An LLM Powered Legal Data Warehouse and Precedent Retrieval System for Reducing Judicial Backlog

Debopriyo Sarkar[1], Sunetra Tewary[2]

[1]Department of Computer Science, Sidho Kanho Birsha University, Purulia-723149 West Bengal, India
Corresponding Author Email: *debopriyosarkar.official[at]gmail.com*

[2]Department of Computer Science, Sidho Kanho Birsha University, Purulia-723149 West Bengal, India
Email: *tewarysunetra[at]gmail.com*

Abstract: ***Judicial systems globally face escalating backlogs, primarily driven by the time intensive nature of legal research and precedent analysis. This paper introduces the Juris Precedent Retrieval & Synthesis Model (JPRS Model), an LLM powered architecture designed to bridge the gap between historical judicial data and active case resolution. By extracting and synthesizing massive volumes of historical rulings into a structured legal data warehouse, the JPRS Model provides judges with a high fidelity 'decision support' framework. Our evaluation demonstrates that the model's core components the Precedent Summarization Module (PRE-SUM) and the Judicial Recommendation Engine (JUR-REC) significantly streamline preliminary case scrutiny, yielding a 38% reduction in review time. Ultimately, this framework ensures that judicial professionals can navigate decades of complex legal reasoning efficiently without sacrificing the refined context of historical rulings.***

Keywords: Large Language Models, Judicial Backlog, Legal Data Warehouse, Decision Support Systems, Precedent Retrieval, Artificial Intelligence in Law

1. Introduction

The prompt delivery of justice is a foundational pillar of any functional legal system. However, the exponential growth in litigation has created severe bottlenecks in courts worldwide. A significant contributor to this delay is the extensive time required for judges and clerks to review past judgments on similar matters, analyse the associated evidence, and synthesize complex, multi decade legal doctrines.

To address this challenge, we present the *JPRS Model (Juris Precedent Retrieval & Synthesis Model)*. This architecture utilizes Large Language Models (LLMs) to transform unstructured court documents into a highly organized Legal Data Warehouse. By deploying specialized modules to summarize lengthy judgments and recommend highly context relevant precedents. The JPRS Model acts as a high fidelity decision support system, mitigating the "search and verify" loop that causes judicial delays while maintaining the human in the loop necessity of legal adjudication.

2. Related Work

The integration of LLMs into the legal domain has seen rapid development, transitioning from basic natural language processing to complex legal reasoning and judgment prediction. However, recent studies highlight significant limitations in allowing AI to act autonomously in judicial decision making.

Research by Xie et al. (2026) emphasizes that integrating LLMs into judicial systems requires rigorous safety examinations against non-legal influences. Their stress tests on LLM generated labour dispute outcomes revealed substantial safety vulnerabilities, noting that models exhibit inherent deviations from real court rulings and are highly susceptible to public opinion and social media sentiment. These biases often lead to unstable and inflated compensation predictions, particularly affecting vulnerable populations in low skilled occupational categories.

Conversely, Posner and Saran (2025) explored how LLMs compare to human judges and law students in formal legal reasoning. By replicating a judicial experiment, they found that LLMs (specifically GPT-4o) act as strict formalists. Unlike human judges who were found to be influenced by whether a defendant was sympathetically portrayed the LLM was strongly affected by legal precedent and the statute itself, largely ignoring the sympathetic nature of the defendant and avoiding broader policy considerations. The LLM followed precedent more consistently than human judges, demonstrating a higher likelihood of affirming a conviction when precedent supported it, and reversing when precedent dictated reversal.

Our JPRS Model diverges from these autonomous judgment generation models by explicitly functioning as a *retrieval and summarization* system rather than a predictive judge. By focusing on surfacing relevant historical data (evidence, laws, and verdicts) rather than dictating the final outcome, the JPRS Model provides the formalist efficiency observed by Posner and Saran without exposing the judicial process to the unpredictable societal biases and hallucinations identified by Xie et al.

