International Journal of Computational Research in Science and Technology
E-ISSN: 3108-0871, Volume 2 Issue 5
Sep-Oct 2026, PP 01-6

AI-Driven Intelligent Judicial Case Management System: A Review

Parth Vishwas Kale, Prof. Bangar A.P., Prof. Bhosale S.B., Dr. A. A. Khatri

Dept. of Computer Engineering,

Jaihind collage of Engineering, Pune,India

Abstract. Courts in many jurisdictions, and in India in particular, face a persistent backlog that manual scheduling and file handling cannot clear. This paper reviews five recent studies that together cover the main building blocks of an AI-driven intelligent judicial case management system: rule-based case prioritisation, machine-learning prediction of case delay and duration, large-language-model (LLM) based judgment prediction, AI-assisted trial workflows for criminal cases, and public perception of automation at different stages of adjudication. The studies are compared on objective, data, method, reported results and stated limitations. On this basis a six-layer conceptual architecture is outlined, with explainability and human oversight as core layers. The review suggests that (i) prioritisation and delay prediction offer the most immediate and least contentious gains, (ii) outcome-prediction accuracies are promising but are not directly comparable across studies and are not sufficient for deployment, and (iii) public acceptance depends on keeping the judge in control of the later stages of decision-making. Open challenges include data quality, bias, explainability, legal regulation and evaluation in real courts.

Keywords: Judicial case management, artificial intelligence, case prioritisation, delay prediction, large language models, explainable AI, smart courts.

I. Introduction

Pendency is one of the central problems of the Indian justice system. The lower courts alone hold about 44.9 million pending cases, roughly 34 million of them criminal, and pendency has been growing by nearly 18% a year in spite of the eCourts project and the National Judicial Data Grid (NJDG) [1]. Much of the day-to-day management of this load is still manual. In one representative description, court staff schedule and prioritise cases from their own experience and subjective judgement, which wastes time and produces inconsistent treatment of urgent matters [2].

Artificial intelligence (AI) is now being applied to several parts of this problem. Machine learning (ML), natural language processing (NLP) and large language models (LLMs) have been used to rank cases by urgency [2], to forecast delay and duration [1], to predict judgments [3] and to speed up criminal trials [4]. A separate line of work asks how people react when algorithms take part in adjudication, using the four stages of decision-making described by Parasuraman et al. [6] as a lens [5]. These studies are scattered: each addresses one function of the court, and none describes how the pieces would fit into a single case management system.

This paper reviews five such studies [1]–[5] and draws them together under the heading of an AI-driven intelligent judicial case management system. The contributions are: (1) a structured comparison of the studies by data, method, result and limitation; (2) a six-layer conceptual architecture that places each technique in a common system; and (3) a discussion of the gaps that must be closed before such a system can be used in practice. The review is limited to these five sources and should be read as a focused survey, not an exhaustive one.

Background

Judicial Case Management

Case management covers the life of a case from filing to disposal: registration, scrutiny, listing and scheduling, hearings, judgment and follow-up. Two terms recur in the literature. Pendency means cases that have been filed and not resolved, whereas delay means cases that stay unresolved beyond a reasonable time; the two-year threshold suggested by the Law Commission of India is used in [1]. Smart courts move parts of this process online. The work in [3] describes, for example, a full-process intelligent trial system developed by the Zhejiang Higher People's Court with Zhejiang University and Alibaba, which extracts information from legal documents, summarises debates, verifies evidence, retrieves similar cases and generates judgment documents.

AI Techniques Used

The reviewed studies use techniques of increasing complexity. A lookup table of statutory sections with a simple scoring rule is used for prioritisation [2]. Tree-based ensembles such as random forests and gradient boosting machines, selected through automated ML (AutoML) and explained with SHAP, partial dependence plots (PDP) and individual conditional expectation (ICE) plots, are used for delay and duration [1]. Decision-tree regression, support vector machines (SVM), LSTM networks and transformers are used for trial time and document analysis [4]. Pre-trained GloVe embeddings and transformer encoders (BERT, ALBERT, RoBERTa and DistilBERT) are fused for judgment prediction [3].

Stages of Decision-Making

Parasuraman et al. [6] divide decision-making into information acquisition, information analysis, decision selection and decision implementation, and allow a different level of automation at each stage. The survey in [5] adopts this scheme for courts. It is also a convenient way to map system modules: filing-stage data collection and retrieval belong to the first stage, scoring and prediction to the second, recommendations to the third, and scheduling and orders to the fourth.

