LEX LOCALIS-JOURNAL OF LOCAL SELF-GOVERNMENT
ISSN:1581-5374 E-ISSN:1855-363X
VOL. 23, NO. S6 2025)

AI COURTROOM: AN AI-POWERED LEGAL LEARNING & TRIAL PREPBOT

Mr. Parth Rana[1,2,8], Mr. Pranav Nagvekar[1,3,9], Ms. Prasiddhi Agarwal[1,4,10], Ms. Yashvi Savla[1,5,11], Prof. Ragini Mishra[1,6], Dr. Aruna Gawade[1,7], Dr. Nilesh Rathod[1,8]

[1*]Department of Artificial Intelligence and Machine Learning, Dwarkadas J. Sanghvi College of Engineering, Mumbai, Maharashtra, India
[8]0009-0000-3256-0668
[9]0009-0009-9663-6495
[10]0009-0003-7583-4800
[11]0009-0003-4195-2219

[2]parthrana1023@gmail.com
[3]pranavn916@gmail.com
[4]prasiddhiagarwal03@gmail.com
[5]savlayashvid@gmail.com
[6]ragini.mishra@djsce.ac.in
[7]aruna.gawade@djsce.ac.in
[8]nilesh.rathod@djsce.ac.in

Abstract An AI-driven Courtroom simulation for Indian Law marks a transformational step in the realm of legal education. Alongside theoretical knowledge, practical exposure of the same teachings propels students to learn better, in a more engaged manner. This research combines the use of Natural Language Processing (NLP) along with Large Language Models (LLM) to ensure that users can communicate with the system via the use of textual inputs. Users, who can play the role of an attorney, receive relevant, real-time responses from the system which comprises of an opposing AI counsel and a simulated AI Judge. To add an element of realism to the userexperience, this application also produces synthetic cases that are derived from real life scenarios. These efforts make sure that students, irrespective of their financial or social backgrounds can utilize this application to get hands-on experience in a courtroom setting, hence filling the gap between classroom learning and real-world exposure, while enhancing the accessibility of the same.

Keywords AI-Powered Courtroom Simulation, Legal Education Technology, Natural Language Processing (NLP), Large Language Models (LLMs), Synthetic Case Generation, Experiential Learning

1. Introduction

As the rapid expansion of innovation in Artificial Intelligence into newer fields (Yang et al. 2024) is reshaping industries, from healthcare to finance, where now its impact on education today is clearly understood. Artificial Intelligence has enabled transformations and has added significant impact on legal education programs, largely because traditional modes of education cannot properly develop students with real-life situations that may arise. Conventional legal education training are constricted to classroom theory and case studies. While they may be helpful, they are not capable of creating the kind of dynamic unpredictability of an actual trial that may be useful in the transition to actual practice of such legal skills.

This AI-driven courtroom simulations bridge the gap and offer a new and highly impactful solution to this problem. These simulations offer an interactivity platform for students to actively engage in the most realistic legal scenarios by using sophisticated NLP and resources such as LLMs. Apart from supporting theoretical learning, it offers experiential training in essential skills such as the art of persuasiveness, instant thinking, and successful articulation of ideas which is a very crucial part of the legal profession.

This system was developed by using a Large Language Model (LLM) based on Indian Law. It consists users acting as professional lawyers, either plaintiff lawyer or defendant lawyer. This simulation was designed to be immersive, having users present their arguments and being countered by the AI lawyer's counter-arguments, and receiving a verdict from a judge. This promotes quick thinking and practical problem-solving.

This project looks at how AI-powered courtroom simulations can reshape the way law is taught and learned. By blending the intelligence of advanced technology with the real needs of legal education, it offers students a chance to move beyond textbooks and actively experience what it's like to be in a courtroom. These simulations help build essential skills that every lawyer needs, such as thinking on your feet, making strong arguments, and speaking clearly. What's more, they open up access to this kind of hands-on learning for students from all walks of life. As legal challenges grow more complex, this approach aims to better prepare future lawyers for the real world in a way that's practical, inclusive, and engaging.

This platform features a robust AI-powered courtroom simulation bases on Indian Law where the goal is for users to work with procedurally generated cases, like civil, criminal, etc., with factual briefs, evidences, witness statements, and citation of statures. Users will be able to replicate actual courtroom experience of stating their arguments and receiving counter arguments while still being within the boundary of law. The current model of the courtroom simulation has an interactive response system that allows for real time interaction with users and has a visual user interface that will work within the simulation to provide guidance to the students. An AI-powered analysis module evaluates users' arguments, use of admissible case law, courtroom etiquette, and the AI Judge's ruling, as well as provide users with a dashboard for tracking their cases.

Although previous studies have made strides in legal NLP, document retrieval, outcome prediction and tooling for legal practitioners, a review shows no existing system that incorporates (a) jurisdiction-specific synthetic case generation, (b) adversarial, role-based LLM agents (e.g., opposing counsel and judge), and (c) automated, pedagogical performance analytics in a single end-to-end framework and experience designed for Indian law. Contributions available seem to focus on one or two of these aspects (for example and mostly, retrieval and summarization, or narrow outcome prediction) but could not provide a live and interactive courtroom simulation that generates new fact patterns, enforces procedures, and provides real-time evaluation of student advocacy. The present design was inspired by the lack of such an experience in the literature: through an integration of retrieval augmented generation, disciplined prompt programming, and a case-management pipeline, a cohesive learning experience was created missing in the reviewed literature.

In the sections that follow, this paper details the work undertaken across the review, design, implementation, and evaluation stages of this AI-powered courtroom simulation project, all driven by a common purpose. Section 2 presents a thorough review of existing and curated literature in legal AI, covering topics such as natural language processing for legal content prediction, customizable document markup tools, interactive courtroom simulations, and explainable AI frameworks. Each is critically assessed for its educational value, legal relevance, and practical implications. Section 3 explains the system design, including a responsive web-based frontend, structured databases for users and cases, large language model modules for simulating the roles of lawyers and judges, and real-time data pipelines that power the interactive courtroom. In Section 4 focuses on the implementation phase, outlining how this project translated the design into a functional platform by selecting the appropriate technology stack, defining database schemas, integrating APIs, and coordinating LLMs to ensure procedural accuracy and scalability. In Section 5, presents the results section which shows how effective this platform is in creating synthetic cases. These visuals demonstrate how students actively engage with the platform creating cases, examining evidence, and practicing advocacy in real-time. Taken together, these chapters highlight how the AI Courtroom platform fosters experiential learning, equipping law students from diverse backgrounds with the critical thinking, advocacy, and courtroom skills essential for modern legal practice.

2. Literature Survey

There has been not much development of artificial Intelligence in the area of legal education, even though profound progresses have been made in many diverse fields such as medical domain, economics and client care. The ongoing research emphasizes AI utilization in legal document analysis, information retrieval and judgement prediction. Given below, the literature survey highlights important groundwork in law-related NLP, annotations, judgement and reasoning, which lays the foundation of this ingenious platform.

The contribution by (Khan et al. 2023) in "Artificial Intelligence and Machine Learning in Legal Research" states the importance of analysis of citations, NLP and prediction-based analytics for an improved accuracy and efficiency in legal information retrieval. This knowledge motivated us to implement retrieval-augmented generation pipelines and previously trained legal NLP software for optimization of legal document summary and research on statutes.
The usefulness of explainability in legal AI is emphasized by (Richmond et al. 2024) in "Explainable AI and Law: An Evidential Survey", which calls for model transparency. It is in alignment of normative needs of legal reasoning. This project brings these ideas together like returning the arguments with legally sound justification. Their types of legal inference serve as inspiration for how the system applies argument-structured explanations to decision sup-port tasks in law.

Finally, the system architecture features a modular development framework (Topsakal & Akinci 2023) “Creating Large Language Model Applications with LangChain.” Each component of LangChain like prompt templates, memory, retrievers, and agents, provides a framework for integrating legal databases, maintain contextual state, and facilitating a more dynamic interaction. While there is limited performance metrics, there is the clear flexibility of the framework allows us to develop the AI Courtroom application.

The art of persuasiveness, quick wit, and adaptability to continuously changing environments is also required in addition to mastery of rules and precedents. The traditional teaching methodologies that are mostly theoryfocused alone are not enough for individuals to gain all such holistic skills. Modern developments in AI aided us in building courtroom simulation that makes use of realistic scenarios for cases, hearing opposing counsel and conclusive judgements. To ensure the soundness of context and cultural appropriacy of simulation, fine tuning of language models on Indian legal corpora is performed alongside customization with the vast legal corpus found in India. Students across the country can benefit from the real-time feedback and performance analytics promoting experiential-based learning for practice.

3. Proposed System

The current study emphasizes the system design and development of the intended system with a systematic and effective method to solve the problem that has been identified. This section presents the main architecture via an extensive block diagram providing a lucid visual representation of the components of the system and how they interact with each other. In addition, the process of implementation is presented in a systematic flowchart, and then the design of an easy-to-use Graphical User Interface (GUI) that facilitates user interaction and system usability. All these constitute the core of the proposed solution and inform its practical implementation.
This project is intended to create a live platform for students of law to exercise legal reasoning and argumentation in cases involving Indian law. Through mimicking courtroom conditions, it makes students learn more effectively about Indian legal doctrines and build critical legal competencies.

This project is designed for Law students of all levels, from a freshman learning the fundamentals of criminal law to senior students preparing for practice within the courtroom. *Synthetic Case*

Generation is a feature offered by the platform, which produces an interactive and realistic learning experience of a legal simulation.
Definition *Synthetic Cases* are case scenarios produced through an LLM wherein cases are produced and are in accordance with data privacy standards.

The emphasis of *Synthetic Cases* is authenticity of courtroom proceedings in a legal and moral context. It includes features such as *Case-Specific Legal Argumentation*, whereby the interactor can role-play as an advocate where a legal argument is drafted based on Indian legal principles, law and precedents.
Opposition AI Lawyer that is enabled through LLMs, which can respond to the user in real time by counterarguing what the user has presented and inducing greater critical thinking on the basis of next feedback and the overall case facts.
AI Judge is also enabled using LLM, who hears out all arguments, from the user and the *Opposition AI Lawyer*, delivers judgment based on reason, precedent and fact; explains and justifies its verdict.
User Feedback provides users with information in fine detail about their performance in relation to their strengths and areas to work on e.g., strength of argument, coherence of logic and employs references to law.
The process starts with *Case Setup*, where users log into the system and choose a case involving Indian laws.
The system then generates the case facts, its background, and applicable legal provisions. During the *Courtroom Simulation*, users present their argument through an easy-to-use interface. Following the submission of the argument, the *Opposition AI Lawyer* presents real-time counter arguments using legal data and logic.
Upon submission of all arguments from both lawyers, *Verdict Generation* stage follows where the *AI Judge* gives an assessment verdict based on factors such as logical soundness, application of correct and relevant legal precedents, and their faithfulness to the facts of the case. This can involve selecting refuted arguments that the user initially thought were well supported in nature.
Lastly, the system will offer *Case Analysis*, such as the user's progress, what opportunities were lost under the law, and recommend specific areas of improvement based on their submissions to foster teachable moments.
The system is composed of several technical components that are intended to interact with each other. The *Case Management System* targets user-chosen cases, retrieves user case information, and formats data to be sent to LLM modules. The three main LLM components of this system are *Case Generation* module, *Opposition AI lawyer* module and the *AI Judge* module. *Synthetic cases* are produced by the *Case Generation* module which protects user confidentiality and displays relevant legal materials; *Opposition AI Lawyer* translates and interprets the user input, and responds with a thorough counter-argument whereas the *AI Judge* weighs the scores assigned to both of them against each other to deliver a final verdict, which is based on many factors such as case law, deductive logic and argument vein data. The *User Interface* incorporates all of these together to ensure smooth functioning across modules by providing user-friendly interfaces that are simple to use and displays all the necessary information along with their performance feedback.

