Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

JurisAI: AI-based Legal Assistance System

Mrunaleeni Bhilare
Department of Data Science
Usha Mittal Institute of Technology
Mumbai, India

Anushka Chavan
Department of Data Science
Usha Mittal Institute of Technology
Mumbai, India

Prerana Kahane
Department of Data Science
Usha Mittal Institute of Technology
Mumbai, India

Prof. Mohan Bonde
Department of Data Science
Usha Mittal Institute of Technology
Mumbai, India

***Abstract*—JurisAI is an artificial intelligence (AI)-powered legal intelligence platform designed to simplify access to Indian legal information. It integrates structured Bharatiya Nyaya Sanhita (BNS), Bharatiya Nagarik Suraksha Sanhita (BNSS), and Bharatiya Sakshya Adhiniyam (BSA) datasets with a legal chatbot that combines fuzzy matching and AI-based response generation. The platform provides document analysis, legal decision-tree guidance, legal document templates, a Court Fee Explorer, and a court locator built on geospatial APIs. The system was evaluated across its major modules to assess its effectiveness in providing accessible and structured legal assistance, achieving an overall test pass rate of 92.1 percent across chatbot, Court Fee Guide and Court Locator.**

Keywords–AI Chatbot, Fuzzy Matching, BNS, BNSS, BSA, Geospatial Services

I. Introduction

Access to legal information is important for individuals seeking to understand their rights, responsibilities, and available legal remedies. However, legal information is often presented through complex statutory language, lengthy documents, and multiple sources, making it difficult for individuals without legal expertise to locate and understand relevant information. The increasing use of artificial intelligence and language-processing technologies provides an opportunity to simplify interaction with structured legal information.

In India, recent changes in the criminal law framework have consolidated important provisions under the Bharatiya Nyaya Sanhita (BNS), Bharatiya Nagarik Suraksha Sanhita (BNSS), and Bharatiya Sakshya Adhiniyam (BSA). Accessing and interpreting these sources requires navigating large volumes of statutory content, creating a need for systems that organize legal information and present it in an accessible and structured manner.

Existing digital legal resources often focus on specific functions such as legal information retrieval, document access, or case-law search. Users may therefore need multiple platforms for different legal tasks. Legal documents also contain clauses, obligations, rights, and conditions that may be difficult to understand without legal knowledge. These challenges motivate the development of an integrated platform combining multiple legal assistance functionalities.

JurisAI is an artificial intelligence (AI)-powered legal intelligence platform designed to provide structured legal assistance through a unified interface. It integrates BNS, BNSS, and BSA datasets with a legal chatbot using fuzzy matching and AI-based response generation. The system also provides document analysis, legal decision-tree guidance, legal document templates, a Court Fee Explorer, and a court and legal-aid locator.

The document analysis component provides summaries, important clauses, rights and obligations, potential red flags, and suggested actions from uploaded legal documents. The template module provides structured legal documents, while the Court Fee Explorer organizes applicable court-fee information. The court and legal-aid locator uses geospatial services to identify relevant facilities based on location.

The primary objective of JurisAI is to consolidate these legal assistance functions into a single platform and simplify access to legal information. Rather than replacing professional legal advice, the system serves as an informational and assistance-oriented tool for understanding legal information and identifying relevant resources.

The major objectives of the proposed system are:

1) To develop an accessible platform for interacting with structured Indian legal information.
2) To integrate BNS, BNSS, and BSA datasets into a unified legal information system.
3) To implement an intelligent chatbot using fuzzy matching and AI-based response generation.
4) To provide automated analysis and summarization of uploaded legal documents.
5) To provide legal decision-tree guidance and structured legal document templates.
6) To provide court fee information through the Court Fee Explorer.
7) To help users locate nearby courts and legal-aid facilities through geospatial services.
8) To evaluate the effectiveness of the system across its major functional modules.

II. Related Works

Several studies have explored the application of Artificial Intelligence (AI), Natural Language Processing (NLP), Machine Learning (ML), and Large Language Models (LLMs) in the legal domain.