3. Methodology

The JPRS Model pipeline relies on two primary LLM driven components designed to ingest, process, store, and retrieve legal documents efficiently.

3.1 PRE-SUM (Precedent Summarization Module)

The PRE-SUM module is tasked with distilling multi page Supreme Court and High Court judgments into actionable briefs. To systematically store these previous judgments in the Data Warehouse, the module utilizes an LLM powered embedding framework.

Each historical case (C_h) is vectorized based on four distinct parameters captured by the LLM: the core issues of the case (I), the probabilities derived from the available evidence (P_e), the relevant laws of the country (L), and the significance of the judge's verdict in relation to those laws and evidence (S).

We define the mathematical formula to store and retrieve these cases as a multidimensional similarity score. When a judge inputs a query for a new active case (C_q), the system calculates the relevance score against historical cases using the following function:

$$\text{Score}\ (C_q,C_h) = w_1 \cdot \cos(I_q,I_h) + w_2 \cdot (1-|P_{e,q} - P_{e,h}|) + w_3 \cdot \text{Jaccard}(L_q, L_h) + w_4 \cdot S_h$$

Where w_1, w_2, w_3, w_4 are dynamically adjusted weights based on the specific legal domain (e.g., constitutional vs. criminal law). This formula ensures that the Data Warehouse indexes cases not just by keyword, but by the mathematical alignment of their evidentiary probabilities and core legal issues.

3.2 JUR-REC (Judicial Recommendation Engine)

Working in tandem with PRE-SUM, the *JUR-REC* module acts as the front-end interface for the presiding judge. When a new case is docketed, JUR-REC leverages the similarity formula to suggest previous cases. It outputs a structured recommendation featuring the historical case's core evidence, the final judgment, and the specific application of the country's law. This provides immediate, highly contextualized decision support tailored to the current case parameters.

4. Results and Discussion

4.1 Experimental Setup

We evaluated the *JPRS Model* using a corpus of constitutional amendment judgments sourced from the eCourts (NJDG) repository. These documents are characterized by their high volume, complex legal reasoning (often spanning decades of precedent), and dense technical language. Our evaluation focuses on the performance of two modules: *PRE-SUM* (Precedent Summarisation Module) and *JUR-REC* (Judicial Recommendation Module).

4.2 Performance of PRE-SUM (Summarization)

The *PRE-SUM* module was tasked with distilling multi page Supreme Court and High Court judgments into actionable briefs. We evaluated this using *BERT Score* and *Legal Intent F1 Score* (to ensure legal terminology and constitutional principles were preserved).

Metric	PRE-SUM Performance	Traditional Keyword Search
Summary Fidelity (BERTScore)	0.93	N/A
Legal Intent Coverage (F1)	89%	62%
Average Processing Time	35 seconds	120+ minutes

Analysis: *PRE-SUM* effectively captured the Ratio Decidendi of constitutional amendments, filtering out redundant procedural noise. Our intent based evaluation confirms that the summaries successfully retained the "Basic Structure" doctrine arguments, which are frequently central to these specific cases.

4.3 Performance of JUR-REC (Recommendation)

The *JUR-REC* module provides the presiding judge with contextually relevant precedents for new cases. We measured performance using *Precision@K* and *MRR (Mean Reciprocal Rank)*.

1) **Precision@3:** The system achieved a **0.87** score, meaning the top 3 recommended cases were considered highly relevant by legal experts for 87% of queries.
2) **Contextual Alignment:** Unlike traditional search, which retrieves cases based on broad keywords, *JUR-REC* demonstrated a superior ability to identify "amendment specific" precedents, effectively linking current constitutional challenges to historical cases in the eCourts database.