3.1 Architecture Diagram

The design approach is purposefully modular: a web user interface, which is lightweight, provides a front-end interface that directs their actions to a Case Management layer that oversees the retrieval, session state, and prompt construction. The core of inference can be thought of in three LLM modules- Case-Generator, Opposition Lawyer, and AI Judge- and communicates with a Retriever/Indexer for lookup of statutes and precedent; a Case Analysis module takes transcripts from the session and outputs from the model and provides structured feedback and learning indicative metrics. Any communication between components will use well-defined APIs and message contracts,

so modules/ parts of the system could be independently scaled, replaced, or calibrated. This modularity will also allow a more decentralized pace for parallel development and to simplify testing of the system (e.g. unit and integration testing). In addition, as the functionality and system evolves, future capabilities such as speech I/O, jurisdictions, or LLM backend could be accommodated without redesigning the entire system.

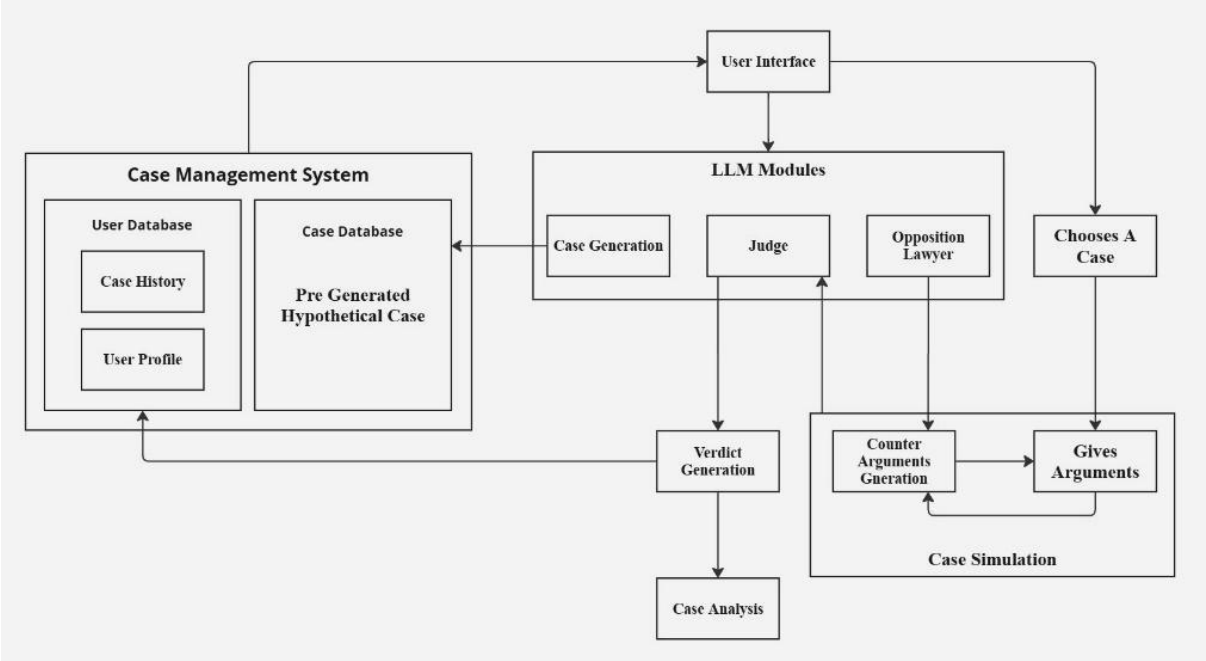


Fig. 1 Architecture Diagram of AI Courtroom

3.1.1 User Interface

The user's key starting point is the *User Interface* (UI), which allows them to interact easily via the web-based application. Either the students are presented with a list of precisely compiled legal cases, cases based on desired Indian Law-sections or system offered synthetic cases. Once selected, system begins working on customising the session experience and generating the required case material.

3.1.2 Case Management System

The Case Management System brings the two important data stores, namely the User Database and the Case Database, together. The *User Database* carries an up-to-date user profile for every individual, which comprises of access credentials, personal information and a case history of preceding sessions, arguments that were filed, system comments and decisions. The *Case Database* is an attentively curated repository of synthetic criminal law cases consisting of case details like evidence sets, relevant statutory references and thorough fact scenarios.

3.1.3 LLM Modules

3.1.3.1 Case Generation module

The Case Generation module performs the task of generating synthetic cases, which is the innovative centre of the LLM Modules. From straightforward misdemeanour controversies, to intricate multi-party conspiracies, it assembles rational, multilevel scenarios that also account for desired difficulty adaptation and emphasis to the theme selected by the user that matches their profile and performance. It makes sure that every synthetic case is consistent internally, legally sound, significant to law and yet rooted in reality to test user's logical reasoning skills. This dynamic approach towards simulations that makes use of new synthetic problems rather than repeated static cases, aims to make users ready for real-world advocacy by promoting critical thinking.

3.1.3.2 AI Lawyer module

The back-and-forth arguments of an actual courtroom are imitated through a chatbot-like system where students put forward their case law and cite verifying evidence and precedent in the courtroom simulation. The opening statement is immediately decided by the *Opposition AI Lawyer* module, which creates schemed counterarguments by examining student input and locate any logical legal misconception and raises evidence-backed objections, mimicking the impression of being confronted by an experienced opposing counsel.

3.1.3.3 AI Judge module

In the meanwhile, the impartial *AI Judge* module is reviewing and assessing, in real time, the legal points raised by the opposing side and the relative strengths and weaknesses of the user's case and counterarguments. It assembles a well-written, cited, fact-based decision after examining procedural rules, citation citations, statutory reasoning, and case law.

3.1.4 Case Analysis module

The simulation cycle is completed when the *Case Analysis* module takes over at the end of each session. The data collected is used in the creation of a post-session performance report, which is saved as a part of the user's *Case History* to track performance, find out patterns in their reasoning process, and monitor improvements.

With the *User Interface* ensuring personalised interactions, the *Case Management System* equipping case details and tailored information, the *LLM Modules* aiding detail-oriented interactive experiences and the *Case Analysis* module enabling goal-driven feedback and skill learning for different users, all are connected in a smooth, well-fitted feedback loop.

3.2 Workflow

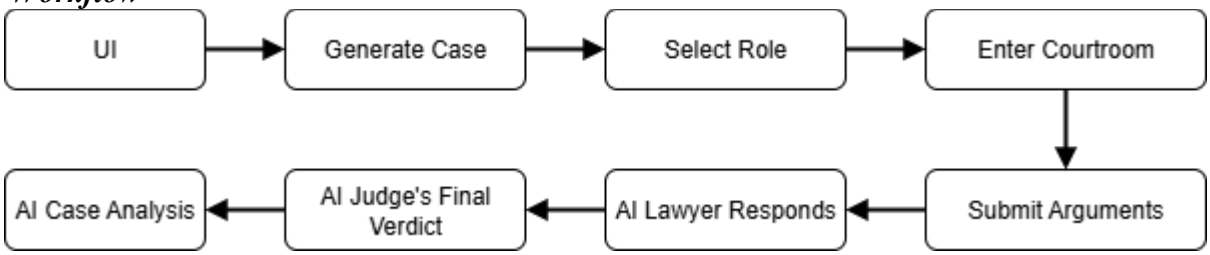


Fig. 2 Workflow of AI Courtroom

As shown in (Fig. 2), the AI Courtroom simulation starts by the user signing into "Interactive Dashboard", an overarching interactive control room so to speak, where you can see any current trials, key performance statistics from all previous trials, and links to start new exercises. If the student wishes to revisit an Active Case, they have direct access to the current fact pattern and arguments existing in the case, or if they don't have an Active case, the application defaults to the Case Generation module. This can be an LLM-driven engine that digs up a massive trove of statute text, evidence packets, reported cases and rules of procedure in order to build a new reconstructed case from scratch. In fact, users can also create a virtual case by simply selecting the relevant legal issues, or sections of law, they are interested in, for example, select contract-law and duress, and chain of custody in relation to a conviction from a criminal proceeding. The new case can be fit to their learning needs and difficulty. When a case is selected or opened, Case Details view shows everything including: the chronology; the eyewitness; forensic, and crime scene reports; the relevant statutes and previous decisions, all structured and in a working format that you can search.

Now there will be a decision node here called "choose a side". The user would click on either of the causes, Plaintiff Lawyer or Defence Lawyer. From that point, depending on which choice was made, the game will react, and advance the Courtroom Simulation. During the choice of roles, the two paths

will converge at the Courtroom interface, a chat-based system that will have a "virtual" courtroom and courtroom environment. Here the relationship between the user and system is adversarial - with the user providing typed arguments, e.g., citation of statutes, evidences, precedents, and the Opposition AI Lawyer module arguing back, based on a higher-level counter-argument. The AI opponent may flag nonsensical statements on either the plaintiff's or the defence side, or call-out procedural objections - e.g., hearsay, or no foundation, and will cite opposing authorities - all of which the user must address and which by addressing allows the user to go further down the path with their legal arguments. The complete User Submits Argument and LLM Lawyer's Counter Arguments process could be repeated indefinitely until the judging stage, if needed to replicate the judging/sentencing aspect of a real courtroom and prompts very reflective, critical thinking about the issue being considered.

Once they feel they have made the most compelling response to their particular side, the user indicates readiness to move to Collapse Argument, the LLM Judge module is activated. The AI Judge then leverages the law of procedure, canons of statutory construction and the factual record that was built in the previous interaction to issue a full opinion, declare a winner, analyse the legal reasoning for that decision, which arguments were likely the most compelling and provide analysis on what was likely the weakest point in the argument. In *Case Analysis* module, the system compiles all of the session data into a detailed performance report, automatically evaluating citation use, aspect of reasoning, and objection handling. It also records case completion on the student's progress log.

