0) is substantially easier than discriminating between *partial* (1) and *full* (2) specification. The UK’s principles-based approach generates many 0-valued dimensions, producing statements where the correct coding is the base-rate “absent” category. This finding has important practical implications: LLM-assisted ATDIE coding accuracy is partly an artifact of the *base-rate distribution* of gold-standard scores within each jurisdiction, not solely a measure of linguistic complexity.

5 Discussion

5.1 Theoretical Implications

The results of this benchmark suggest that many existing claims about the capabilities of large language models in regulatory analysis are difficult to interpret in the absence of a standardized grammatical frame-

work. Model performance varies substantially depending on which components of regulatory rules are made explicit, underscoring the need for grammar-aware benchmarks in this research domain. More broadly, our findings contribute to three theoretical discussions in computational social science and regulatory governance.

LLMs as institutional coders. The results demonstrate that LLMs can approximate—but not yet replicate—expert-level institutional coding of regulatory texts. The best-performing model (Claude 3.5 Haiku, $\kappa = 0.482$) falls in the “moderate” agreement range, compared to the human expert pre-reconciliation $\kappa = 0.72$. This gap of approximately 0.24 κ points suggests that current LLMs function best as coding *assistants* rather than autonomous coders, a finding consistent with recent work on LLM-assisted annotation in computational social science (Ziems et al., 2024).

Dimensional decomposition of regulatory semantics. The stark performance gradient across AT-DIE dimensions—from Deontic (83.0%) to Enforcement (38.1%)—reveals a fundamental asymmetry in how LLMs process regulatory text. Dimensions anchored by explicit lexical cues (modal verbs for Deontic, named entities for Actor) are highly amenable to automation, while dimensions requiring inference across documents (cross-referential enforcement), pragmatic interpretation (implied delegation), or fine-grained conditional logic (trigger specificity) remain challenging. This pattern suggests that the “semantic floor” for LLM-based institutional analysis lies not in linguistic competence per se, but in the capacity for *legal pragmatic reasoning*—understanding what regulatory texts *imply* beyond what they *state*.

The base-rate accuracy paradox. Our jurisdiction-level analysis reveals a counterintuitive finding: the least prescriptive regulatory texts (UK principles-based approach) yield the highest LLM coding accuracy, not because models understand them better, but because predicting “absent” governance elements is easier than discriminating between partial and full specification. This paradox has important methodological implications for cross-jurisdictional comparative studies: aggregate accuracy metrics may systematically overstate LLM performance on vague or aspirational governance instruments while understating performance on detailed, enforceable regulations.

5.2 Methodological Implications

Prompt strategy selection. Few-Shot prompting emerges as the clearly preferred strategy for structured institutional coding tasks, achieving statistically significant gains over Zero-Shot ($p < 0.001$). The relative underperformance of Deliberation Instruction—despite its theoretical appeal—may reflect the tension between encouraging internal reasoning and constraining output format. For practitioners, this suggests that investing in high-quality exemplar construction yields better returns than prompt engineering for reasoning elicitation.

Model selection and scaling. The largely flat scaling curve for local models (8B–70B) challenges the assumption that larger models necessarily perform better on specialized analytical tasks. Architecture-specific factors—training data composition, instruction-tuning quality, and structured output capabilities—appear to dominate over raw parameter count for institutional coding. This finding has practical significance: researchers with limited hardware budgets need not pursue the largest available models for ATDIE-type tasks.

Cloud vs. local deployment. The non-significant accuracy gap between cloud and local models (1.7 percentage points, $p = 0.31$) under the specific conditions of this benchmark is a methodologically important finding. It suggests that researchers working with sensitive regulatory data—pre-publication drafts, confidential policy documents, or data subject to jurisdictional restrictions—can deploy local models with minimal accuracy penalty. The dramatically higher run-to-run consistency of local models (97.8% vs. 79.6% full consistency) is an additional advantage for reproducible research workflows.