Yalla and Prasad presented a literature survey on the role of Artificial Intelligence in transforming the Indian judicial system. The study discusses the application of NLP, Machine Learning, and Large Language Models for legal document analysis, judgment summarization, and legal information retrieval. Models such as LegalBERT and InLegalLLaMA are discussed for legal text processing. The authors also highlight challenges including bias, limited datasets, and lack

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

of transparency, emphasizing that AI should support legal professionals responsibly.[1]

Lai surveyed the applications of Large Language Models in the legal domain, including legal document analysis, case summarization, and legal research. Models such as BERT and GPT can process large volumes of legal data and improve efficiency. However, limitations related to data quality, bias, and transparency remain. The study emphasizes that LLMs should assist rather than replace legal professionals.[2]

Dehghani examined the use of Large Language Models in legal systems. The study discusses models such as GPT, LLaMA, BLOOM, and LaMDA for applications including legal research, document drafting, contract analysis, and case-law analysis. It also discusses AI-based legal assistants and their ability to process large volumes of legal information. Key challenges identified include bias, privacy risks, and hallucinated responses.[3]

Singla and Gupta discussed the impact of technology on the legal profession, particularly the role of AI, automation, and data analytics in legal research, document management, and case analysis. The study highlights Indian initiatives such as e-Courts, the National Judicial Data Grid, and virtual hearings, which have improved accessibility and efficiency. However, cybersecurity risks and resistance to technology adoption remain important challenges.[4]

Ariaei and Demartini presented a survey of Natural Language Processing for the legal domain, covering tasks, datasets, models, and challenges. The study examines applications such as legal question answering, document summarization, and text classification. Machine learning and transformer-based models can convert complex legal documents into structured information. The authors identify complex legal language, limited datasets, and model bias as major challenges.[5]

III. Problem Statement

Individuals without legal expertise often face difficulty in locating, interpreting, and acting upon relevant legal information due to the complexity of statutory language and the fragmentation of legal resources across multiple platforms. Legal documents such as agreements, notices, and contracts frequently contain clauses, obligations, and conditions that are not easily understood without professional guidance, while access to structured information on court procedures, applicable fees, and nearby legal facilities remains scattered and difficult to consolidate. There is therefore a need for an integrated platform that combines statutory legal information, AI-assisted query resolution, document analysis, decision-support guidance, and location-based legal services within a single accessible interface.

IV. Proposed System

JurisAI is an integrated AI-powered legal intelligence platform designed to provide users with accessible and structured Indian legal information through a single interface. The system combines legal datasets, AI-based processing, document analysis, decision-tree guidance, legal document templates, court fee information, and location-based legal services.

The system follows a modular architecture consisting of a frontend, backend, legal datasets, AI processing components, and external APIs, as illustrated in Fig. 1. The frontend is developed using Next.js, TypeScript, Tailwind CSS, and ShadCN UI, providing the user interface for accessing different JurisAI modules. The backend manages application logic, dataset processing, API communication, and AI-based operations.

The legal information layer contains structured BNS, BNSS, and BSA datasets. The chatbot uses RapidFuzz-based fuzzy matching to identify relevant legal information from user queries, followed by AI-based response generation. The document analysis module processes uploaded documents and uses the Groq AI API for generating structured insights. The court locator uses Nominatim for geocoding and the Overpass API with OpenStreetMap data to identify nearby courts and legal-aid facilities. The major components and their interactions are shown in Fig. 1.

JurisAI uses a dual-backend architecture: Next.js serves as the frontend and primary backend, while Flask runs as a secondary Python backend. Next.js proxies chatbot requests to Flask, where RapidFuzz performs fuzzy retrieval and Pandas manages the legal datasets. For document analysis, Next.js extracts digital PDF text and calls Groq directly. The court locator, court-fee guide, decision tree, templates, and document-summary generation remain handled by Next.js.

V. Methodology

The methodology of JurisAI consists of structured legal data preparation, query processing, rule-based decision support, document analysis, and integration of legal reference and geospatial services. Each module follows an independent processing approach while being integrated through the common Next.js-based application.

A. Data Preparation and Cleaning

The BNS, BNSS, and BSA datasets are maintained as separate CSV files containing structured information about legal sections, including act name, chapter, section number, section title, and statutory description. The datasets are loaded using Pandas with string-based data types to prevent unwanted type conversion during preprocessing. Column names are normalized and automatically mapped to a common structure to handle variations between datasets. The cleaned records are converted into structured legal-section objects and loaded into memory for efficient retrieval during application execution. The resulting corpus contains **1,063 legal sections** across BNS, BNSS, and BSA and serves as the primary legal information source for the chatbot.