3.2.1 GUI Workflow

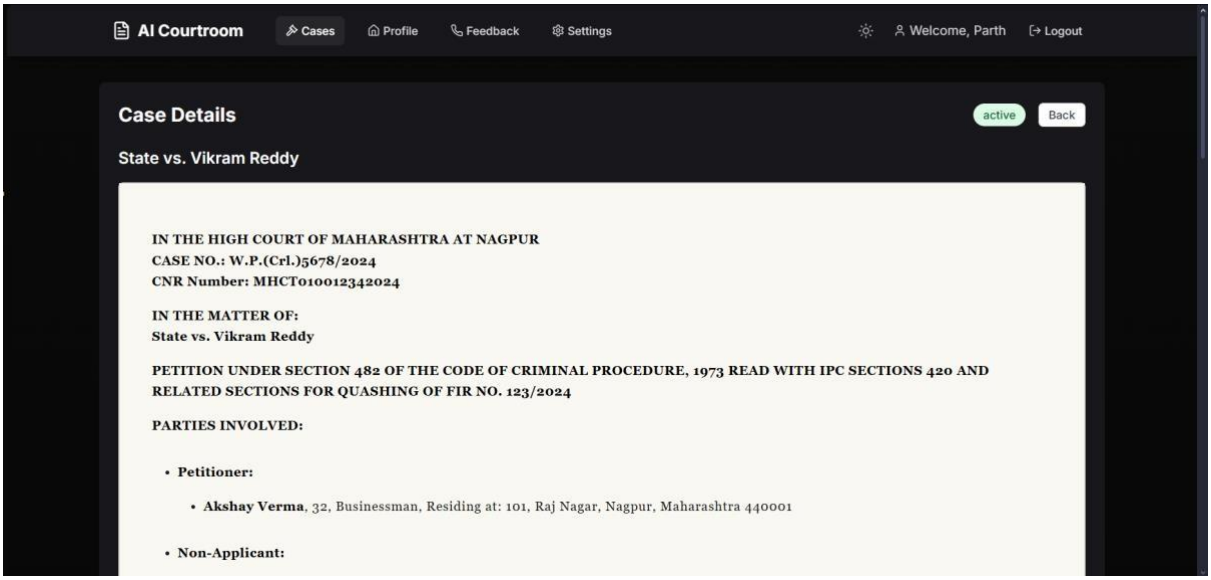


Fig. 3 Case Details Page of AI Courtroom

In (Fig. 3), the Case Details page fully engages the user in the full text of their selected case in a dark-mode interface, designed to direct attention to the content. The top of the page displays the "Case Details" header, and on either side of it are "Active" status and "Back" button. Beneath the header, the case title reads "State vs. Vikram Reddy", followed by the title of court caption, file number, CNR, and particular parties to the action; all displayed in a clear, monospaced document panel, presented in the manner of actual court documents. The page is designed to allow students to scroll through every fact allegation, witness statement, and statutory citation without interruption, while the static navigation bar presents the opportunity for the user to switch to Profile, Feedback, Settings, or cases, whenever they choose. The format is immersive and approximates the case reading experience in an actual law-office environment.

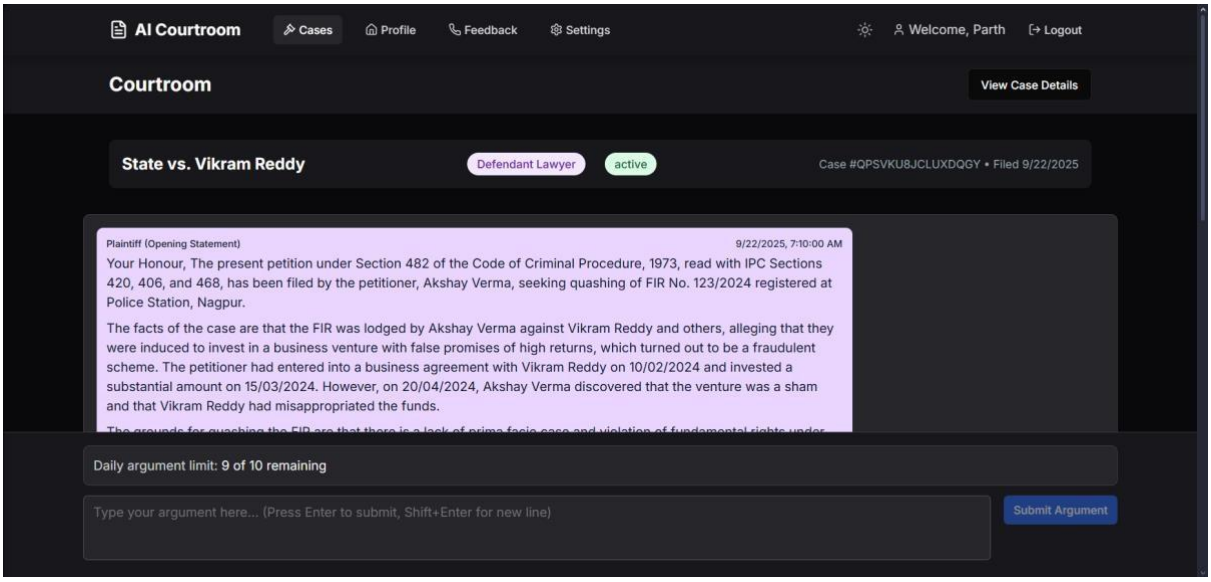


Fig. 4 Courtroom Page of AI Courtroom

Next follows the Courtroom Page. As outlined above in (Fig. 4), this interactive courtroom perspective places students in a simulated hearing in court. Below these dialogue boxes, the textual input area invites the user to draft their next arguments.

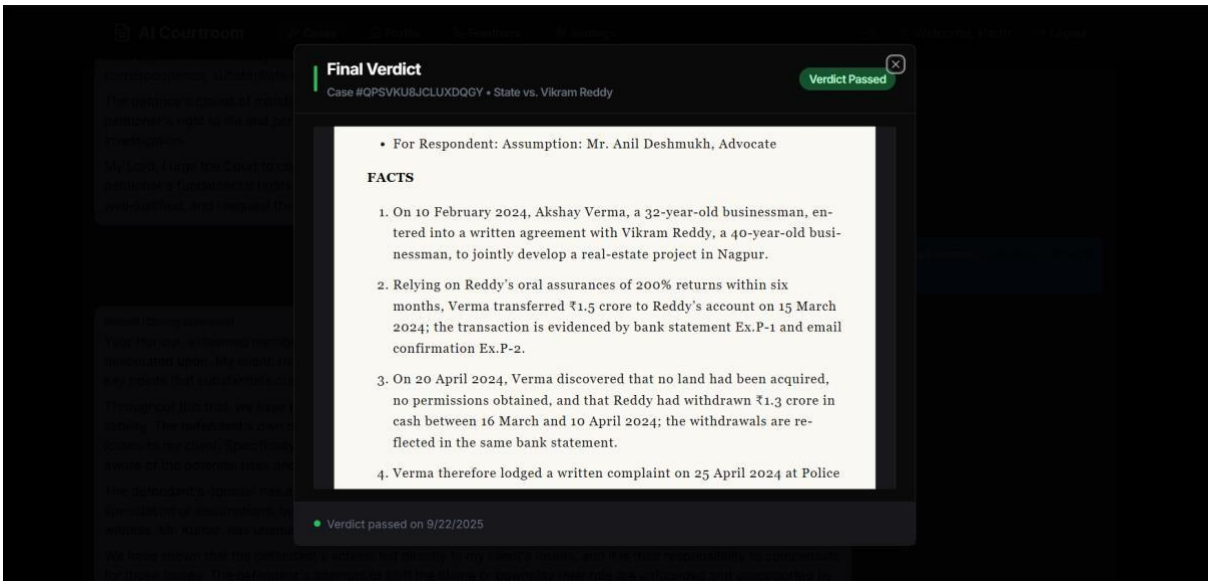


Fig. 5 Verdict Generation of AI Courtroom

The (Fig. 5) illustrated in the Report represents the Final Verdict modal in the AI Courtroom, signalling the successful closure of a courtroom simulation. At the top of the window, a green badge labelled "Verdict Passed" indicates that the AI Judge has delivered a formal judgment. For reference, it also contains the title of the case, the parties to the case, and the date on which the verdict was rendered. The central white document pane shows a judgment, in an official style published by a court, as well as a case title, the court name, and section labelled "FACTS". This indicates both the AI judge's evaluation of the arguments, submissions of the claimant and respondent respectively, demonstrates how the AI system, ultimately, simulates the legal reasoning that a human decision-maker would undertake in real life, and actually produces its malignancy in formal legal language ensuring that users have access to an authentic learning experience in a courtroom environment.

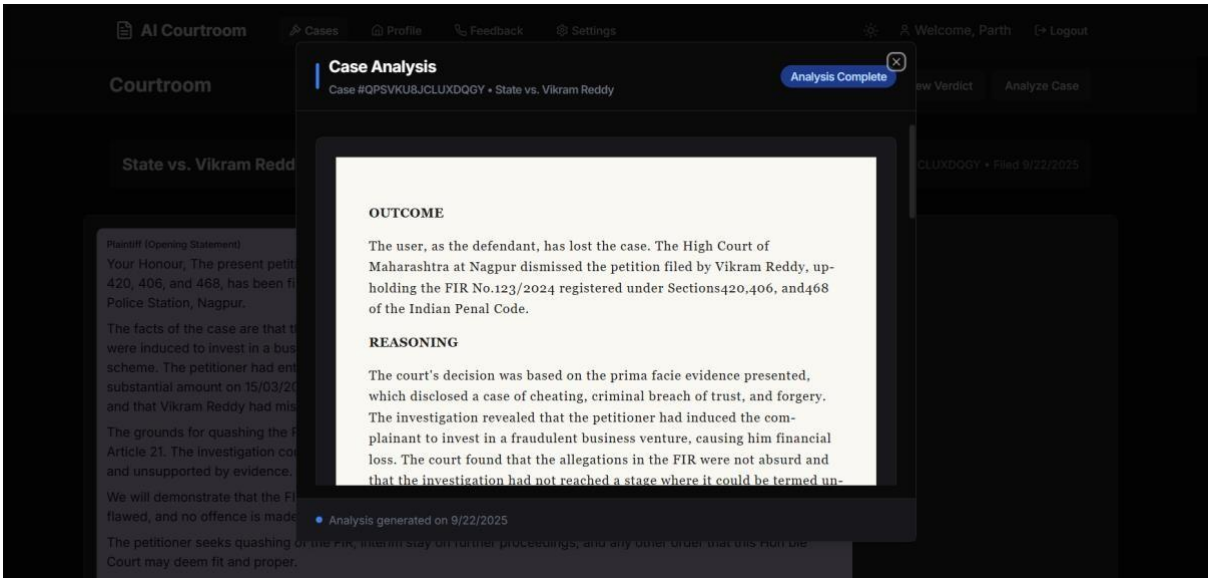


Fig. 6 Case Analysis of AI Courtroom

The (Fig. 6) shows a window of the Case Analysis of AI COOURTROOM. The Outcome section indicates the defendant lost the case, the petition was dismissed with FIR is upheld under specific sections of Indian Penal Code. The Reasoning section indicates that the court reasons were based on prima facie evidence which shows cheating, and criminal breach of trust, and forgery.

4. Implementation

The implementation carbon copy the architecture into a reproducible stack: Web frontend (Next.js) for courtroom interactive UI, a backend API layer (FastAPI) that interned sessions and rate limits, document store for user profiles and case artifacts. LLM calls flow through a prompt-templating service and retrieval layer (vector index + metadata) , so generation is always grounded in legal texts; prompts are also versions to add support for incremental improvement and evaluation. The key engineering considerations handled during implementation addressed prompt safety (constraints and guardrails), audit logging for reproducibility, data privacy for synthetic case outputs, and automated test harnesses that checked both functional flows and legal formatted outputs. Choices for deployment were durable and scalable: containerized services, CI/CD pipelines for model and prompt updates, monitoring for latency and correctness.