II. Review of Existing Work

1. Case Prioritisation

Mali et al. [2] propose a Judicial Case Priority Management System for Indian courts. Each case carries a list of Indian Penal Code (IPC) sections, and a lookup table assigns a preset priority value to every section. For a case with a single section, the case priority is that section's value; an unrecognised section receives an infinite value, which is the lowest priority. For several sections, the priority is the smaller of the weighted average and the minimum of the section values (lower values mean higher priority):

$$P_{case} = \min(P_{avg}, P_{high}) \quad (1)$$

where Pavg is the weighted average priority and Phigh is the lowest value among the sections [2]. Scores run from 0 (urgent or high-profile) to 10 (less urgent), and cases are sorted in ascending order of score. The system has six modules: an input interface, a case management module, a relational database (case records, IPC sections, user accounts, case status), a priority calculation module, a scheduler and an output module that announces priorities through the court's public address system [2]. The design is simple, deterministic and easy to explain, which suits a first deployment. Its weaknesses are also clear. The score depends only on statute sections; case age, time already spent pending, stage of the case and the situation of the parties are not used. The weights are not disclosed, and the paper reports no quantitative evaluation. Because the table is keyed to IPC sections, it also needs updating after the recent revision of Indian criminal law, a point raised separately in [1].

2. Delay and Duration Prediction

Bhatnagar and Huchhanavar [1] use an AutoML framework (H2O) on more than 22.4 million criminal cases from Indian lower courts, drawn from the 79-million-case e-courts dataset of Ash et al. cited in [1]. Only information available when the case is filed is used (district, section, case type, judge position and similar fields), so that a prediction can be made at the earliest point.

A classification model labels a case as on-time or delayed (more than two years); regression models estimate duration in months.

Replacing one regression model with two, one for on-time and one for delayed cases, lowered the RMSE from 8.33 to 4.90 months for on-time cases and from 23.32 to 13.79 months for delayed cases. A separate classifier reached 80% accuracy with a weighted F1 score of 84.1% [1]. SHAP, PDP and ICE plots show that district, section and case type are the most influential features, while religion-based features and advocate gender matter least. The authors also test hybrid classification-regression routing. Limits stated in [1] include the restriction to criminal cases, gender and religion fields that were estimated by ML classifiers from names, and the need to refresh the data for revised criminal laws. For delayed cases an error of about 13.8 months remains large, so the outputs are indicative rather than exact.

3. Judgment Prediction with LLMs

Abdel Samee et al. [3] predict whether the European Court of Human Rights (ECtHR) finds a violation or no violation. Their Cross-Domain Neural Knowledge Fusion (CDKF) system combines four GloVe embeddings (50, 100, 200 and 300 dimensions) with four transformer encoders (BERT, ALBERT, RoBERTa, DistilBERT) and fuses the features before classification. Over all articles, BERT gave the highest accuracy (82.14%), followed by RoBERTa (78.57%) and CDKF (76.28%); the transformers outperformed the static embeddings [3]. The abstract of [3] states an accuracy of around 83%, whereas the conclusion reports the figures above, so the conclusion values are used here. The authors argue that CDKF, although slightly less accurate, is more confident because of its ensemble design, and that BERT and RoBERTa may fluctuate with the data.

Earlier work summarised in [3] reports accuracies of roughly 69–82% for ECtHR and US Supreme Court prediction, so the new results are in the same range. Two cautions apply. The task is binary classification on a foreign dataset, not Indian case management, and the study gives limited evidence on how well it would transfer to Indian courts.

4. AI-Assisted Trial Efficiency

Chen [4] evaluates AI tools on 500 criminal case records of minor, ordinary and complex cases. The tools are an AI case management system that monitors progress and predicts trial duration, a document generation platform and an evidence analysis tool. Decision-tree regression and SVM predict trial time and verdict accuracy, while LSTM and transformer models process text. Data were split 80/20 and five-fold cross-validation was used. Trial efficiency is expressed as the percentage saving in time:

$$E = (T_c - T_{ai}) / T_c \times 100\% \quad (2)$$

where Tc and Tai are the mean trial times without and with AI [4]. AI assistance is reported to cut trial time by 40% and misjudgment rate by 55%, with $p < 0.05$ in a t-test; time savings of 37.5% and 37.8% are reported for civil and administrative cases. The author states that these results are preliminary because of the limited dataset and case types, and that scalability, bias and different legal systems remain open [4]. The study also stresses that AI should assist judges and not replace them.