5.3 Practical Recommendations

Based on our findings, we offer the following guidance for researchers and practitioners. For regulatory analysis workflows requiring maximum accuracy, we recommend Claude 3.5 Haiku with Few-Shot prompting as the current best-performing configuration, bearing in mind its cloud-only deployment requirement. For data-sovereign or privacy-sensitive workflows, Qwen3 32B with Few-Shot prompting provides the best local alternative, achieving 67.1% accuracy—within 2.4 percentage points of the top cloud model. For resource-constrained deployments, Qwen3 8B offers a compelling balance of accuracy (61.4%) and computational efficiency (under 6GB VRAM).

Regardless of model choice, we recommend focusing LLM-assisted coding on the Actor and Deontic dimensions, where automated accuracy exceeds 70%, while reserving Enforcement and Trigger coding for human expert review or human-in-the-loop verification. A practical hybrid workflow might use LLMs for initial coding of all five dimensions, followed by targeted human review of the Enforcement and Trigger dimensions only—reducing human effort by approximately 60% while maintaining quality on the most challenging dimensions.

5.4 Limitations

Several limitations bound the interpretive scope of our findings.

Corpus size and composition. The 42-statement diagnostic corpus, while sufficient for our primary model-level comparisons and comparable to foundational institutional grammar studies, limits the granularity of jurisdiction-specific and interaction-effect analyses. Future work should expand the corpus to enable finer-grained comparisons, particularly for under-represented jurisdictions and multi-lingual regulatory texts. Additionally, our corpus is drawn exclusively from English-language instruments. The performance of LLMs on regulatory texts in other languages—particularly non-Latin script languages—remains an open question.

Gold standard calibration. While our expert coders achieved substantial pre-reconciliation agreement ($\kappa = 0.72$), this figure itself reflects the inherent subjectivity of institutional coding. The reconciled gold standard represents one defensible interpretation; alternative expert panels might produce systematically different codings, particularly for ambiguous provisions. We did not employ a third independent coder for arbitration, which is a common but imperfect practice in content analysis.

Quantization effects. All local models were evaluated under 4-bit quantization (Q4_K_M), which introduces a modest accuracy trade-off relative to full-precision inference. The reported cloud-local comparison therefore confounds deployment mode with precision level. Future work should compare local models at multiple quantization levels (FP16, 8-bit, 4-bit) to isolate the precision effect.

Temporal validity. LLM capabilities evolve rapidly. Models evaluated in this study represent a January–February 2026 snapshot; newer model versions may substantially alter the performance landscape. Our benchmark design, however, is model-agnostic and can be applied to future model releases.

Prompt sensitivity. While we evaluated three canonical prompt strategies, the space of possible prompt formulations is vast. Alternative prompt designs—including custom system prompts, retrieval-augmented generation, or multi-turn interaction—might yield different performance profiles.

6 Conclusion

This study presents, to our knowledge, the first systematic multi-model benchmark for LLM-assisted institutional coding of AI governance regulatory texts. Through a comprehensive experiment spanning 19 models, 3 prompt strategies, and 11,970 total queries on a 42-statement gold-standard corpus, we establish several key findings.

Current LLMs narrow but do not close the gap with expert human coders on ATDIE-based institutional analysis. The best-performing model (Claude 3.5 Haiku, 69.5% accuracy, $\kappa = 0.482$) achieves “moderate” agreement with human gold standards, compared to “substantial” agreement between human experts ($\kappa = 0.72$). Performance is highly dimension-dependent: Deontic coding is highly amenable to automation (83.0%), while Enforcement remains resistant (38.1%) due to reliance on cross-referential and pragmatic reasoning. Few-Shot prompting consistently outperforms alternatives, and model scale shows a surprisingly flat relationship with accuracy under quantized inference conditions.

Perhaps most significantly for research methodology, locally deployed open-source models achieve 97.2% of cloud model accuracy under the conditions of this benchmark ($p = 0.31$), with substantially higher run-to-run consistency. This finding supports the viability of privacy-preserving, data-sovereign workflows for regulatory text analysis—a critical consideration for researchers working with sensitive governance documents.