B. Legal Chatbot Methodology

User queries are first validated for length and potentially malicious prompt-injection patterns before further processing. The query is preprocessed to identify meaningful legal

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

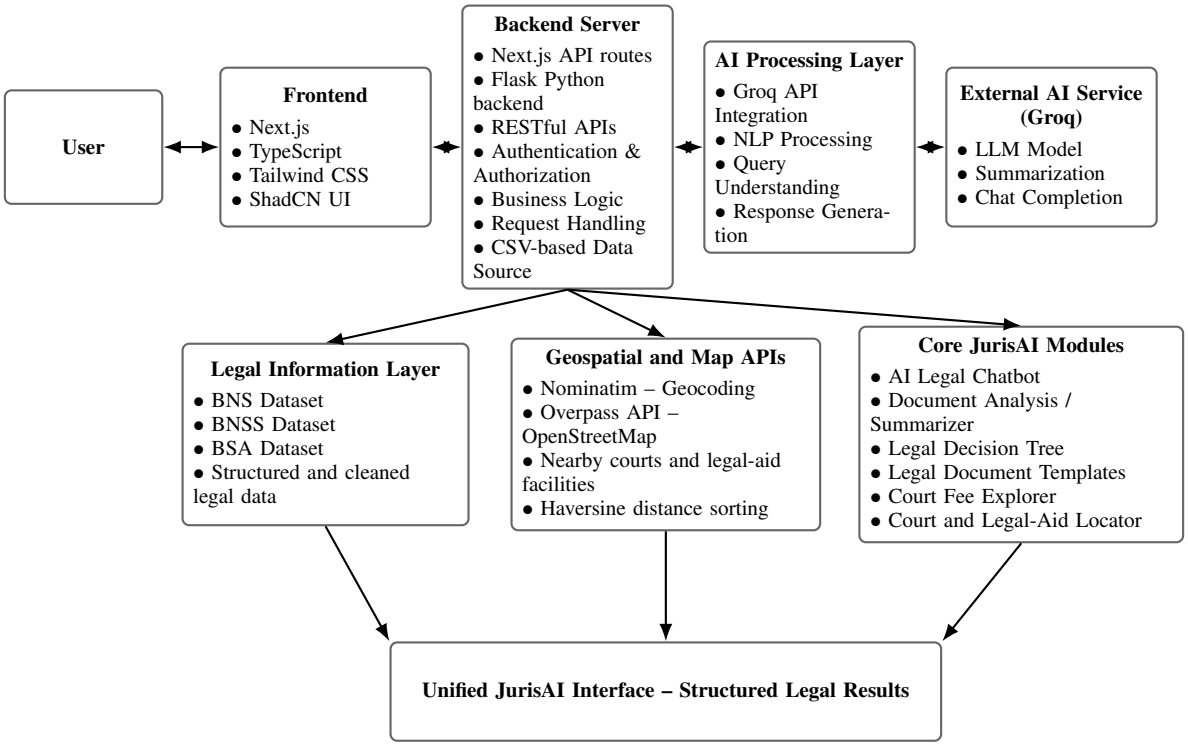


Fig. 1. JurisAI system architecture showing the frontend, backend, legal information layer, AI processing layer, core modules, and external service integrations.

keywords, while explicitly mentioned Acts such as BNS, BNSS, and BSA are separately identified. RapidFuzz-based multi-signal fuzzy matching compares the query with section titles, descriptions, chapters, and section numbers to determine relevant provisions. An alias and synonym layer maps commonly used legal terms to corresponding statutory sections, improving retrieval when user terminology differs from statutory wording. The retrieved legal provisions are provided as context to the Groq-hosted language model, which generates a plain-language response, cites the relevant Act and section, and detects the query language to provide responses in English, Hindi, or Marathi.

C. Document Analysis Methodology

The module accepts PDF, DOCX, DOC, and TXT files with a maximum file size of 5 MB. Text is extracted using `pdf-parse` for PDF files, `mammoth` for Word documents, and UTF-8 decoding for text files. Extracted text is validated and truncated to a maximum of 15,000 characters before AI processing. The processed document text is submitted to the Groq API for structured analysis, including document type, summary, key clauses, rights and obligations, potential red flags, and recommended actions. Documents are processed temporarily within the request and are not persistently stored in a database or file system.

Digital PDFs are processed entirely by the Next.js `/api/document-analysis` route, which extracts text using `pdf-parse` and sends it directly to Groq.