5. Perceived Fairness of Automation

Barysė and Sarel [5] ask how fair people think a court outcome is when algorithms assist at each of the four stages in [6]. In an exploratory survey of Lithuanian respondents, including legal professionals, an intermediate level of automation was most often considered fairest only for information acquisition (40.15% of responses); for analysis, selection and implementation, low levels were preferred. Legal professionals judged automation in the implementation stage as less fair than other respondents did [5]. The result is a useful design rule: automation is better accepted in early, information-handling stages than in the stages that decide and enforce. The sample is a convenience sample and the study is exploratory, so the findings are suggestive only.

International Journal of Computational Research in Science and Technology
E-ISSN: 3108-0871, Volume 2 Issue 5
Sep-Oct 2026, PP 01-6

Table 1: Comparative Summary of the Reviewed Studies

Ref.	Focus	Method and Data	Key Reported Result	Stated / Observed Limitation
[2]	Case prioritisation	IPC-section lookup with weighted-average / minimum rule; relational database	Priority score 0–10 per case; cases sorted by score; no quantitative evaluation reported	Static table; no empirical validation; IPC sections need updating
[1]	Delay and duration prediction	AutoML (random forest, GBM) with SHAP; 22.4 million criminal cases; filing-time features	Delay classifier 80% accuracy, weighted F1 84.1%; RMSE 8.33→4.90 months (on-time), 23.32→13.79 months (delayed)	Criminal cases only; gender and religion estimated by ML; large error for delayed cases
[3]	Judgment outcome prediction	CDKF fusion of GloVe and BERT, ALBERT, RoBERTa, DistilBERT; ECtHR data	BERT 82.14%, RoBERTa 78.57%, CDKF 76.28% (violation vs. no violation)	Binary task on European data; accuracy may fluctuate with data
[4]	Trial efficiency	Decision-tree regression, SVM, LSTM / transformers; 500 criminal case records	Trial time −40%, error rate −55% ($p < 0.05$)	Small sample; preliminary; scalability and bias unresolved
[5]	Perceived fairness	Survey using four decision stages of [6]; Lithuanian respondents	Intermediate automation preferred only for information acquisition	Convenience sample; exploratory

IV. Conceptual Architecture

The reviewed work covers separate functions. Putting them together gives the six-layer architecture of Fig. 1. This architecture is the author's synthesis for the purpose of this review; none of the five papers proposes it as such.

- Data and NLP Layers: The data layer gathers filing details, statute sections, orders, judgments and hearing history, for example from the eCourts platform and NJDG used in [1]. The NLP layer cleans text, extracts legal entities and produces embeddings with transformer encoders, as in [3] and [4]. Records should be anonymised before use, as in [4].
- Intelligence Layer: Three modules sit here: a priority scorer in the spirit of [2], a delay and duration predictor as in [1], and an outcome or precedent-support module as in [3] and [4]. Priority could combine the statute-based score of [2] with predicted delay from [1], so that cases at risk of becoming delayed are raised early. The outcome module should support the judge with similar cases and not issue verdicts.
- Explainability Layer: Every score or prediction is shown with its main drivers, using SHAP, PDP and ICE as in [1], and a periodic audit checks error rates across case types, districts and party groups. The study in [4] also reports checking bias across case types.
- Case Management and Oversight Layers: The case management layer holds the scheduler, cause list, alerts, status tracking and role-based access described in [2]. The oversight layer keeps the judge or registrar in control: outputs are advisory, can be overridden, and are logged. This follows the conclusion of [4] that AI should assist and not replace judges, and the finding of [5] that people accept automation more readily in information-handling stages than in decision and implementation stages.

International Journal of Statistics and Mathematical Research Trends
E-ISSN: 3108-0871, Volume 2 Issue 5
Sep-Oct 2026, PP 01-6

