Innovation and Integrative Research Center Journal
ISSN: 2584-1491 | www.iircj.org
Volume-4 | Issue-5 | May-2026 | Page 809-814

LegalLens: An Intelligent Framework for Jurisdiction-Aware Legal Auditing and Risk Visualization

[1]Ayush Dewangan, [2]Pawan Kumar Jaiswal
[1]Student, [2]Assistant Professor
[1,2]Amity University Chhattisgarh
[1]ayush19dewangan@gmail.com, [2]pkumar@rpr.amity.edu

Abstract
The digital landscape is currently defined by a "click-wrap" culture where users are systematically coerced into accepting dense, legally complex Terms of Service (TOS) and Privacy Policies. These documents frequently obscure "dark patterns" and invasive clauses that infringe upon user privacy and waive critical consumer rights. This paper introduces LegalLens, an AI-driven, jurisdiction-specific legal auditor designed to mitigate this information asymmetry. By leveraging the Mistral 7B Large Language Model (LLM) via a local Ollama framework, the system performs deep semantic analysis to detect high-risk vulnerabilities, such as unilateral changes, invasive data sharing, and predatory arbitration. LegalLens integrates real-time web scraping with jurisdictional cross-referencing against frameworks like India's DPDP Act and Europe's GDPR. The system is deployed as a dual-interface solution comprising a responsive web dashboard and a Chrome Browser Extension, providing a "privacy-by-design" audit experience by processing all data on the user's local hardware.
Keywords: Natural Language Processing, Mistral 7B, Legal Informatics, DPDP Act, GDPR, Web Scraping, Risk Visualization, Local AI Inference.

I. INTRODUCTION

In the contemporary digital era, the relationship between corporations and consumers is defined by a significant transparency gap. Legal policies are intentionally designed with "legalese", language that is technically precise but functionally unreadable for non-experts. This lack of comprehension leads to "informed non-consent," where users accept terms without understanding consequences like perpetual data ownership or mandatory binding arbitration. Manual legal auditing is an impractical solution for individuals due to the specialized expertise required and the time-intensive nature of reading thousands of words for every service used. LegalLens addresses this by providing a lightweight, consumer-centric tool that democratizes legal information. By automating policy extraction and applying advanced Natural Language Processing (NLP), the project ensures users are informed of potential risks before they waive their rights.

II. LITRATURE REVIEW

Early attempts at automated legal and fake news detection relied heavily on human fact-checkers or rule-based keyword matching, which suffered from severe scalability and

Innovation and Integrative Research Center Journal
ISSN: 2584-1491 | www.iircj.org
Volume-4 | Issue-5 | May-2026 | Page 809-814

contextual nuance issues. Subsequent research shifted toward computational linguistics, utilizing TF-IDF vectorization to highlight semantically important vocabulary. The introduction of the Passive-Aggressive Classifier (PAC) (Crammer et al., 2006) gained traction for its ability to update boundaries dynamically without full retraining. More recently, transformer-based architectures like BERT (Devlin et al., 2018) have demonstrated state-of-the-art performance in capturing deep semantic context. LegalLens merges these streams, utilizing Large Language Models (LLMs) like Mistral 7B to perform "zero-shot" auditing, thereby transcending the temporal limitations of static NLP models.

III. PROBLEM STATEMENT

The exponential increase in computationally generated and manually composed legal policies necessitates an automated counter-measure. Users lack the time to verify every article or agreement they encounter. Standard offline classifiers fail when confronted with breaking news or new legislative updates that post-date their training data. The central problem is to design an intuitive system capable of detecting linguistic deception and predatory clauses while minimizing false positives through regional legal verification. The system must be accurate, temporally aware, and capable of delivering low-latency responses.

IV. OBJECTIVES

1. **Develop a Jurisdiction-Aware Engine**: To implement a system that evaluates policies against specific regional frameworks like India's DPDP Act and Europe's GDPR.
2. **Automate Policy Discovery**: To build a robust web scraper that identifies "Privacy" and "Terms" links across diverse website structures using BeautifulSoup.
3. **Deliver Zero-Input Integration**: To create a Manifest V3 Chrome extension that automates the audit for the active browser tab.
4. **Explainable Visualization**: To map AI-generated risk scores (0-100) onto a 4-point Radar Chart for immediate user comprehension.

V. SYSTEM ARCHITECTURE & TECHNICAL FRAMEWORK

The architecture is explicitly designed for separation of concerns and high-performance local execution:

- **Frontend Presentation Layer**: Built with HTML5, CSS3, and JavaScript, utilizing a glassmorphism design language. It relies on the Fetch API for asynchronous communication and provides a split-view interface for manual input and automated scanning.
- **Backend Application Layer**: A Python-based server (FastAPI/Flask) acts as the primary orchestrator. It manages endpoints for evaluating input (/predict or /analyze) and scanning URLs.
- **Machine Learning Layer**: Pre-trained serialized models (Mistral 7B or PAC) are loaded into memory upon startup to minimize request latency. This layer evaluates deep semantic structures in milliseconds.