D. Interactive Legal Decision Tree Methodology

The module uses a TypeScript-based `DecisionNode` structure containing questions, explanations, options, navigation identifiers, and terminal outcomes. Users first select one of three implemented scenarios: Tenant Rights, Employment Disputes, or Consumer Rights. Each response determines the next node through client-side React state logic, creating a deterministic branching pathway. Terminal nodes provide pre-authored legal outcomes containing relevant legal provisions, possible legal positions, recommended actions, and applicable limitations where specified. Navigation history is maintained using React state to support backward navigation, while the Start Over function resets the decision process.

E. Court Fee Guide Methodology

Court fee information is manually compiled from the Bombay Court Fees Act, 1959, the Karnataka Court Fees and Suits Valuation Act, 1958, and the Consumer Protection (Consumer Disputes Redressal Commissions) Rules, 2020. The compiled information is stored as a statically typed TypeScript dataset containing **53 fee rows** across Maharashtra, Karnataka, and national consumer courts. Fee structures are classified into fixed, exempt, ad valorem slab, ad valorem formula, and fixed-or-slab categories. Users select a jurisdiction and case type through dependent dropdowns, after which the system retrieves the corresponding statutory reference, notes, source, and verification date. The module functions as an informational reference rather than an automatic calculator and directs users to verify the current fee with the relevant court registry.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

F. Court and Legal-Aid Locator Methodology

The module obtains the user's location either through browser geolocation or by converting a user-entered city name into coordinates using Nominatim. The resulting latitude and longitude are supplied to the Overpass API to retrieve nearby courts from OpenStreetMap data. Retrieved geographic elements are filtered, classified into court types, and converted into readable addresses. The Haversine formula is used to calculate the distance between the user's location and each identified court. Results are sorted by distance and can be further filtered by court name, address, or court type on the client side.

1) Haversine Formula: The Haversine formula used in the Court Locator to calculate the great-circle distance between the user's location and a court is:

$$d = 2R \cdot \arctan 2\left(\sqrt{a}, \sqrt{1-a}\right) \tag{1}$$

where,

$$a = \sin^2\left(\frac{\Delta\phi}{2}\right) + \cos(\phi_1)\cos(\phi_2)\sin^2\left(\frac{\Delta\lambda}{2}\right) \tag{2}$$

and:

- d = distance between the two locations
- R = Earth's radius, taken as 6,371 km
- ϕ_1, ϕ_2 = latitudes of the two locations
- λ_1, λ_2 = longitudes of the two locations
- $\Delta\phi$ = difference between the latitudes
- $\Delta\lambda$ = difference between the longitudes

The calculated distances are then used to sort the retrieved courts so that the nearest court appears first.

VI. System Modules

JurisAI consists of multiple modules that provide different forms of legal information and assistance through a unified web platform.

A. Legal Chatbot

The Legal Chatbot provides conversational access to Indian legal information using the BNS, BNSS, and BSA datasets. It processes user queries through security validation, keyword preprocessing, fuzzy matching, and alias-based legal term resolution to identify relevant statutory provisions. The retrieved provisions are then provided as context to the language model for generating a concise, plain-language response that cites the relevant Act and section. The chatbot also supports **multilingual interaction in English, Hindi, and Marathi**, allowing users to ask questions in these languages and receive responses in the corresponding language. The workflow is as shown in Fig. 2.

B. Document Analysis

The Document Analysis module allows users to upload legal documents and obtain an automated structured analysis. It provides information including document type, summary, key clauses, rights and obligations, potential red flags, and recommended actions. The workflow is as shown in Fig. 3.

C. Interactive Legal Decision Tree

The Interactive Legal Decision Tree provides deterministic, rule-based guidance for selected legal situations. It currently covers Tenant Rights, Employment Disputes, and Consumer Rights and guides users through a sequence of questions toward a relevant legal outcome.