4.4 Impact on Judicial Efficiency

By integrating *PRE-SUM* and *JUR-REC*, the JPRS model addresses the information bottleneck in constitutional litigation:

1) **Reduced Cognitive Load:** Judges can review the AI generated summaries provided by *PRE-SUM* in minutes, as opposed to manually navigating thousands of pages of historical eCourts records.
2) **Backlog Mitigation:** Simulation results indicate a **38% reduction** in the time required for preliminary case scrutiny. By surfacing highly relevant precedents through *JUR-REC*, the system minimizes the "search and verify" loop that contributes to judicial delays.

4.5 Discussion on Constitutional Data

A unique challenge in constitutional amendment cases is the evolution of judicial interpretation over time. The *JPRS Model* proved robust in handling this by prioritizing chronologically significant precedents, a feature that allows judges to see not just what the law was, but how its interpretation has evolved. While the system shows occasional sensitivity to highly niche legislative amendments, it remains a highly effective decision support tool for standardized constitutional review.

5. Limitations

A self-reflective analysis of the model's performance reveals certain constraints. While the JPRS Model excels at processing broad constitutional doctrines, the LLM architecture exhibits occasional sensitivity to highly niche legislative amendments where historical training data is sparse. Furthermore, because LLMs inherently possess a formalistic tendency to strictly adhere to the text of the statute and past precedents, the model may occasionally filter out deeply nuanced, case specific human elements that a judge might otherwise consider relevant for equitable relief. Finally, processing extremely lengthy, multi decade rulings sometimes brush against the token context limits of underlying foundational models, necessitating the aggressive chunking of text which can occasionally lead to a minor loss of broader narrative context.

6. Ethical Considerations

Deploying AI in the legal sphere demands rigorous ethical guardrails.

- **Addressing Bias:** LLMs are susceptible to absorbing and amplifying societal biases present in their training data. External pressures, such as public opinion on emotionally charged topics, have been shown to distort LLM generated legal outcomes, particularly regarding vulnerable or low skilled populations.
- **Data Safety:** To ensure judicial integrity, the JPRS Model operates within a closed loop Data Warehouse, meaning it does not pull unverified data from the open internet, thereby shielding the recommendation engine from real-time social media sentiment or public pressures.
- **Human in the Loop:** Most importantly, the JPRS Model is strictly a *decision support* tool, not an autonomous adjudicator. It is designed to assist, not replace, the presiding judge. Maintaining a human in the loop ensures that the rigid formalism of the AI is balanced by human equity, empathy, and contextual understanding.

7. Conclusion

The proposed model advances legal AI by bridging the gap between LLM reasoning and formal jurisprudence. By codifying case issues, evidence-based probabilities, and legal outcomes into a centralized data warehouse, the system serves as a trustworthy repository of judicial logic. This repository acts as an essential decision support mechanism, empowering judges to navigate complex legal scenarios with greater precision and reliance on established legal frameworks.

References

[1] Xie, Y., He, Y., Yang, X., Gai, X., & Hui, P. (2026). *LLM Safety in Judicial AI: A Stress Test of Social Media Influence on Real-World Judgments*. Proceedings of the Fortieth AAAI Conference on Artificial Intelligence (AAAI-26). Association for the Advancement of Artificial Intelligence.

[2] Posner, E. A., & Saran, S. (2025). *Judge AI: Assessing Large Language Models in Judicial Decision-Making*. University of Chicago Law School, Coase-Sandor Institute for Law & Economics Research Paper No. 25-03.

[3] Han, S., Takashima, Y., Shen, S. Z., Liu, Y., Thuo, R. K., Knowlton, S., Piskac, R., Shapiro, S., Cohan, A., & Liu, C. (2025). COURTREASONER: Can LLM Agents Reason Like Judges? *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 35291-35306.

[4] Linze, C., Yufan, C., Zhe, H., & Song, D. J. (2026). Towards Trustworthy Legal AI through LLM Agents and Formal Reasoning. *arXiv preprint arXiv:2511.21033v2*.

[5] Supreme Court of India. (n.d.). National Judicial Data Grid. https://njdg.ecourts.gov.in/