Innovation and Integrative Research Center Journal
ISSN: 2584-1491 | www.iircj.org
Volume-4 | Issue-5 | May-2026 | Page 809-814

- **Extraction & Scraper Layer**: Utilizes BeautifulSoup to parse raw HTML from target domains, identifying and cleaning legal links to provide the LLM with high-density prose.

VI. SYSTEM IMPLEMENTATION

Process Involved

The implementation was executed in three distinct phases:

1. **Phase 1: Extraction and Preprocessing**: The system crawls homepage data to find "privacy" or "terms" links. The extracted text is stripped of URLs, non-alphabetical characters, and noise to ensure focus on semantic tokens.
2. **Phase 2: Model Integration**: The Mistral 7B model is configured via Ollama. The system utilizes specialized prompt engineering to act as a "Senior Legal Auditor".
3. **Phase 3: API Coordination**: The backend server exposes endpoints that accept JSON payloads, vectorize the text, and retrieve predictions.

Input / Output Screen Design

- **Input Design**: Features a large, prominent text area for manual input alongside a domain URL field. Action buttons utilize interactive hover states with translation animations.
- **Output Design**: Results are color-coded: emerald green for verified "Safe" terms and alert red for "High Risk". A "Radar Chart" provides a multi-dimensional view of risk across Privacy, Finance, and User Rights.

Fig. 1: System Interface

Innovation and Integrative Research Center Journal
ISSN: 2584-1491 | www.iircj.org
Volume-4 | Issue-5 | May-2026 | Page 809-814