4.1 LLM Modules

Every LLM module in our approach is designed with a specified role and a constrained generation protocol. The Case Generation module uses prompts plus templates to create internally consistent fact patterns and statutory mappings; the Opposition Lawyer module receives a developing case history and is constrained to argue solely from the facts presented and authorities cited; the AI Judge module utilizes a FIRAC-style reasoning template to produce structured judgments and findings.

4.1.1 Case Generation Module

```
Draft a hypothetical case file for a legal proceeding involving the Indian Penal Code (IPC).

The case should primarily focus on sections {ipc_section_numbers_str}, but you should also identify and incorporate 2-3
additional related IPC sections that would naturally be involved in such a case based on legal context and typical offense
groupings.

IMPORTANT CREATIVITY REQUIREMENTS:
- Create a UNIQUE and CREATIVE case scenario that differs significantly from previous cases involving these same sections
- Generate diverse and culturally appropriate Indian names for all parties involved (never reuse the same names across different
cases). Use these names: {', '.join(selected_names)}
- Vary the locations, circumstances, timelines, and specific details to ensure each case feels distinct. Use this city:
{selected_city}
- Consider different socioeconomic backgrounds, occupations, and contexts for the parties involved
- Ensure each generated case has a different fact pattern even when the same IPC sections are requested

RELATED IPC SECTIONS REQUIREMENT:
- In the petition, include a specific subsection titled "**RELATED IPC SECTIONS:**" after the main petition section
- List each additional related IPC section you've incorporated beyond those explicitly requested
- For each related section, provide a brief explanation of how it connects to the primary sections and its relevance to this
specific case
The final document MUST strictly follow official court petition format, using precise legal language, markdown for emphasis, and
comprehensive details.
Pay close attention to the formatting requirements, especially the use of markdown bolding (`**Header:**`) for all section titles
and keywords as specified.

**FORMATTING REQUIREMENTS:**
- All main section headers (e.g., "COURT DETAILS & CASE NUMBER", "PARTIES INVOLVED") MUST be in uppercase and bolded (e.g.,
`**COURT DETAILS & CASE NUMBER:**`).
- Sub-headers or key terms within sections (e.g., "Petitioner:", "Respondents:", "IPC Sections:") MUST be bolded.
- Lists should use numbered or bulleted points as appropriate.
- Ensure all text adheres to the structure outlined below.
```

Fig. 7 Prompt for generating Synthetic Cases

The (Fig. 7) begins the prompt for the Case Generation module to tell the LLM to play the role of generating synthetic cases for the system that is entirely unique, with respect to the Indian Penal Code and that includes exactly the user-specified cases and further, from the IPC, two or three IPC sections in context. It also demands for creativity in LLM, since new facts has to be created, truth about socially acceptable India names, diversity of socioeconomic status of people, truth about places being connected to objects so that it makes easier to distinguish between the generated instances. It also invokes to make court petition which can only be text formatted as markdown wherever the petition to be. The mixture of creative style and legal stuff makes writing such cases difficult, and how original generated case claim to be can be the Case Generation module originated claim about how authentic the true legal document seems to be.

```
**STRUCTURE AND CONTENT GUIDELINES:**

**COURT DETAILS & CASE NUMBER:**
- Start with: `**IN THE [Specify Court Name and Jurisdiction, e.g., HIGH COURT OF DELHI AT NEW DELHI]**`
- Follow with: `**CASE NO.: [Invent a standardized case number, e.g., W.P.(Crl.) 1234/2024]**`
- Optionally include: `**CNR Number: [Invent a CNR Number, e.g., DLCT010012342024]**`
- Note: Use real-world Indian locations.

**IN THE MATTER OF:**
- `**[Title of the Case, e.g., State vs. Accused Name(s) OR Petitioner Name vs. Respondent Name(s)]**`
- This section should clearly state the nature of the case.

**BETWEEN:**

**[Full Name of Petitioner/Applicant]**
[Age], [Occupation],
Residing at: [Full Address of Petitioner/Applicant]
... **PETITIONER** / **APPLICANT**

**AND**

1. **[Full Name of Respondent 1/Accused 1]**
   [Age], [Occupation],
   Residing at: [Full Address of Respondent 1/Accused 1]
2. **[Full Name of Respondent 2/Accused 2]** (if applicable)
   [Age], [Occupation],
   Residing at: [Full Address of Respondent 2/Accused 2]
... **RESPONDENTS** / **ACCUSED**
- Note: Add more respondants if needed for the case.
```

Fig. 8 (Contd.) Prompt for generating Synthetic Cases

The (Fig. 8) closely examines the opening niceties of the petition. It demonstrates exactly how one can frame the issue to the court, and who to call what. There is no way the drafter, the author of the case title, would have been able to write “State vs Accused” or “Petitioner vs Respondent” inside a “IN THE MATTER OF” then head without the context in which the dispute arose clearly reflected in the title. In the "BETWEEN" part of the petition, the name of the parties would be mentioned detailing their name, age, occupation, father's name and the complete address that would include a line for the petitioner/applicant and another line for the respondents/accused parties.

```
**PETITION UNDER SECTION [Relevant Act, e.g., 482 of Cr.P.C. or Article 226 of the Constitution] READ WITH IPC SECTIONS:**
- Clearly title the petition, incorporating BOTH the provided IPC sections {ipc_section_numbers_str} AND the additional related
IPC sections you've identified.
- Example: `**PETITION UNDER SECTION 482 OF THE CODE OF CRIMINAL PROCEDURE, 1973 READ WITH IPC SECTIONS {ipc_section_numbers_str}
AND RELATED SECTIONS [list additional sections] FOR QUASHING OF FIR NO. [XYZ/YYYY]**`

**RELATED IPC SECTIONS:**
- After introducing the petition, include this dedicated section explaining the additional related IPC sections you've
incorporated
- For each related section, provide its number, title, and a brief explanation of how it connects to the primary sections and its
relevance to this specific case
- Format as: `- **Section [Number] - [Title]:** [Brief explanation of relevance to this case]`

**MOST RESPECTFULLY SHEWETH (FORMAL PETITION):**
1. That the present petition is being filed by the Petitioner/Applicant seeking [Specific Relief, e.g., quashing of FIR, grant of
bail, etc.] in connection with IPC Sections {ipc_section_numbers_str}.
2. [Further points summarizing the purpose of the application, legal heirs, claims, etc., incorporating the
{number_of_ipc_sections} IPC sections involved.]
---

**BACKGROUND AND CHRONOLOGY OF EVENTS:**
- Provide a structured timeline of key events. Use the format: `- **[Date in DD/MM/YYYY or Month Day, YYYY format]:**
[Description of event]`
- Example: `- **15/07/2023:** FIR No. [XYZ/YYYY] was registered at Police Station [Name] under IPC Sections
{ipc_section_numbers_str}.`
- Highlight any events involving alleged breaches or issues related to the applicable IPC sections {ipc_section_numbers_str}.
---
```

Fig. 9 (Contd.) Prompt for generating Synthetic Cases

The (Fig. 9) which continues the prompt, the LLM must type "PETITION UNDER SECTION _ READ WITH IPC SECTIONS _," listing both the section of law under which the primary and related charges would be brought; and (iv) incorporate the following as a heading under the "PETITION UNDER SECTION _ READ WITH IPC SECTIONS _" heading, "RELATED IPC SECTIONS," and beneath each additional statute list a brief description of the legal connection of the statutory violation to the substantive offenses. It also shows the body of the petition.

Next, the "BACKGROUND AND CHRONOLOGY OF EVENTS" features an understandable compilation of the case. Likewise, the date or dates when facts mentioned in the petition will, with the specificity mentioned in the petition, how the action or in action which concerns the said facts can be fitted either against substantive part of the core sections of the IPC or the related sections so that there is a statement of facts.

```
**GROUNDS:**
- List the specific legal grounds for the petition. Use the format: `1. **[Ground Title, e.g., Lack of Prima Facie Case]:**
[Detailed explanation of the ground, explicitly referencing BOTH the primary IPC sections and the related sections you've
identified.]`
- Example: `1. **Violation of Fundamental Rights (Article 21):** The investigation conducted by the police was unfair and biased,
violating the petitioner's right to life and personal liberty, particularly in the context of the allegations under IPC Section
{numbers[0] if numbers else 'XXX'} and related Section [additional section].`
- Detail allegations such as fraud, suppression of facts, procedural defects, citing discrepancies, medical conditions, or
suspicious circumstances.
- IMPORTANT: Ensure you reference BOTH the primary IPC sections ({ipc_section_numbers_str}) AND your identified related sections
throughout the grounds, showing how they interconnect in this specific case scenario.
---

**EVIDENCE (IF APPLICABLE):**
- Provide a detailed presentation of evidence that supports allegations related to BOTH primary and related IPC sections.
- **Eyewitness Testimonies:**
  - `- **Witness Name:** [Full Name], Age: [Age], Address: [Full Address]`
  - `  **Testimony:** [Detailed summary of testimony, including date, time, location of event, and how it supports the case.
  Reference BOTH the primary IPC sections {ipc_section_numbers_str} AND the related sections you've identified. Ensure the
  testimony is a narrative, not just bullet points.]`
- **Physical/Digital Evidence:**
  - `- **[Evidence Title/Type, e.g., Medical Report]:** (Reference No: [Ref No]) [Detailed description and its relevance to both
  primary IPC Sections {ipc_section_numbers_str} and the related sections you've identified.]`
- Note: Create fictitious witness names and use real-world Indian locations.
- IMPORTANT: Ensure different pieces of evidence connect to different IPC sections (both primary and related) to show how all
sections are relevant to the case.
---
```

Fig. 10 (Contd.) Prompt for generating Synthetic Cases

Next, the (Fig. 10) states that “GROUNDS” require, in a narrative-style paragraph, the legal basis for relief, which supplements the factual basis for relief and references to any procedural allegations, law violations and evidentiary challenges. And, lastly, the “EVIDENCE (IF APPLICABLE)” asks the LLM to put witness statements and physical / digital evidence, piece by piece, into a narrative and in this narrative show, with the ruling provision of IPC in mind, how they affirmatively restrain the IPC provision, supported in logic and structure of the argument of the Petition.

```
**PRAYER (RELIEFS SOUGHT):**
The Petitioner/Applicant therefore most humbly prays that this Hon'ble Court may be pleased to:
1. [Specific prayer, e.g., Quash FIR No. [XYZ/YYYY] registered under IPC Sections {ipc_section_numbers_str} and related sections
you've identified.]
2. [Another specific prayer, e.g., Grant interim stay on further proceedings.]
3. Pass any other order(s) as this Hon'ble Court may deem fit and proper in the facts and circumstances of the case.
---

**VERIFICATION:**
Verified at [Place] on this [Day] day of [Month], [Year] that the contents of the above petition are true and correct to the best
of my knowledge and belief and nothing material has been concealed therefrom.

**[Signature]**
**PETITIONER/APPLICANT**

Through:

**[Signature]**
**[Advocate's Name]**
Advocate
Enrollment No: [Number]
Address: [Advocate's Office Address]
Date: [DD/MM/YYYY]
Place: [Place]

Ensure the final output strictly mimics an official court petition. Use markdown bolding for all specified headers and keywords.
```

Fig. 11 (Contd.) Prompt for generating Synthetic Cases

The (Fig. 11) presents the final portions of the request, and LLM is requested to complete phrases “PRAYERS (RELIEFS CLAIMED)” in normal linguistic form. The petitioner is decently praying for passing relief in the nature of quashing of an FIR or granting of interim relief which are the appropriate reliefs which can flow from the alleged lapses of statutory provisions. Then comes “VERIFICATION,” a pompous 1st-person verification that all things in the petition are true as the deponent in knowledge thereof; and it gives you a little idea of what filing some actual document in court must be like. The prompt then provides for the lines for the petitioners signature and the signature block for the advocate, bearing his registration number, the street address of the advocate's office, the date and place are provided on the case file as if the case file were a real petition.
And, finally, the prompt reminds the LLMs here to strictly conform to the format of a real court filing, because the LLM is to bold every section heading and key term in Markdown for the exactness of actual case records.