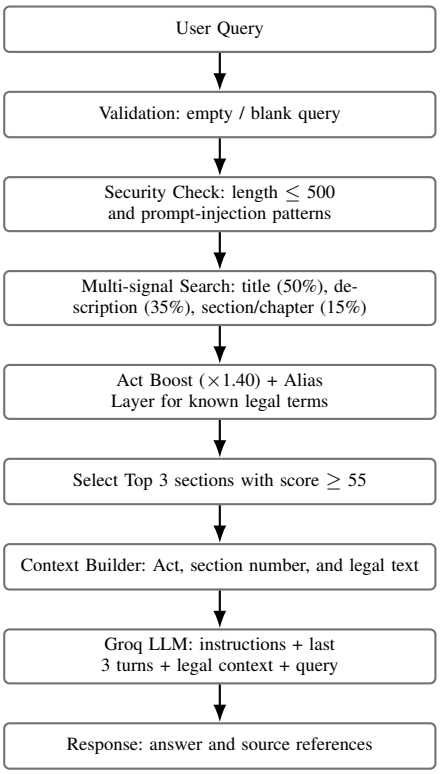


Fig. 2. AI Legal Chatbot workflow.

D. Legal Document Templates

The Legal Document Templates module provides structured templates for commonly required legal documents. It is intended to help users understand and prepare documents using an organized format while providing a starting point for legal documentation.

E. Court Fee Guide

The Court Fee Guide provides statutory court-fee information for selected case types across Maharashtra, Karnataka, and national consumer courts. Instead of automatically calculating fees, it presents the applicable legal source, fee structure, notes, and verification information.

F. Court and Legal-Aid Locator

The Court and Legal-Aid Locator helps users identify nearby courts and legal facilities using geographic information. It combines location acquisition, OpenStreetMap data, geocoding, and Haversine-based distance calculation to present relevant locations sorted by proximity.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

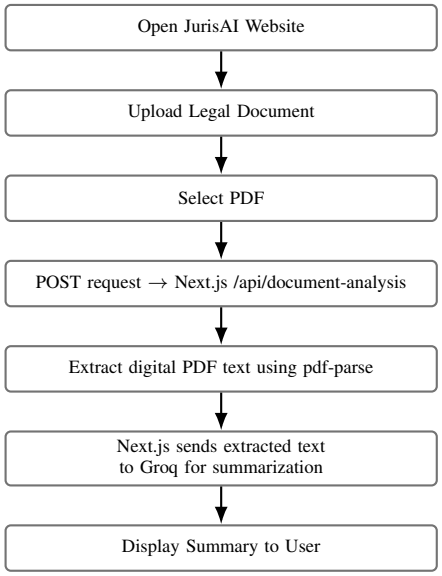


Fig. 3. PDF Document Summarizer workflow.

VII. Results and Evaluation

The JurisAI system was evaluated across its major components to measure retrieval accuracy, hallucination behavior, document summarization quality, functional correctness, security, and system performance. The evaluation was performed using a combination of automated test scripts, live end-to-end testing, human reference summaries, and manual black-box testing.

A. Evaluation Methodology

The evaluation was divided into four major modules: (1) AI Legal Chatbot retrieval evaluation, (2) hallucination testing, (3) document summarization evaluation, and (4) overall system testing. Each module used a separate test set and evaluation criteria appropriate to its functionality.

B. AI Legal Chatbot Retrieval Evaluation

The retrieval engine of the AI Legal Chatbot was evaluated to determine whether the fuzzy search mechanism could retrieve the correct legal section for a user's query. A ground-truth dataset containing 40 queries was prepared. The queries were divided into five categories: 20 direct queries containing the Act name, 5 natural-language queries without the Act name, 5 alias-dependent queries using legal synonyms, 5 typo/fuzzy queries, and 5 out-of-scope queries.

Among the 40 queries, 35 were in-scope queries and 5 were out-of-scope queries. A prediction was considered a true positive when the correct legal section was retrieved at rank 1. If the correct section appeared within the top three results but not at rank 1, it was counted as a TP@3 case. An in-scope query for which the correct result was not found within the top three was treated as a false negative. For out-of-scope queries, results below the defined similarity threshold were treated as true negatives, while results above the threshold were treated as false positives.

The evaluation was performed using a custom Python script, `evaluate_accuracy.py`, which sent live queries to the actual search engine rather than evaluating a static set of predicted outputs. The following metrics were calculated: Top-1 accuracy, Top-3 accuracy, per-Act precision, recall and F1-score, macro-averaged precision, recall and F1-score, and true negative rate (TNR).