The ATDIE-Bench v1.0 corpus and evaluation framework introduced here provide a standardized infrastructure for tracking LLM progress on institutional coding tasks. As models continue to improve, periodic re-evaluation against this and expanded benchmarks will be essential for understanding whether LLMs are approaching the threshold of autonomous institutional coding or whether certain dimensions of regulatory analysis—particularly those requiring legal pragmatic reasoning—represent a more fundamental challenge for current architectures.

Future work should expand the evaluation corpus to include multi-lingual regulatory texts, explore hybrid human–LLM coding workflows, investigate the effect of retrieval-augmented generation on enforcement recognition, and develop dimension-specific fine-tuning strategies to address the identified performance bottlenecks.

We argue that future research on automated analysis of AI governance and regulatory texts should report performance against ATDIE-Bench or comparable grammar-based benchmarks in order to ensure methodological transparency and comparability across studies. In this sense, ATDIE-Bench is intended not merely as an evaluation dataset, but as a shared reference point for cumulative research in algorithmic governance.

Editor’s Note. This article is published as part of the inaugural issue of the *Journal of Algorithmic Governance & Policy (JAGP)*, Volume 1, Issue 1. The ATDIE-Bench v1.0 corpus, gold-standard codings, and evaluation scripts are publicly available at <https://github.com/CPCL-Journals/ATDIE-Benchmark>. JAGP welcomes submissions that advance the empirical and theoretical foundations of algorithmic governance through rigorous, reproducible methodologies.

References

- Basurto, X., Kingsley, G., McQueen, K., Smith, M., & Weible, C. M. (2010). A systematic approach to institutional analysis: Applying Crawford and Ostrom’s grammar. *Political Research Quarterly*, 63(3), 523–537.
- Bommarito, M. J., II, & Katz, D. M. (2022). GPT takes the bar exam. *arXiv preprint arXiv:2212.14402*.
- Bradford, A. (2020). *The Brussels Effect: How the European Union Rules the World*. Oxford University Press.
- Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33, 1877–1901.
- Chalkidis, I., Fergadiotis, M., Malakasiotis, P., Aletras, N., & Androutsopoulos, I. (2020). LEGAL-BERT: The mupets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020* (pp. 2898–2904).
- Cohen, J. (1988). *Statistical Power Analysis for the Behavioral Sciences* (2nd ed.). Lawrence Erlbaum Associates.