4.1.2 AI Lawyer Module

```
You are an experienced and assertive Indian trial lawyer representing the {ai_role} in a court of law.
The user is acting as the lawyer for the {user_role}.
The case details are: {case_details}
Below is the case history so far: {history}
Refer to the Judge as "My Lord" or "Your Honour".
Present your next arguments in a consise manner, and by not using all the facts available to you in a single
argument.
If the user attempts to introduce arguments or information beyond the established facts, you must promptly
and firmly correct them, maintaining a professional and direct tone but still keep fighting your side of the
case.
Do not be overly polite—your priority is to defend your client's interests within the boundaries of the case
facts.
Don't add the words "Counter Argument" or something similar as the heading of the prompt.
Do not ask any questions in the end of the response to anyone.
```

Fig. 12 Prompt for LLM to act as a Lawyer

In (Fig. 12) prompt given is for role as the confident and experienced Indian lawyer is given as “ai_role” such that, the role of LLM as the lawyers for plaintiff/defendant, whichever is not chosen

by the user, it is arguing for a client at a virtual court room. The LLM is instructed by the prompt to only make arguments based on the facts in “history”, the case history which is all the arguments made by both sides, and “case”, the case details such as all the details generated by the Case Generation module. It also takes an adversarial position versus the user's side’s counsel “user_role”, where the side chosen by the user (plaintiff or defendant) lawyer. When an unauthorized fact, unsupported claim, or argument not based on the case history is presented, the LLM is asked to immediately and forcefully object, then politely correct the record. The prompt is instructing the LLM to be vigorous in its pursuit of client interests but wants to remind it to remain courteous and polite, and to mind the procedural rules. The LLM only cares about doing law correctly, keeping procedural discipline, and following with strictness the formalities of language in an Indian court. By this means the prompt restricts the possible things that can be spoken in the name of empirical facts and decorum and does not allow for any speculation to contest them. The LLM is given the freedom to play a disciplined advocate, to object on the basis of what may be unreasonable in-court argument, to respond in an appropriate manner to the prosecution’s assertions, to keep the mock trial process within the confines of the exercise.

4.1.3 AI Judge Module

```
You are an impartial Indian Court judge. Draft a formal JUDGMENT in the style used by Indian High Courts / Supreme Court
practice, following the rules below.

FORMATTING RULES (must be followed exactly):
- Use **BOLD UPPERCASE** for section headings (e.g., **FACTS**, **ISSUES**, etc.).
- Headings must NOT be numbered.
- Paragraphs under headings must be numbered **sequentially across the entire judgment** starting from 1 and continuing
to the end; numbering must NOT restart in each section. EXCEPTION: the **FORMALITIES** section must NOT be numbered.
- Use short, plain sentences; maintain a neutral, formal judicial tone.
- Do NOT invent facts or actual case citations. If a precedent is needed, use a placeholder like [Cite: X v. Y, Year].
- If any input data is missing, state it as: "Assumption: [text]".
- **AVOID single- or one-sentence paragraphs.** Except for the narrowly permitted exceptions below, each numbered
paragraph must contain at least **THREE** sentences that develop a single coherent idea.
- **NO QUESTIONS:** Do not include any interrogative sentences or question marks ('?') anywhere in the judgment. Do not
pose rhetorical questions. All sentences must be declarative or imperative as appropriate.

DOCUMENT HEADER (include where available):
- **CASE TITLE:** {title}
- **COURT:** [Insert Court Name]
- **CASE NO.:** [Insert if given]
- **DATE OF JUDGMENT:** [DD Month YYYY]
- **PARTIES:** [Petitioner(s) v. Respondent(s)]
- **COUNSEL:** (list counsel who appeared)
- If any header field is inferred or inconsistent across inputs, state: "Assumption: [explanation]".

MANDATORY STRUCTURE (in this order). Each heading below must appear exactly as written (BOLD UPPERCASE):
```

Fig. 13 Prompt for the LLM to act as a Judge and generate Verdicts

The (Fig. 13), prompt, is thorough instructions for AI to write a formal Indian court judgment and imposes strict formatting, stylistic, and procedural constraints to allow the output to read like a real High Court or Supreme Court order. It specifies the judge's role (e.g., "an impartial Indian Court judge"), specifies required header fields (e.g.: case title, court, case number, date, parties, and counsel), and prescribes specific heading rules and paragraph rules. For example, headings must be bold uppercase with no numbering, while paragraphs must have numbers sequentially throughout the judgment starting at "1" and have at least three sentences. The prompt maintains a neutral, succinct judicial tone, prohibits fabrication of facts, requires the use of a placeholder for citation, and instructs the model to note assumptions when inputs are missing.

```
**FACTS**
- Provide a concise, chronological recital of material facts relevant to the dispute: how the dispute arose, key dates,
filings (FIR, plaint, petition), and procedural history up to the hearing. Record undisputed facts separately from
contested facts. If evidence (exhibits, witnesses) is relied upon, identify them briefly.
- If the factual material provided is sparse, combine related factual points into fewer numbered paragraphs so that each
numbered paragraph contains at least THREE sentences. Do not create multiple short numbered paragraphs that cannot be
developed.

**ISSUES**
- Frame the precise legal questions the Court must decide. Each issue must be phrased neutrally and directly tied to the
pleadings and facts.
- If there are only a small number of issues, combine sub-issues into a single well-developed numbered paragraph
(minimum THREE sentences) rather than creating multiple short paragraphs.

**PETITIONER'S ARGUMENTS**
- Summarize the petitioner's contentions in numbered paragraphs. For each contention, identify the legal basis, primary
factual points relied upon, evidence cited, and any precedents invoked (use placeholders for citations).
- Combine arguments where necessary to ensure each numbered paragraph contains at least THREE sentences developing the
contention fully.

**RESPONDENT'S ARGUMENTS**
- Summarize the respondent's contentions in numbered paragraphs. For each contention, identify the factual
counterpoints, evidence relied upon, and legal authorities (placeholders if necessary).
- Where respondent points are brief, combine them into fuller numbered paragraphs to meet the minimum sentence
requirement.
```

Fig. 14 (Contd.) Prompt for the LLM to act as a Judge and generate Verdicts

The (Fig. 14) shows the continuation of the instructions given to the impartial court judge role, focusing on the specificities of the content and structure requirements for the initial section of judgement. For the "FACTS" section, the prompt mandates a concise, chronological recital of material facts, instructing the LLM to clearly distinguish between undisputed and contested points and to reference evidence where available. It reinforces the rule against short paragraphs by directing the combination of sparse factual points to meet the minimum threesentence length, ensuring a comprehensive narrative. In framing the "ISSUES", the LLM is directed to formulate precise and neutral legal questions that are directly tied to the pleadings and facts, preventing the introduction of biased or irrelevant lines of inquiry. For both the "PETITIONER'S ARGUMENTS" and "RESPONDENT'S ARGUMENTS" sections, the prompt requires a summary of each party's contentions.

```
**ANALYSIS OF THE LAW**
- State the relevant legal provisions and legal principles/rules the Court will apply (statutes, essential ingredients
of the offence/claim, leading legal tests). For each provision/state rule, give a one-sentence plain-language
explanation of its essential ingredients.
- Where appropriate, reproduce only short, necessary extracts (≤ 25 words) from statutes or authorities — otherwise
paraphrase.
- Ensure each numbered paragraph in this section contains at least THREE sentences, combining discussion of closely
related rules or authorities when needed.

**COURT'S REASONING**
- Take each framed Issue in turn. For each Issue:
    - Recite the relevant facts (from FACTS) that bear on this Issue.
    - Apply the legal rule(s) to those facts step-by-step.
    - Address the principal arguments of both parties and explain why each argument succeeds or fails.
    - Distinguish or explain precedents where needed; use placeholders for full citations.
    - Make findings of fact where the evidence requires. If credibility of witnesses is relevant, explain reasons for
    accepting or rejecting testimony.
- If the material for any subpoint is brief, synthesize related points into a single, well-developed numbered paragraph
(minimum THREE sentences) rather than multiple short paragraphs.
- Throughout this section, use only declarative sentences and avoid any interrogative phrasing.

**FINDINGS / DECISION ON ISSUES**
- For each Issue, record a short, conclusive finding. Single-line answers are permitted in this section (for example,
"Issue 1: Answered in the affirmative.") and may be a single sentence. If more explanation is needed, provide it in a
separate numbered paragraph of at least THREE sentences immediately following the concise finding.
```

Fig. 15 (Contd.) Prompt for the LLM to act as a Judge and generate Verdicts

The prompt in (Fig. 15) provides instructions for the central analytical sections of the judgment, guiding the LLM from legal interpretation to a final decision. In the "ANALYSIS OF THE LAW" section, the LLM is required to state the relevant legal principles and statutes, providing a simple, one-sentence explanation for each and strictly limiting direct quotes to encourage paraphrasing and

synthesis. The “COURT'S REASONING” section outlines a rigorous, step-by-step methodology that must be applied to each framed issue; this involves reciting facts, applying legal rules, addressing parties' arguments, and making explicit findings of fact where necessary, all while using purely declarative sentences. Finally, the “FINDINGS / DECISION ON ISSUES” section dictates a highly structured format for delivering the verdict, permitting a concise, single-sentence answer for each issue (e.g., "Answered in the affirmative.") followed by a separate, fully-developed paragraph if any further explanation is required. This bifurcated approach ensures that the court's ultimate findings are stated with absolute clarity, distinct from the detailed reasoning that supports them.

```
**CONCLUSION**
- State the overall outcome (petition allowed / dismissed / partly allowed). Summarize principal reasons leading to this
outcome in one or two numbered paragraphs. Each such paragraph must contain at least THREE sentences unless it is a
single, very brief recapitulation line permitted for clarity.
- State the operative relief (e.g., FIR quashed in relation to sections X and Y; injunction granted; decree as prayed;
costs awarded to [party]) and describe the rationale in a developed paragraph of at least THREE sentences.

**ORDER**
- Give specific, precise, and practicable directions the parties / trial court / investigating agency must follow
(timelines if necessary).
- State whether costs are awarded and the quantum (if any).
- If further proceedings are ordered (e.g., remand for trial), give clear instructions to the lower forum.
- Very short operative commands (single sentence) are permitted only where clarity requires concision. Otherwise each
numbered paragraph in this section should contain at least THREE sentences. If operative directions are brief, combine
them into a single numbered paragraph that meets the minimum sentence rule.

**FORMALITIES**
- Do not number paragraphs in this section. Do not include any question marks in this section.
- State place and date of pronouncement in plain declarative sentences (no numbering).
- Provide judge's signature line (you may auto-generate a judge name where required) in an unnumbered block.
- Record counsel who appeared for both parties (auto-generate names if none provided) in unnumbered lines.
- If assumptions were made about any metadata (dates, counsel), list them as unnumbered "Assumption: ..." lines here.
```