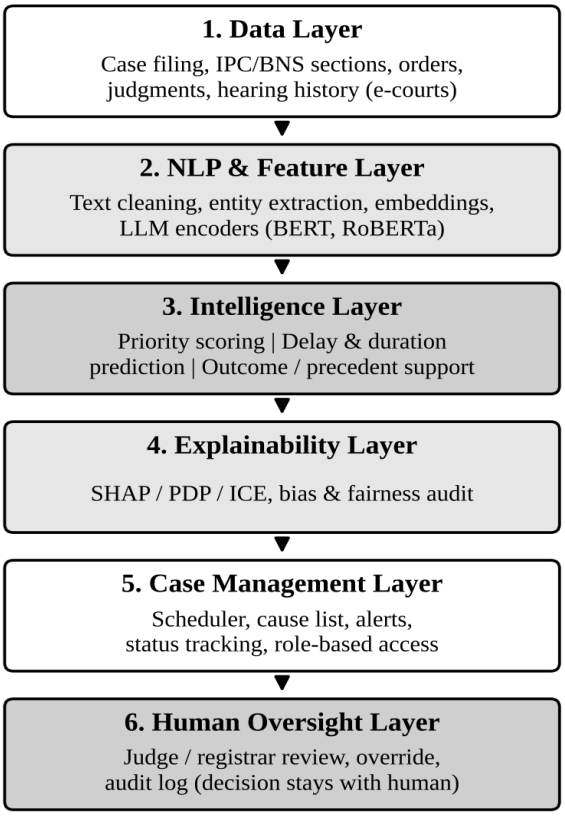


Fig.1. Conceptual Architecture of an AI-Driven Judicial Case Management System

Challenges and Research Gaps

- Data Quality and Coverage: Predictions depend on complete and consistent records. The study in [1] notes missing values and unclear labels in its data, and [4] depends on case data of sufficient quality and standard form. Results from a European dataset [3] or a 500-case sample [4] cannot be assumed to hold for Indian courts without testing.
- Bias and Fairness: Models can inherit bias from training data. In [1], gender and religion were estimated by ML classifiers, and the authors say that this needs to be reviewed for bias. Any deployed system needs routine fairness audits, and protected attributes should be excluded or tested with care.
- Explainability and Trust: Judges and litigants need to see why a case was ranked or predicted as it was. The XAI tools in [1] show one workable approach. The authors of [3] also stress interpretability. The public attitudes reported in [5] suggest that opaque automation lowers perceived fairness.
- Legal and Ethical Framework: Chen [4] calls for clear rules on the scope of AI use, protection of the independence of judges and the rights of parties, and training for court staff. Static rules such as the IPC-based table in [2] must also follow changes in the law.
- Evaluation and Comparability: Accuracy values in Table I use different tasks, data and metrics, so they cannot be ranked against each other. The prioritisation system [2] has no reported evaluation at all, and [4] calls its own results preliminary. Studies on real Indian court data, with common metrics and ideally a pilot, are needed.

V. Conclusion

This review examined five studies relevant to an AI-driven intelligent judicial case management system. Together they show that the main components are feasible: statute-based prioritisation [2], filing-stage delay and duration prediction [1], LLM-

based outcome prediction [3] and AI-assisted trial workflows [4]. They also show that how and where AI is used matters for acceptance [5]. A layered architecture with explainability and human oversight is proposed as a way to combine these components.

Future work could (i) combine statute-based priority with learned delay prediction and test it on pending Indian cases, (ii) extend delay models to civil cases and to more features from the life of a case, as suggested in [1], (iii) use legal-domain LLMs for Indian-language judgments, (iv) run pilot studies in real courts with common metrics, and (v) study the legal and ethical framework needed for deployment. Because this review covers only five studies, a wider literature search should be done before drawing firm conclusions.

Acknowledgment
The authors thank Jaihind College of Engineering & Prof. Bangar sir for their support and guidance during this work.

References

1. M. Bhatnagar and S. Huchhanavar, “Hybrid machine learning modelling with explainability for predicting case delays and durations in Indian lower courts,” J. Big Data, vol. 13, art. 4, 2026, doi: 10.1186/s40537-025-01340-1.
2. N. Mali, T. Jagdale, S. Deokar, G. Gujar, and S. Badgujar, “The Judicial Case Priority Management System,” Int. J. Adv. Res. Comput. Commun. Eng. (IJARCCE), vol. 14, no. 6, pp. 213–217, Jun. 2025, doi: 10.17148/IJARCCE.2025.14640.
3. N. Abdel Samee, M. Alabdulhafith, S. M. A. H. Shah, and A. Rizwan, “JusticeAI: A large language models inspired collaborative and cross-domain multimodal system for automatic judicial rulings in smart courts,” IEEE Access, vol. 12, pp. 173091–173107, 2024, doi: 10.1109/ACCESS.2024.3491775.
4. Q. Chen, “Improving the trial efficiency of criminal cases with the assistance of artificial intelligence,” Discover Artif. Intell., vol. 5, art. 110, 2025, doi: 10.1007/s44163-025-00353-2.
5. D. Barysė and R. Sarel, “Algorithms in the court: does it matter which part of the judicial decision-making is automated?,” Artif. Intell. Law, vol. 32, pp. 117–146, 2024, doi: 10.1007/s10506-022-09343-6.
6. R. Parasuraman, T. B. Sheridan, and C. D. Wickens, “A model for types and levels of human interaction with automation,” IEEE Trans. Syst., Man, Cybern. A, vol. 30, no. 3, pp. 286–297, May 2000 (cited as adopted in [5]).