- Crawford, S. E. S., & Ostrom, E. (1995). A grammar of institutions. *American Political Science Review*, 89(3), 582–600.
- Davison, A. C., & Hinkley, D. V. (1997). *Bootstrap Methods and Their Application*. Cambridge University Press.
- Efron, B., & Tibshirani, R. J. (1993). *An Introduction to the Bootstrap*. Chapman & Hall/CRC.
- European Parliament and Council. (2024). Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). *Official Journal of the European Union*, L series, 2024/1689.
- Floridi, L., & Cowls, J. (2019). A unified framework of five principles for AI in society. *Harvard Data Science Review*, 1(1).
- Frantz, C. K., & Siddiki, S. N. (2021). Institutional grammar 2.0: A specification for encoding and analyzing institutional design. *Public Administration*, 99(1), 182–199.
- Frantz, C. K., & Siddiki, S. N. (2022). *Institutional Grammar: Foundations and Applications for Institutional Analysis*. Palgrave Macmillan.
- Gilardi, F., Alizadeh, M., & Kubli, M. (2023). ChatGPT outperforms crowd-workers for text-annotation tasks. *Proceedings of the National Academy of Sciences*, 120(30), e2305016120.
- Guha, N., Nyarko, J., Ho, D. E., Ré, C., Chilton, A., Nanamori, A., ... & Henderson, P. (2023). LegalBench: A collaboratively built benchmark for measuring legal reasoning in large language models. *arXiv preprint arXiv:2308.11462*.
- Hagendorff, T. (2020). The ethics of AI ethics: An evaluation of guidelines. *Minds and Machines*, 30(1), 99–120.
- He, X., Lin, Z., Gong, Y., Jin, A., Zhang, H., Lin, C., ... & Duan, N. (2023). AnnoLLM: Making large language models to be better crowdsourced annotators. *arXiv preprint arXiv:2303.16854*.
- Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., Casas, D. de las, ... & Sayed, W. E. (2023). Mistral 7B. *arXiv preprint arXiv:2310.06825*.
- Jobin, A., Ienca, M., & Vayena, E. (2019). The global landscape of AI ethics guidelines. *Nature Machine Intelligence*, 1(9), 389–399.
- Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., & Iwasawa, Y. (2022). Large language models are zero-shot reasoners. *Advances in Neural Information Processing Systems*, 35, 22199–22213.
- Krippendorff, K. (2018). *Content Analysis: An Introduction to Its Methodology* (4th ed.). Sage Publications.
- Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33(1), 159–174.
- Li, C.-K. (2025). A policy grammar for algorithmic governance: Responsibility, boundaries, and instruments. *Journal of Algorithmic Governance & Policy*, 1(1), 1–XX.
- Liang, P., Bommasani, R., Lee, T., Tsipras, D., Soylu, D., Yasunaga, M., ... & Koreeda, Y. (2023). Holistic evaluation of language models. *Transactions on Machine Learning Research*.
- OpenAI. (2024). GPT-4o system card. OpenAI Technical Report.
- Ostrom, E. (2005). *Understanding Institutional Diversity*. Princeton University Press.
- Ouyang, S., Zhang, J. M., Harman, M., & Wang, M. (2023). LLM is like a box of chocolates: The non-determinism of ChatGPT in code generation. *arXiv preprint arXiv:2308.02828*.
- Pangakis, N., Wolken, S., & Fasching, N. (2023). Automated annotation with generative AI requires validation. *arXiv preprint arXiv:2306.00176*.
- Savelka, J., Ashley, K. D., Gray, M. A., Westermann, H., & Xu, H. (2023). Can GPT-4 support analysis of textual data in tasks requiring highly specialized domain expertise? In *Proceedings of the 19th International Conference on Artificial Intelligence and Law* (pp. 196–205). ACM.
- Siddiki, S. N., Weible, C. M., Basurto, X., & Calanni, J. (2011). Dissecting policy designs: An application of the institutional grammar tool. *Policy Studies Journal*, 39(1), 79–103.
- Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., ... & Scialom, T. (2023). LLaMA 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.
- Turpin, M., Michael, J., Perez, E., & Bowman, S. R. (2023). Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36.
- UK Government. (2023). *A Pro-Innovation Approach to AI Regulation*. Department for Science, Innovation and Technology. Cm 815.
- Wang, X., Wei, J., Schuurmans, D., Le, Q., Chi, E., Narang, S., ... & Zhou, D. (2023). Self-consistency improves chain of thought reasoning in language models. In *Proceedings of ICLR 2023*.

- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., ... & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35, 24824–24837.
- The White House. (2023). Executive Order 14110: Safe, secure, and trustworthy development and use of artificial intelligence. *Federal Register*, 88(210), 75191–75226.
- Ziems, C., Held, W., Shaikh, O., Chen, J., Zhang, Z., & Yang, D. (2024). Can large language models transform computational social science? *Computational Linguistics*, 50(1), 237–291.

Appendix A: Full Benchmark Results — All 57 Model–Prompt Conditions

Table 9 reports accuracy, Cohen’s κ , weighted κ , API success rate, and per-dimension accuracy for all 57 model–prompt conditions (19 models $\times$ 3 strategies). Each cell represents the mean across 5 independent replications (temperature = 0.0). Models are grouped by deployment mode; the best value in each column is shown in **bold**.