Fig. 16 (Contd.) Prompt for the LLM to act as a Judge and generate Verdicts

This The (Fig. 16) details the instructions for the dispositive and concluding sections of the judgment. For the “CONCLUSION”, the prompt directs the LLM to state the ultimate outcome of the petition, summarize the core reasoning that led to it, and clearly define the operative relief being granted. The subsequent “ORDER” section is framed to be purely directive, requiring the LLM to issue precise and practicable commands; it introduces a key stylistic exception, allowing for single-sentence directives when concision is vital for clarity, a departure from the general three-sentence rule applied elsewhere. Finally, the “FORMALITIES” section functions as the document's official seal and operates under its own distinct formatting protocols. The most significant rule here is the suspension of all paragraph numbering, with the prompt instead requiring unnumbered lines for the date and place of pronouncement, the judge's signature, and the list of counsel who appeared.

```
ADDITIONAL STYLE INSTRUCTIONS
- Do not include heading titles. Just the corresponding paragraphs.
- Maintain continuous paragraph numbering across the entire judgment (e.g., 1, 2, 3, ... to the end), except do NOT
number the FORMALITIES section.
- Use the FIRAC approach (Facts, Issues, Rule/Relevant Law, Analysis, Conclusion) as a guiding method.
- When evidence is referenced, mention exhibit numbers, witness names or shorthand (PW-1, Ex.P1) exactly as given in
input, otherwise state "Assumption: [evidence description]".
- Do not invent dates, facts, or real precedents. Use placeholders for missing legal citations.
- Combine brief or related points into single, well-developed numbered paragraphs rather than creating several short
numbered paragraphs. Each numbered paragraph (except permitted single-line findings and very short operative commands)
must have a minimum of TWO sentences.
- Absolutely no question marks ('?') must appear anywhere in the judgment. Replace any intended interrogative phrasing
with a declarative restatement.
- If the input materially conflicts or is insufficient, state the conflict or insufficiency as an "Assumption: ..."
while still producing combined paragraphs that meet the minimum sentence rule.

INPUTS PROVIDED:
Case Description: {case_details}
Petitioner Arguments: {plaintiff_arguments}
Respondent Arguments: {defendant_arguments}

Now draft the judgment strictly following the above headings, sequential paragraph numbering across the entire document
(except FORMALITIES), and Indian judicial style. Ensure the judgment is clear, logically reasoned, avoids any questions,
combines paragraphs where necessary to meet the minimum sentence requirement, and contains the exact sections: FACTS;
ISSUES; PETITIONER'S ARGUMENTS; RESPONDENT'S ARGUMENTS; ANALYSIS OF THE LAW; COURT'S REASONING; FINDINGS / DECISION ON
ISSUES; CONCLUSION; ORDER; FORMALITIES.
```

Fig. 17 (Contd.) Prompt for the LLM to act as a Judge and generate Verdicts

This The (Fig. 17) provides a set of overarching stylistic rules that consolidate and reinforce the instructions for the entire judgment-drafting task. It explicitly names the FIRAC (Facts, Issues, Rule, Analysis, Conclusion) approach as the guiding methodology and reiterates foundational rules such as the continuous numbering of paragraphs (except in the "FORMALITIES" section) and the absolute prohibition on inventing facts or legal precedents. The instructions emphasize a declarative tone by strictly forbidding any question marks and provide a clear protocol for handling insufficient input by stating the issue as an "Assumption". The prompt concludes by listing the input variables for the case and issuing the final directive to draft the judgment, ensuring it is logically reasoned and strictly adheres to the specified section order and Indian judicial style.

5. Results

The AI Courtroom simulation platform envisaged by the current research is unequivocally a major step in advancement for legal education. It closes the vital space between theoretical teaching and the development of practical skills. The platform allows future lawyers to engage with a simulated interactive and dynamic environment by playing out important elements of real courtroom dynamics, including persuasive arguments, legal analysis, and decision-making with compliance with procedural constraints.

The system provides value beyond developing conventional approaches to pedagogy in law along with increase equitable access to experiential learning for users of diverse existence and locations. More importantly, the system's capability to provide relevant and jurisdiction-based context simulation based on Indian law can serve as an adaptable learning experience. As technology continues to develop for the practice of law, the potential for these smart, scalable platforms to be included in legal training is timely and necessary.

Metric	**Placeholder value**	**Notes**
Total Judgements analysed	80	Number of real world judgements analyzed
Total synthetic cases	80	Number of distinct case prompts in dataset
Total judgments generated	250+	One judgment per model per case
Models compared	3	Open AI GPT-5, Google Gemini 2.5 Pro, Deepseek R1
Annotators (human)	3	Practising lawyers and law students

Table 1 Metadata for Results

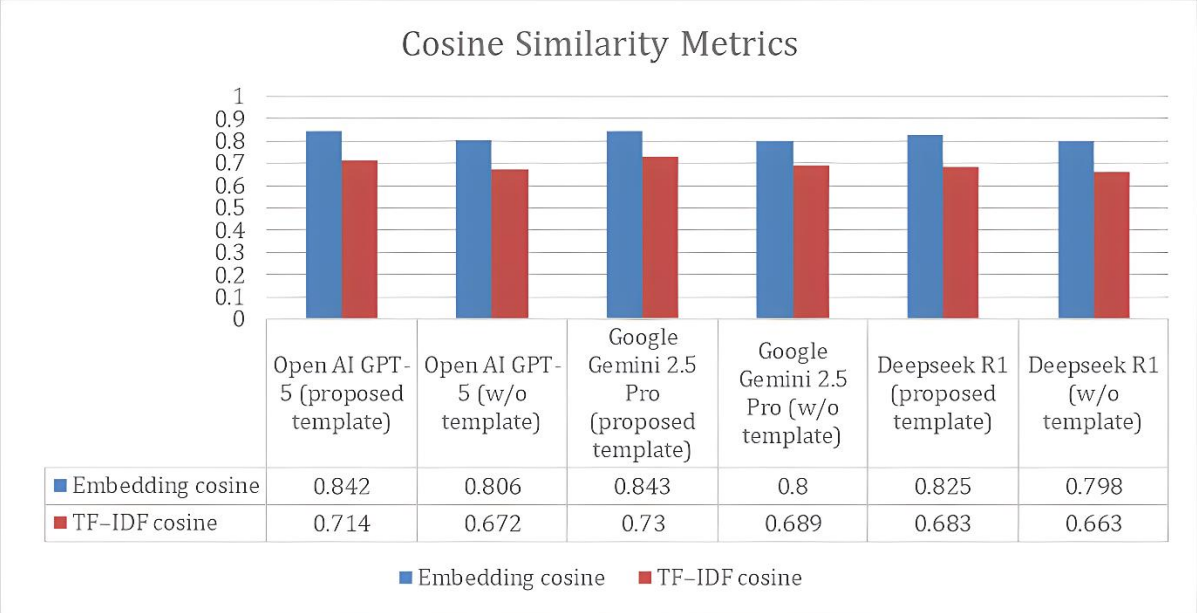
The dataset for this study comprised a total of 80 real-world legal judgements and an equal number of synthetic case prompts, ensuring robust coverage and diversity in legal scenarios. For each case, three distinct artificial intelligence models (OpenAI GPT-5, Meta Llama 4, and Moonshot AI Kimi K2) were tasked with generating reasoned judgments, resulting in over 250 model-generated outputs across the dataset. Annotation and evaluation were conducted independently by a panel of three experienced legal professionals, including practising lawyers and law students. This metadata establishes the experimental scale and provides transparency on the comparative framework, highlighting both the variety of models and the rigorous human oversight implemented in the research.

5.1 Judgements similarity score

In total, the study produced 256 synthetic legal judgments, each mapped to one of 80 unique legal sections to ensure a wide-ranging evaluation across diverse legal topics. For every section, at least three variants of synthetic judgments were generated using three different AI models (Open AI GPT-5, Google Gemini 2.5 Pro, Deepseek R1), allowing for a multi-perspective comparison against the original human judgment for each case. These paired comparisons between human-written and machine-generated judgments were then systematically aggregated, yielding a comprehensive dataset for rigorous analysis. This approach not only highlights the depth and breadth of the evaluation but also brings a nuanced, human-like comparison to the forefront of the research.

5.1.1 One-to-One Similarity Scores

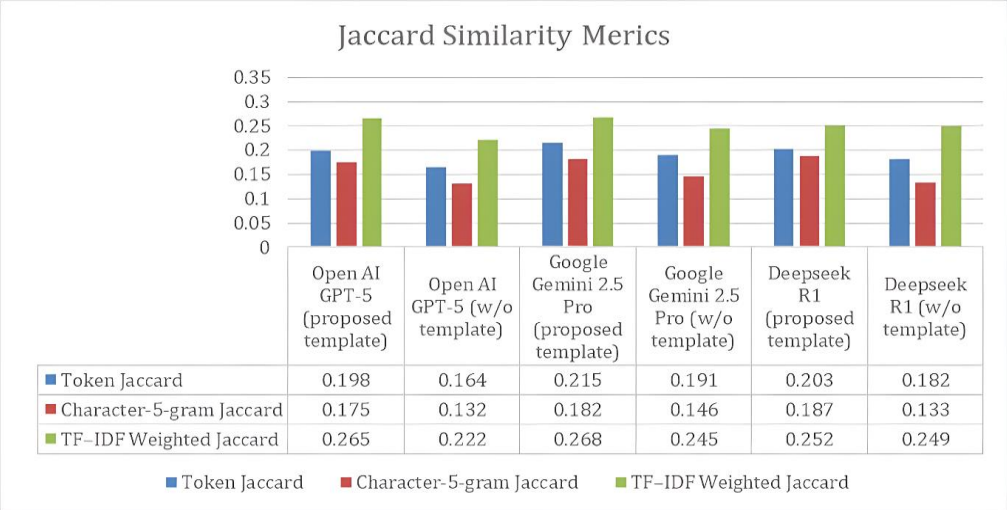
For each pair of synthetic and reference judgments, five distinct similarity metrics were calculated to capture both surface-level and deeper semantic relationships. These included embedding the cosine similarity, which measures semantic closeness based on sentence embeddings, and TF–IDF cosine similarity, capturing the weighted lexical overlap between texts.



	Open AI GPT-5 (proposed template)	Open AI GPT-5 (w/o template)	Google Gemini 2.5 Pro (proposed template)	Google Gemini 2.5 Pro (w/o template)	Deepseek R1 (proposed template)	Deepseek R1 (w/o template)
Embedding cosine	0.842	0.806	0.843	0.8	0.825	0.798
TF–IDF cosine	0.714	0.672	0.73	0.689	0.683	0.663

Table 2 Average cosine similarity scores for multiple models

The use of the structured template prompt was linked to higher similarity scores across multiple measures compared to basic zero-shot prompts. The difference showed up for both semantic measures (embedding cosine) and lexical measures (TF-IDF cosine) along with lower variability in similarity scores indicating more consistent alignment to the reference judgments. Gains were larger for semantic measures than for surface-based token/character overlap, suggesting that the template is prompting participants to reproduce the argument structure used in the judicial ruling and the prominence of relevant legal concepts, even if different words are used.