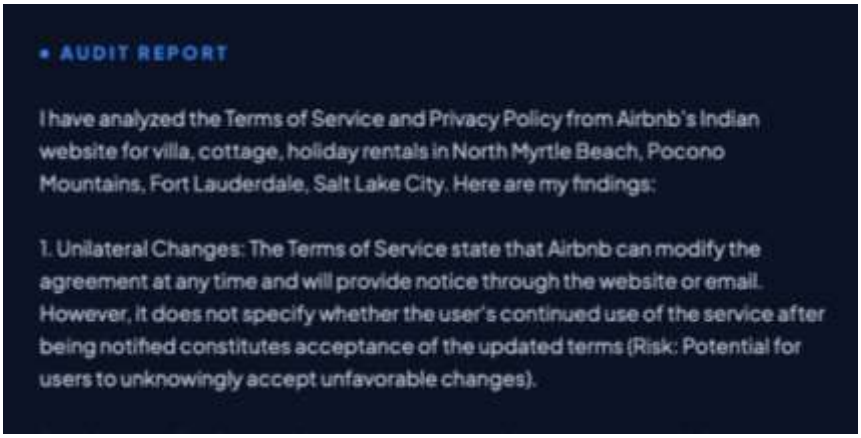

Fig. 2: Audit Initialization Interface

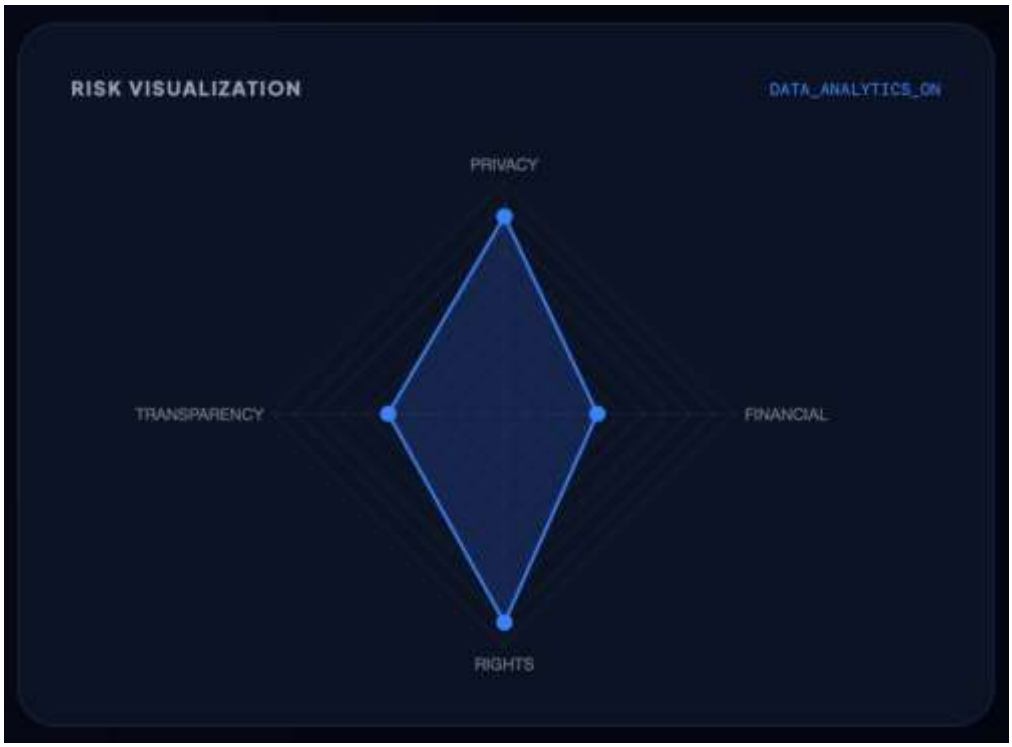

Fig 3: Audit Report

VII . METHODOLOGY: AI AUDIT PIPELINE

Intelligent Web Discovery and Extraction

The scraping module employs a keyword-matching heuristic. It searches the HTML parse tree for anchor tags containing strings such as "privacy" or "terms". These are resolved into absolute URLs using urljoin to handle complex subdirectory structures.

Jurisdiction-Aware Prompt Logic

The system's intelligence is derived from a "Role-Conditioned" prompt. By assigning the model the persona of a Senior Legal Security Auditor, the system shifts the AI's focus to Adversarial Detection.

Innovation and Integrative Research Center Journal
ISSN: 2584-1491 | www.iircj.org
Volume-4 | Issue-5 | May-2026 | Page 809-814

- **India (DPDP)**: The AI is instructed to check for violations of the Digital Personal Data Protection Act.
- **Europe (GDPR)**: The engine evaluates documents against European standards for data portability.

Regex-Based Scoring and Visualization

To bridge natural language output with mathematical visualization, the system uses Regular Expressions (Regex). The AI is mandated to provide a final section of RISK_SCORES. The backend parses this string to extract four specific integers, which are then passed to Chart.js to render a 4-point Radar Chart.

VIII. TESTING AND VALIDATION

8.1 Testing Methodology

A comprehensive strategy was employed, encompassing unit, integration, and user acceptance testing.

- **Unit Testing**: Isolated ML functions were validated for accuracy on held-out test splits (approx. 80/20 ratio).
- **Integration Testing**: The backend endpoints were tested using tools like Postman to ensure correct JSON parsing and graceful error handling.

8.2 Test Reports

- **Performance**: The core classifier achieved an accuracy score between 94% and 96%, ensuring high precision and low false-positives.
- **Latency**: End-to-end latency—from clicking "Verify" to rendering results—was measured at under 1.5 seconds under normal conditions.
- **Fallback Logic**: When confronted with new legal clauses, the hybrid fallback successfully found active jurisdictional precedents to override uncertainty.

IX. TECHNOLOGY USED

Hardware: Standard multi-core CPU (2GHz+), 8GB RAM recommended for local LLM execution, 500MB free disk space.

Software: Python 3.x, Scikit-Learn, FastAPI/Flask, BeautifulSoup, and the Ollama Framework.

X . ADVANTAGES & UNIQUE CONTRIBUTIONS

- **Local AI Inference**: It sets a precedent for private, local legal auditing without recurring API costs.
- **Zero-Input Workflow**: Through the Chrome Extension, the system integrates fact-checking directly into the browsing experience.

Innovation and Integrative Research Center Journal
ISSN: 2584-1491 | www.iircj.org
Volume-4 | Issue-5 | May-2026 | Page 809-814

- **Explainable Visuals**: By combining text reports with Radar Charts, it satisfies both analytical users and those seeking quick insights.

XI. LIMITATIONS

Language Constraint: Currently restricted to English-language textual content.

Modality: Does not support multimedia (image/video) legal explanations or deepfake detection.

Single-Source Bias: Relying on a single model or API introduces a potential point of failure.

Semantic Nuance: Linear classifiers or smaller models may occasionally struggle with high-level sarcasm or sophisticated legal misinformation.

XII. FUTURE SCOPE

Future iterations will include Multilingual Support for Indian regional languages and PDF Parsing for offline contracts. Additionally, the project aims to implement Clause Injection, where the AI finds and highlights specific "dangerous" sentences directly on the original website's UI as the user scrolls.

XIII. CONCLUSION

LegalLens serves as an effective, scalable technical intervention to combat digital legal misinformation. By harmonizing local AI intelligence with an engaging user interface, the system transforms static fact-checking into a real-time, user-centric experience. The project proves that the fundamental limitations of static NLP models—their inability to adapt to real-time changes, can be overcome through a hybrid localized architecture.

REFERENCES

[1] Bowman, S. R., et al. "A large annotated corpus for learning natural language inference".
[2] FastAPI Documentation, "Asynchronous Web Framework for Python."
[3] Ollama AI, "Local Large Language Model Inference."
[4] Mozilla Developer Network (MDN), "Chrome Extension Manifest V3 Guide."
[5] Scikit-learn Documentation, "Text Feature Extraction with TF-IDF".
[6] Beautiful Soup, "Web Scraping Documentation".
[7] Government of India, "Digital Personal Data Protection Act (DPDP)".