Table 9: Complete ATDIE-Bench v1.0 Results (19 Models $\times$ 3 Prompt Strategies)

Model		Deploy	Prompt	Acc.	κ	κ_w	SR	A	T	D	I	E
<i>Cloud API Models</i>												
Claude	3.5	Cloud	CoT	60.1	.344	.415	100	76.7	42.9	85.7	61.9	33.3
Haiku												
Claude	3.5	Cloud	FS	68.8	.471	.533	100	77.6	61.9	82.4	64.3	57.6
Haiku												
Claude	3.5	Cloud	ZS	63.0	.409	.473	51	87.0	32.4	91.7	63.0	40.7
Haiku												
Claude 3.5 Son-		Cloud	CoT	58.1	.349	.373	100	71.4	47.6	81.0	57.1	33.3
net												
Claude 3.5 Son-		Cloud	FS	63.3	.410	.411	100	71.4	61.9	81.0	57.1	45.2
net												
Claude 3.5 Son-		Cloud	ZS	59.5	.374	.383	100	64.3	59.5	81.0	59.5	33.3
net												
DeepSeek-V3		Cloud	CoT	59.3	.322	.367	100	63.8	55.7	81.0	62.9	33.3
DeepSeek-V3		Cloud	FS	59.8	.343	.360	100	64.8	60.5	78.6	59.5	35.7
DeepSeek-V3		Cloud	ZS	59.6	.338	.366	100	61.9	57.6	81.0	64.3	33.3
DeepSeek-R1		Cloud	CoT	56.9	.319	.371	84	56.5	52.0	83.6	58.2	34.5
(Cl.)												
DeepSeek-R1		Cloud	FS	58.9	.358	.404	63	57.1	57.9	83.5	54.9	41.4
(Cl.)												
DeepSeek-R1		Cloud	ZS	57.7	.327	.375	80	56.0	50.6	84.5	61.9	35.7
(Cl.)												
Gemini	2.0	Cloud	CoT	58.8	.343	.362	100	64.3	58.1	81.0	57.1	33.3
Flash												
Gemini	2.0	Cloud	FS	57.0	.325	.341	100	64.3	51.9	81.0	53.3	34.8
Flash												
Gemini	2.0	Cloud	ZS	59.5	.357	.377	100	65.2	57.1	81.0	61.0	33.3
Flash												
Gemini 1.5 Pro		Cloud	CoT	59.3	.361	.412	69	60.0	60.7	84.1	59.3	32.4

Continued on next page...

(continued from previous page)