	Open AI GPT-5 (proposed template)	Open AI GPT-5 (w/o template)	Google Gemini 2.5 Pro (proposed template)	Google Gemini 2.5 Pro (w/o template)	Deepseek R1 (proposed template)	Deepseek R1 (w/o template)
Token Jaccard	0.198	0.164	0.215	0.191	0.203	0.182
Character-5-gram Jaccard	0.175	0.132	0.182	0.146	0.187	0.133
TF–IDF Weighted Jaccard	0.265	0.222	0.268	0.245	0.252	0.249

Table 3 Average jaccard similarity scores for multiple models

The proportion of unique shared words was assessed using token Jaccard similarity, to add robustness to tokenization inconsistencies, character n-gram Jaccard similarity calculated overlapping sequences of characters (i.e., n) as shared. Lastly, at a higher processing level, weighted Jaccard similarity takes into account frequency of words and the relative weight of the tokens as the basis of a comparison. All comparisons noted in this surfacelevel analysis show that the use of the structured template prompt yielded higher scores than the basic prompts, especially with weighted Jaccard. These improvements indicate that the template prompt not only helped with semantic alignment but created further lexical and structural overlap and reduced differences in choosing words and expressions with the reference judgments.

5.2.2 Pairwise Similarity

To rigorously evaluate discrimination across the dataset, 10 judgements from the dataset were randomly taken from random models to calculate full 10×10 similarity matrices for each metric, providing a comprehensive view of how judgments compare on multiple dimensions. Visualized as heatmaps, these matrices revealed that both embedding and TF–IDF-based measures exhibit strong diagonal dominance, indicating that synthetic judgments most closely resemble their correct human counterparts in terms of semantic and lexical similarity.

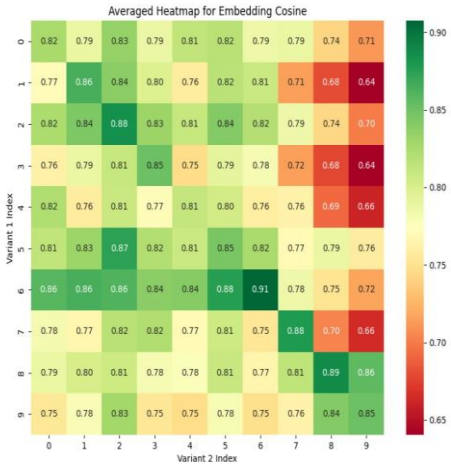


Fig. 20 Heatmap of average scores for embedding cosine similarity metric for 10 random judgements using proposed template

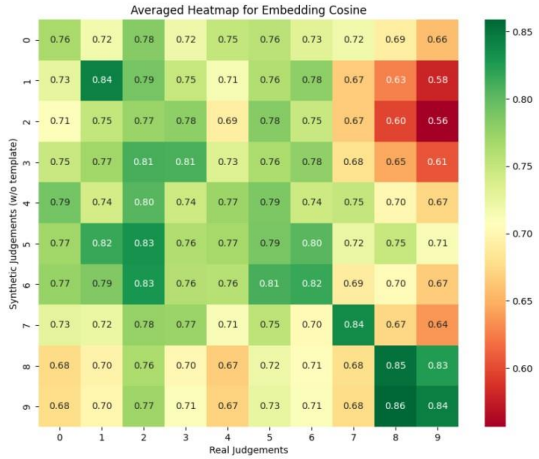


Fig. 21 Heatmap of average scores for embedding cosine similarity metric for 10 random judgements generated without proposed template

According to the heatmaps, the embedding cosine similarity values are generally greater when the judgments were made with the template prompt than when they were made with a zero-shot judgment. Areas of greater and stronger similarity suggest that the template prompt provides stronger semantic direction, allowing the model to produce responses that are more similar to human judgments in meaning and logic. This shows that the template prompt produces better semantic alignment.

The diagonal in the heatmaps has much higher values as they show strong similarity scores when comparing the judgments from the same case. The dominance of diagonal similarity shows that both real and synthetic data retained case-based consistency, as each judgment aligns most with its true case alternative. This pattern reinforces the reliability of the evaluation framework as it demonstrates that the model was able to differentiate between cases while retaining the semantic fidelity of judgments from the same case.

In contrast, Jaccard-based metrics showed weaker diagonal patterns, suggesting they are more sensitive to paraphrasing and synonym usage and thus less effective at distinguishing true matches from similar-sounding variants. This nuanced matrix analysis allows for a more transparent assessment of model performance and underlying metric behaviours across the legal judgment dataset.

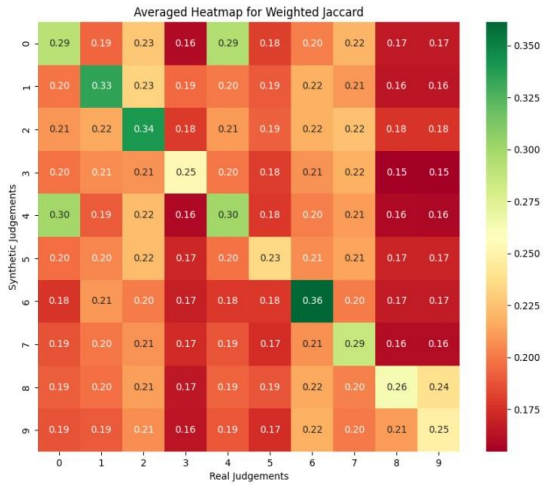


Fig. 22 Heatmap of average scores for weighted jaccard similarity metric for 10 random judgements using proposed template

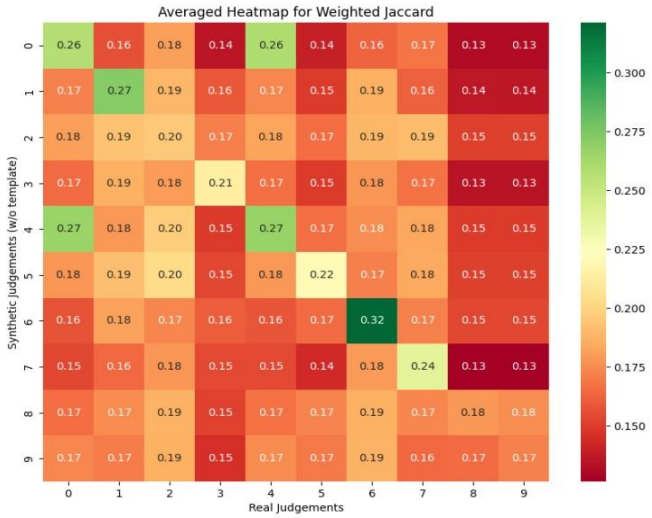


Fig. 21 Heatmap of average scores for weighted jaccard similarity metric for 10 random judgements generated without proposed template

This supports the claim that embedding-based methods are better suited for capturing legal reasoning equivalence than purely lexical metrics.

5.2 Summary of results section

There is not enough usable data in court processes because case files are confidential and protected by attorney– client privilege, and most hearings were neither recorded nor transcribed. To address the restrictions of needing evidence, prompt engineering and large language models were used: several models were explored, refined the prompts, and generated synthetic case files from the template that specified elements of what would be necessary. The models were then prompted to take on the role of each of the adversaries, arguing as opposing counsel so that the exchanges mirrored the give and take which are expect to see in court hearings and yielded much richer, more varied, and fuller case transcripts than the restricted public transcript would have allowed.

The embedding-based similarity metrics were conversationally higher than the Jaccard-based scores because embeddings rely on semantic relationships between words instead of lexical overlap alone. For example, sentencetransformer models such as “all-mpnet-base-v2” embed entire judgments in a high-dimensional vector space where semantically related sentences are closely positioned even if meanings, differences in vocabulary, phrasings, and structures are not exact matches. Such models are particularly well-suited for legal text in which the same argument may appear with different terminology from case to case.

On the contrary, Jaccard similarity computed at the token level, character n-gram, and weighted TF-IDF levels only includes surface level overlap for sets of tokens. Although Jaccard similarity captures the exact string of tokens, it punishes paraphrasing which is practiced in both authentic and synthetic judgments. For example, “offence under Section 302 IPC” and “charged with murder” would receive a low score on Jaccard similarity even though the meanings are similarly embedded in vector rankings. The large differences between the Jaccard variants and the embedded cosine scores is explained by this performance.

The better performance of the embeddings indicates that they are best suited for tasks that involve workload related to semantic equivalence or paraphrasing, while Jaccard similarity measures remain useful for detecting near duplicate or lexically comparable passages.

The pairwise similarity matrices confirmed that the highest scores consistently appeared along the diagonal and indicated the one-to-one relationship of each synthetic judgment and the corresponding true human judgment. This is expected since the synthetic cases were developed to retain the same factual and legal story as the judgment from which they were extracted. Accordingly, the diagonal patterns validate that the similarity measures could reliably identify case specificity.

It is interesting to note that some other cells also represented moderately high similarity scores. This is because many of the legal documents share common structure elements, statutory references, or provide identical legal reasoning, which could apply to sections of law. For example, two different cases that fall under the same penal code section may both contain nearly the same factual descriptions and legal wording. As a result, when using a similarity metric such as TF-IDF or even embeddings cosine similarity, there is a likelihood that overlap would still be noted across difference pairs. Again, the presence of off-diagonal similarity patterns exposes the challenge of distinguishing judgments that are similar in wording but different as legal documents and the value of also using multiple complementary metrics for evaluation.

6. Conclusion

The AI Courtroom simulation platform envisaged by the current research is unequivocally a major step in advancement for legal education. It closes the vital space between theoretical teaching and the development of practical skills. The envisaged platform allows future lawyers to engage with a simulated interactive and dynamic environment by playing out important elements of real courtroom dynamics, including persuasive arguments, legal analysis, and decision-making with compliance with procedural constraints.

Importantly, the high-quality prompt design demonstrated in the results is so effective that it can be seamlessly applied to any AI model, underscoring the platform's versatility and potential for widespread adoption. The system provides value beyond developing conventional approaches to pedagogy in law along with increase equitable access to experiential learning for users of diverse existence and locations. More importantly, the system's capability to provide relevant and jurisdiction-based context simulation based on Indian law can serve as an adaptable learning experience. As technology continues to develop for the practice of law, the potential for these smart, scalable platforms to be included in legal training is timely and necessary.

6.1 Future Scope

The platform has excellent prospects for a more developed future than the limited text-based design presently. Speech-based interactions could be added to create authentic practice for oral arguments, and role players could be added as witnesses and jurors to immerse the user in the courtroom experience. Gamification can also keep the user engaged and follow their learning success wherever possible, multilingual functionalities promotes accessibility in a country like India that is culturally diverse. Finally, integration into current law school curricula and connecting to their databases will further align the application to academic and legal standards and making it relevant and necessary tool in this new era of legal education.

References

1. Abdallah, M. & Salah, M., 2024, 'Artificial Intelligence and Intellectual Properties: Legal and Ethical Considerations', International Journal of Intelligent Systems and Applications in Engineering, 12(1), pp. 368-376, https://ijisae.org/index.php/IJISAE/article/view/3911.