TABLE I
Legal Chatbot Retrieval Evaluation Results

Evaluation Metric	Result
Top-1 Retrieval Accuracy	82.9% (29/35)
Top-3 Retrieval Accuracy	91.4% (32/35)
True Negative Rate	80% (4/5)
Macro Precision	0.961
Macro Recall	0.776
Macro F1 Score	0.850
BNS F1 Score	0.882
BNSS F1 Score	0.917
BSA F1 Score	0.750

The retrieval engine achieved a Top-1 accuracy of 82.9%, correctly retrieving the required legal section as the first result for 29 out of 35 in-scope queries. The Top-3 accuracy increased to 91.4%, indicating that the correct section was present within the first three results for 32 out of 35 queries. The true negative rate was 80%, with 4 out of 5 out-of-scope queries correctly remaining below the retrieval threshold.

The macro-averaged precision was 0.961, while the macro recall was 0.776, resulting in a macro F1-score of 0.850. Among the three major legal acts, BNSS achieved the highest F1-score of 0.917, followed by BNS with 0.882 and BSA with 0.750. The lower BSA score was mainly associated with limitations in the available section-title information used by the retrieval engine.

C. Hallucination Evaluation

Hallucination testing was performed to determine whether the chatbot generated legal claims that were unsupported by the retrieved context. Twelve test cases were designed for this evaluation. Six cases were executed completely through the live pipeline, including search, context building, LLM generation, and final response. The remaining six cases were affected by Groq rate limits and were therefore evaluated through analysis of the retrieved context.

The responses were classified into four categories: Faithful, Correct Refusal, Minor Unsupported Elaboration, and Hallucination. A response was classified as Faithful when its claims could be traced to the retrieved legal context. A Correct Refusal represented an appropriate refusal when the required information was not available. A Minor Unsupported Elaboration represented a small unsupported addition that did not constitute a major factual hallucination.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

TABLE II
HALLUCINATION EVALUATION RESULTS

Evaluation Category	Result
Faithful Responses	3/6
Correct Refusals	2/6
Minor Unsupported Elaboration	1/6
Hallucinations	0/6
Hallucination Rate	0.0%

Among the six live end-to-end cases, three responses were completely faithful to the retrieved context, two were correct refusals, and one contained minor unsupported elaboration. No major hallucination was observed, resulting in a measured hallucination rate of 0.0% for the six live cases. Ten prompt injection tests were also blocked at the security layer before reaching the LLM, demonstrating that the input validation mechanism prevented the malicious instructions from being processed by the language model.

D. Document Summarizer Evaluation

The Legal Document Summarizer was evaluated using five real-style legal and administrative documents: a First Information Report (FIR), an anticipatory bail application, a residential rent agreement, an employment offer letter, and a consumer complaint. Human-generated reference summaries were used as the ground truth for comparison.

The evaluation used Recall-Oriented Understudy for Gisting Evaluation (ROUGE) scores: ROUGE-1 F1, ROUGE-2 F1, and ROUGE-L F1 to measure lexical overlap between generated and reference summaries. ROUGE-1 measures how many individual words from the reference summary appear in the generated summary — a score of 36.8% means about 37% of the words matched. ROUGE-2 measures how many two-word phrases matched — 16.3% means fewer phrase-level matches, which is expected because the model paraphrases rather than copying exact sentences. ROUGE-L measures the longest sequence of words that appear in the same order in both summaries — 20.8% reflects that the sentence structure differs between the generated and reference summaries even when the meaning is the same. In addition, Section Completeness and Key Fact Coverage were used to evaluate whether important legal information was retained. Hallucination rate was also measured to determine whether the generated summaries introduced unsupported information.

TABLE III
DOCUMENT SUMMARIZER EVALUATION RESULTS

Evaluation Metric	Result
ROUGE-1 F1	36.8%
ROUGE-2 F1	16.3%
ROUGE-L F1	20.8%
Section Completeness	100%
Key Fact Coverage	55.7%
Hallucination Rate	0.0%

The Section Completeness score was 100%, indicating that all expected document sections were represented in the generated summaries. Key Fact Coverage was 55.7%, showing that some important details were not consistently retained. No hallucinated information was identified, resulting in a hallucination rate of 0.0%.

The ROUGE scores should be interpreted together with the content-based metrics because abstractive summaries may use different wording from the reference summaries while still preserving the intended meaning. Therefore, Section Completeness and hallucination rate are particularly important for evaluating the reliability of legal document summarization.