Model		Deploy	Prompt	Acc.	κ	κ_w	SR	A	T	D	I	E
Gemini 1.5 Pro		Cloud	FS	60.9	.383	.433	86	66.1	59.4	84.4	60.0	34.4
Gemini 1.5 Pro		Cloud	ZS	61.8	.400	.448	78	65.0	56.4	83.4	64.4	39.9
GPT-4o		Cloud	CoT	53.2	.306	.309	100	64.3	28.1	78.6	61.9	33.3
GPT-4o		Cloud	FS	55.9	.332	.337	100	62.4	42.9	81.0	60.0	33.3
GPT-4o		Cloud	ZS	53.4	.303	.317	100	63.3	27.6	81.0	61.9	33.3
GPT-4o-mini		Cloud	CoT	61.0	.342	.382	100	79.5	50.0	81.0	55.7	38.6
GPT-4o-mini		Cloud	FS	60.5	.357	.391	100	64.3	59.5	83.3	54.8	40.5
GPT-4o-mini		Cloud	ZS	61.6	.349	.398	100	83.8	47.6	81.0	56.2	39.5
Qwen-Max		Cloud	CoT	57.4	.346	.361	100	68.9	47.8	80.4	56.9	33.0
Qwen-Max		Cloud	FS	63.7	.421	.432	100	74.3	61.0	81.4	60.0	41.9
Qwen-Max		Cloud	ZS	58.3	.360	.375	100	69.5	48.6	81.9	58.1	33.3
Qwen-Plus		Cloud	CoT	61.5	.370	.395	100	81.0	50.5	84.3	55.2	36.7
Qwen-Plus		Cloud	FS	64.6	.407	.440	100	81.4	59.5	83.3	53.8	44.8
Qwen-Plus		Cloud	ZS	61.0	.371	.385	100	78.1	50.0	83.8	56.2	37.1
Locally Deployed Models (4-bit Q4_K_M, RTX 5880 48GB)												
DeepSeek-R1 8B		Local	CoT	58.1	.360	.375	100	76.2	33.3	83.3	64.3	33.3
DeepSeek-R1 8B		Local	FS	57.6	.361	.349	100	78.6	26.2	85.7	64.3	33.3
DeepSeek-R1 8B		Local	ZS	59.5	.365	.393	100	85.7	33.3	85.7	59.5	33.3
DeepSeek-R1 32B		Local	CoT	53.7	.286	.295	100	73.8	25.7	85.7	50.0	33.3
DeepSeek-R1 32B		Local	FS	59.0	.330	.358	100	69.0	40.5	83.3	61.9	40.5
DeepSeek-R1 32B		Local	ZS	54.7	.304	.306	100	76.2	26.2	85.7	51.9	33.3
DeepSeek-R1 70B		Local	CoT	54.3	.320	.310	100	76.2	28.6	81.0	52.4	33.3
DeepSeek-R1 70B		Local	FS	60.0	.367	.404	100	83.3	40.5	81.0	59.5	35.7
DeepSeek-R1 70B		Local	ZS	54.2	.321	.302	100	71.0	28.6	78.6	59.5	33.3
Gemma2 27B		Local	CoT	57.5	.351	.378	100	85.7	26.2	88.1	54.3	33.3
Gemma2 27B		Local	FS	58.9	.348	.376	100	73.8	33.3	87.6	63.8	35.7
Gemma2 27B		Local	ZS	58.2	.362	.383	100	88.1	28.6	88.1	52.9	33.3
LLaMA 3.3 70B		Local	CoT	54.9	.315	.323	100	69.0	40.5	79.0	52.4	33.3
LLaMA 3.3 70B		Local	FS	58.7	.355	.361	100	64.3	52.9	85.7	57.1	33.3
LLaMA 3.3 70B		Local	ZS	53.4	.295	.316	100	69.5	35.7	78.6	50.0	33.3
Mistral 24B	Small	Local	CoT	54.8	.317	.340	100	64.3	45.2	83.3	47.6	33.3
Mistral 24B	Small	Local	FS	55.6	.289	.331	100	73.3	40.5	83.3	47.6	33.3
Mistral 24B	Small	Local	ZS	55.7	.338	.350	100	64.3	45.2	81.0	54.8	33.3

Continued on next page...

(continued from previous page)

Model	Deploy	Prompt	Acc.	κ	κ_w	SR	A	T	D	I	E
Phi-4 14B	Local	CoT	56.2	.324	.357	100	69.0	42.9	83.3	52.4	33.3
Phi-4 14B	Local	FS	60.7	.370	.402	100	61.0	66.2	83.3	57.6	35.2
Phi-4 14B	Local	ZS	54.3	.297	.328	100	61.9	40.5	83.3	52.4	33.3
Qwen3 8B	Local	CoT	60.8	.347	.396	100	80.5	49.5	85.7	54.8	33.3
Qwen3 8B	Local	FS	61.5	.384	.407	100	83.3	57.6	78.6	54.8	33.3
Qwen3 8B	Local	ZS	61.2	.348	.400	100	85.2	49.0	83.3	55.2	33.3
Qwen3 32B	Local	CoT	59.1	.356	.387	100	73.8	45.2	83.3	60.0	33.3
Qwen3 32B	Local	FS	67.3	.446	.496	100	81.4	69.0	83.3	61.9	41.0
Qwen3 32B	Local	ZS	59.5	.369	.391	100	76.2	42.9	85.7	59.5	33.3

Note: CoT = Deliberation Instruction (chain-of-thought); FS = Few-Shot; ZS = Zero-Shot. Acc. = micro-average accuracy (%). SR = API success rate (%). A/T/D/I/E = per-dimension accuracy for Actor, Trigger, Deontic, Instrument, Enforcement. All local models run under 4-bit quantization (Q4_K_M) on a single NVIDIA RTX 5880 48GB GPU. Each reported value is the mean of 5 independent replications at temperature 0.0. Best values in each column shown in **bold**. Total experiment: 19 models $\times$ 3 prompts $\times$ 42 statements $\times$ 5 replications = **11,970 API calls**.

Academic Editor *Xin Huang*, *cpcl@cpcl.hk*