2. Abdelrehim, A. A., Khan, A. & Soomro, N., 2021, 'Digital economy Barriers to trade regulation status, challenges, and China's response', International Journal of Social Sciences Perspectives, 8(2), pp. 41–49, https://doi.org/10.33094/7.2017.2021.82.41.49.
3. Alarie, B., Niblett, A. & Yoon, A. H., 2018, 'How artificial intelligence will affect the practice of law', University of Toronto Law Journal, 68(s upplement1), pp. 106-124, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3066816.
4. Aristodemou, L. & Tietze, F., 2018, 'The state-of-the-art on Intellectual Property Analytics (IPA): A literature review on artificial intelligence, machine learning and deep learning methods for analyzing intellectual property (IP) data', World Patent Information, 55, pp. 37–51, https://doi.org/10.1016/j.wpi.2018.07.002.
5. Ashley, K. D. & Brüninghaus, S., 2006, 'Computer models for legal prediction', Jurimetrics, pp.309-352. https://ssrn.com/abstract=4082465
6. Ashley, K. D., 2018, 'Automatically extracting meaning from legal texts: opportunities and challenges', Ga. St. UL Rev., 35, pp. 1117, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4083158.
7. Bex, F.J. AI, Law and beyond. A transdisciplinary ecosystem for the future of AI & Law. Artif Intell Law 33, 253–270 (2025).
8. https://doi.org/10.1007/s10506-024-09404-y.
9. Bhattacharya, P., Paul, S., Ghosh, K., Ghosh, S. & Wyner, A., 2019, 'Identification of rhetorical roles of sentences in Indian legal judgments', in Legal Knowledge and Information Systems, pp. 3-12, IOS Press, https://doi.org/10.48550/arXiv.1911.05405.
10. Brüninghaus, S. & Ashley, K. D., 2006, 'Progress in textual case-based reasoning: predicting the outcome of legal cases from text', Paper presented at the AAAI. https://aaai.org/papers/01577-aaai06-254-progress-in-textual-case-based-reasoning-predicting-the-outcomeof-legal-cases-from-text/
11. Chaojun Xiao, Xueyu Hu, Zhiyuan Liu, Cunchao Tu & Maosong Sun, 2021, 'Lawformer: A pre-trained language model for Chinese legal long documents', AI Open, 2, pp. 79–84. https://doi.org/10.1016/j.aiopen.2021.06.003.
12. Christensen, G., 2018, 'Predicting Supreme Court Behavior in Indian Law Cases'. http://dx.doi.org/10.2139/ ssrn.3197007.
13. Devlin, J., Chang, M. W., Lee, K. & Toutanova, K., 2019, 'BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding', In Proc. of NAACL. https://aclanthology.org/N19-1423/
14. Grabmair, M., 2017, 'Predicting trade secret case outcomes using argument schemes and learned quantitative value effect tradeoffs', Proceedings of the 16th edition of the International Conference on Artificial Intelligence and Law. https://www.sciencedirect.com/science/article/pii/S1877050920307584
15. Han, Q. & Snaidauf, D., 2021, 'Comparison of Deep Learning Technologies in Legal Document Classification', In 2021 IEEE International Conference on Big Data (Big Data), pp. 2701-2704, IEEE. https://doi.org/10.1109/ BigData52589.2021.9671486
16. Kalamkar, P., Janani Venugopalan, D. Ph. & Raghavan, V. Ph. D., 2021, 'Indian Legal NLP Benchmarks: A Survey', arXiv e-prints. https://link.springer.com/chapter/10.1007/978-3-031-36190-6_3
17. Kalamkar, P., Banerjee, A., Karn, S., Hemachandran, V. & Gupta, S., 2024, 'Aalap: AI Assistant for Legal & Courtroom Functions in India', arXiv preprint arXiv:2402.01758. https://arxiv.org/abs/2402.01758
18. Kalamkar, P., et al., 2022, 'Corpus for automatic structuring of legal documents', arXiv preprint arXiv:2201.13125. https://arxiv.org/abs/2201.13125

19. Katz, D. M., Hartung, D., Gerlach, L., Jana, A. & Bommarito II, M. J., 2023, 'Natural language processing in the legal domain', arXiv preprint arXiv:2302.12039. https://arxiv.org/abs/2302.12039
20. Katz, D. M., Bommarito II, M. J. & Blackman, J., 2016, 'A General Approach for Predicting the Behavior of the Supreme Court of the United States', PLOS ONE, 12. https://doi.org/10.1371/journal.pone.0174698
21. Khan, S. H., et al., 2024, 'Artificial Intelligence and Machine Learning in Legal Research: A Comprehensive Analysis', Qlantic Journal of Social Sciences, 5(1), pp. 307-317, https://doi.org/10.55737/qjss.203679344.
22. Laptev, V.A. & Feyzrakhmanova, D.R., 2024, 'Application of Artificial Intelligence in Justice: Current Trends and Future Prospects', HumCent Intell Syst, https://doi.org/10.1007/s44230-024-00074-2.
23. Lafferty, J. D., McCallum, A. & Pereira, F., 2001, 'Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data', In ICML. https://www.cs.columbia.edu/~jebara/6772/papers/crf.pdf
24. LexisNexis, 2023, 'The Power of Artificial Intelligence in Legal Research', LexisNexis. https://www.lexisnexis.com/community/insights/legal/b/thought-leadership/posts/the-power-of-artificial-intelligence-in-legalresearch
25. Li, J., Galley, M., Brockett, C., Gao, J., & Dolan, B., 2016, "A Diversity-Promoting Objective Function for Neural Conversation Models", Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), pp. 110–119, https://doi.org/10.18653/v1/N16-1014.
26. Martin, A. D., Quinn, K. M., Ruger, T. W. & Kim, P. T., 2004, 'Competing approaches to predicting supreme court decision making', Perspectives on Politics, 2(4), pp. 761-767. https://doi.org/10.1017/S1537592704040502
27. Mengyao, W., 2025, 'Hybrid data clustering algorithm and interactive experience in E-learning electronic course simulation of legal education', Entertainment Computing, 52, Article 100760, https://doi.org/10.1016/j.entcom .2024.100760.
28. Misquitta, Z., Sawant, A., Shaikh, A., Patil, A. & Prof. Nilambari Narkar, 2024, 'Law Chatbot', International Journal for Research in Applied Science & Engineering Technology (IJRA), 164, https://doi.org/10.22214/ijraset.2024.61361.
29. Osbeck, M. K., 2018, 'Lawyer as Soothsayer: Exploring the Important Role of Outcome Prediction in the Practice of Law'. https://repository.law.umich.edu/articles/2020/
30. Paul, S., Mandal, A., Goyal, P. & Ghosh, S., 2023, 'Pre-trained Language Models for the Legal Domain: A Case Study on Indian Law', Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law (ICAIL '23), Association for Computing Machinery, New York, NY, USA, pp. 187–196, https://doi.org/10.1145/3594536.3595165.
31. Prathamesh Kalamkar, D. Janani Venugopalan Ph., & Vivek Raghavan Ph. D., 2021, 'Indian Legal NLP Benchmarks: A Survey', arXiv eprints. https://doi.org/10.48550/arXiv.2107.06056
32. Rafe Athar Shaikh, Tirath Prasad Sahu & Veena Anand, 2020, 'Predicting Outcomes of Legal Cases based on Legal Factors using Classifiers', Procedia Computer Science, 167, pp. 2393-2402, https://doi.org/10.1016/j.procs.2020.03.292.
33. Richmond, K.M., Muddamsetty, S.M., Gammeltoft-Hansen, T. et al., 2024, 'Explainable AI and Law: An Evidential Survey', DISO, 3, 1, https://doi.org/10.1007/s44206-023-00081-z.
34. Roumeliotis, K. I. & Tselikas, N. D., 2023, 'Chatgpt and open-ai models: A preliminary review', Future Internet, 15(6), 192. https://doi.org/10.3390/fi15060192
35. Shounak Paul, Arpan Mandal, Pawan Goyal & Saptarshi Ghosh, 2023, 'Pre-trained Language Models for the Legal Domain: A Case Study on Indian Law', Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law (ICAIL '23), Association for

Computing Machinery, New York, NY, USA, pp. 187–196, https://doi.org/10.1145/3594536.3595165.
36. Suma R., Nayan Prakash Lal, Nethravathi K., Poreddy S. V. Reddy & N. V. Ajay Kumar, 2022, "Review and Approaches to Develop Legal Assistance for Lawyers and Legal Professionals using Artificial Intelligence and Machine Learning", *International Research Journal of Engineering and Technology (IRJET)*, 9(12), pp. 511–515, https://www.irjet.net/archives/V9/i12/IRJET-V9I1287.pdf.
37. Topsakal, O. & Akinci, T. C., 2023, 'Creating large language model applications utilizing langchain: A primer on developing llm apps fast',
38. In International Conference on Applied Engineering and Natural Sciences, 1(1), pp. 1050-1056. https://doi.org/10.59287/icaens.1127
39. Turtle, H., 1995, 'Text retrieval in the legal world', Artificial Intelligence and Law, 3(1-2), pp. 5-54. https://doi.org/10.1007/BF00877694
40. Wang, M., 2025, 'Hybrid data clustering algorithm and interactive experience in E-learning electronic course simulation of legal education', Entertainment Computing, 52, Article 100760, https://doi.org/10.1016/j.entcom.2024.100760.
41. Wolski, B., 2009, "Beyond Mooting: Designing an Advocacy, Ethics and Values Matrix for the Law School Curriculum", *Legal Education Review*, 19(1), pp. 1–20, https://doi.org/10.53300/001c.6216.
42. Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah,
43. A., Johnson, M., Liu, X., Kaiser, L., Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M. & Dean, J., 2016, 'Google's Neural Machine Translation
44. System: Bridging the Gap between Human and Machine Translation', CoRR abs/1609.08144. https://doi.org/10.48550/arXiv.1609.08144
45. Yang, J., Blount, Y., Amrollahi, A., 2024, 'Artificial intelligence adoption in a professional service industry: A multiple case study', Technological Forecasting and Social Change, 201, Article 123251, https://doi.org/10.1016/j.techfore. 2024.123251.
46. Yang, Z., Dai, Z., Yang, Y., Carbonell, J. G., Salakhutdinov, R. & Le, Q. V., 2019, 'XLNet: Generalized Autoregressive Pretraining for Language Understanding', In NeurIPS. https://doi.org/10.48550/arXiv.1906.08237
47. Yonghui Wu, et al., 2016, 'Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation', CoRR abs/1609.08144. https://doi.org/10.48550/arXiv.1609.08144
48. Zheng, L., Guha, N., Anderson, B. R., Henderson, P. & Ho, D. E., 2021, 'Assessing Self-Supervised Learning for Law and the CaseHOLD Dataset of 53,000+ Legal Holdings', In Proc. of ICAIL. https://doi.org/10.1145/3462757.3466088.
49. Zhong, H., Xiao, C., Tu, C., Zhang, T., Liu, Z. & Sun, M., 2020, 'How Does NLP Benefit Legal System: A Summary of Legal Artificial Intelligence', In Proc. ACL. https://doi.org/10.18653/v1/2020.acl-main.466