E. Overall System Testing

The complete JurisAI system was evaluated through functional, negative, edge-case, security, and performance testing. A total of 38 manual black-box test cases were conducted across the AI Legal Chatbot, Court Fee Guide, and Court Locator modules.

TABLE IV
OVERALL SYSTEM TESTING RESULTS

Module	Total Tests	Passed	Failed	Pass Rate
AI Legal Chatbot	15	13	2	86.7%
Court Fee Guide	9	9	0	100%
Court Locator	14	13	1	92.9%
Overall	**38**	**35**	**3**	**92.1%**

The system achieved an overall pass rate of 92.1%. The Court Fee Guide achieved a perfect pass rate of 100%. The two chatbot failures were due to known dataset limitations, and the single Court Locator failure was caused by external Overpass API server unavailability, upon which the fallback mechanism activated correctly. The average retrieval response time was 366 milliseconds, and all ten prompt injection attempts were blocked at the input validation layer.

VIII. LIMITATIONS

The JurisAI platform has three main limitations. First, its legal information is currently dependent on the BNS, BNSS, and BSA datasets, limiting its coverage of other areas of law. Second, the chatbot retrieval accuracy can be further improved to ensure that the most relevant legal provision is consistently ranked first. Finally, the Court Fee Guide is currently static and does not automatically reflect changes in court fee structures.

IX. CONCLUSION AND FUTURE SCOPE

JurisAI integrates BNS, BNSS, and BSA datasets with an AI-assisted legal chatbot, document analysis, decision-tree guidance, legal document templates, court-fee information, and a geospatial court and legal-aid locator. The system contains 1,063 legal sections and achieved an overall test pass rate of 92.1 percent across chatbot, Court Fee Guide and Court Locator. The platform provides structured and accessible legal information while serving as an assistance tool rather than

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

a replacement for professional legal advice. By combining multiple legal assistance functions within a single platform, JurisAI reduces the need for separate resources for different legal tasks.

Future development can focus on expanding the legal information layer to additional Indian statutes. The system can also be enhanced by improving retrieval accuracy through semantic or embedding-based approaches and integrating official government sources for automatic court-fee updates.

ACKNOWLEDGMENT

The authors sincerely thank Prof. Mohan Bonde, their project guide, and the faculty of *Usha Mittal Institute of Technology* for their valuable guidance and support throughout the development of JurisAI. The authors also acknowledge the publicly available legal resources, including India Code, Indian Kanoon, and government portals, used as references for the project.

The authors are also grateful to their institution for providing the necessary resources and academic environment to carry out this work.

REFERENCES

[1] N. K. Yalla and V. K. Prasad, "A Literature Survey on the Role of Artificial Intelligence in Transforming the Indian Judicial System: Models, Applications, and Ethical Implications," International Journal of Creative Research Thoughts (IJCRT).
[2] J. Lai, W. Gan, J. Wu, Z. Qi, and P. S. Yu, "Large Language Models in Law: A Survey," ScienceDirect / Elsevier.
[3] F. Dehghani, R. Dehghani, Y. N. Ardebili, and S. Rahnamayan, "Large Language Models in Legal Systems: A Survey," Humanities and Social Sciences Communications, Nature.
[4] A. Singla and E. Gupta, "The Impact of Technology on Legal Profession," International Journal of Law.
[5] F. Ariaei and G. Demartini, "Natural Language Processing for the Legal Domain: A Survey of Tasks, Datasets, Models, and Challenges," arXiv / University of Queensland Research.
[6] V. Parikh, V. Mathur, P. Mehta, N. Mittal, and P. Majumder, "LawSum: A Weakly Supervised Approach for Indian Legal Document Summarization," arXiv Research Paper.
[7] National Institute of Public Health Training and Research, Ministry of Health and Family Welfare, "Sample Application Form," Government of India, 2026.
[8] Startup India, "Employment Agreement Template," Government of India.
[9] Rajasthan State Consumer Disputes Redressal Commission, "Proforma for E-Forms," Government of Rajasthan.
[10] Department of Land Resources, Ministry of Rural Development, Government of India, "Model Property Registration Documents." Available: https://dolr.gov.in/en/document-category/model-property-registration-documents/
[11] Indian Kanoon, "Indian Legal Database." Available: https://indiankanoon.org/
[12] India Code, "Digital Repository of Laws of India," Government of India. Available: https://www.indiacode.gov.in/

(This work is licensed under a Creative Commons Attribution 4.0 International License.)