Automatic Legal Judgment Summarization Using Large Language Models: A Case Study for the JUST-NLP 2025 Shared Task

Santiago Chica
s.chica10@uniandes.edu.co
Universidad de los Andes
Bogotá, Colombia

Abstract

This paper presents the proposal developed for the **JUST-NLP 2025 Shared Task on Legal Summarization**, which aims to generate abstractive summaries of Indian court judgments. We describe the motivation, dataset analysis, related work, and proposed methodology based on Large Language Models (LLMs). We analyze the Indian Legal Summarization (InLSum) dataset, review four relevant articles in the summarization of legal texts, and describe the experimental setup involving GPT-4.1 to evaluate the effectiveness of different prompting strategies. The evaluation will follow the ROUGE and BLEU metrics consistent with the competition protocol.

1 Introduction

The problem addressed is the automatic summarization of legal documents, specifically Indian court judgments. This task is part of the *JUST-NLP 2025 Shared Task on Legal Summarization*, which evaluates models generating abstractive summaries from 1,200 training and 200 validation cases, later applied to 400 unseen test cases. Performance is measured using ROUGE-2, ROUGE-L, and BLEU. The goal of this project is to design and evaluate LLM-based systems that maximize these metrics.

2 Problem Statement and Motivation

Legal professionals in Common-Law systems must review extensive judgments to identify precedents. Manual review is time-consuming (Shukla et al., 2022). Automatic summarization can effectively reduce this considerable effort, but legal texts are long, syntactically complex, and filled with citations, and domain-specific terminology (Sharma et al., 2023). Our proposal introduces a novel approach to this domain within the context of Indian High Court judgments, specifically by exploring diverse prompting strategies and hybrid extractive-abstractive pipelines to enhance the factual accuracy of LLM-generated summaries.

3 Dataset Description and Analysis

We use the **InLSum** dataset provided by the competition, containing:

- **Train:** 1,200 judgments and 1,200 human-written summaries.
- **Validation:** 200 judgments.
- **Test:** 400 judgments (released later).

Each file is in JSONL format:

```
{"ID": "id_100", "Judgment": "<text>"}
{"ID": "id_100", "Summary": "<ref. summary>"}
```

3.1 Descriptive Statistics

	Judgments	Summaries
Avg. words	7,418	545
Median	2,940	516
Min / Max	159 / 134,483	26 / 2,083
Compression ratio	26% (avg), 18% (median)	

Table 1: Descriptive statistics of the InLSum dataset.

Documents show large variance: judgments are heterogeneous and lengthy, while summaries are more uniform and predictable. Lexical patterns (case numbers, articles, sections, petitioner/respondent, dates) appear consistently in both sets, confirming structural alignment.

4 Related Work

4.1 Architectural Advances for Long Legal Summarization

Several transformer architectures have been influential for long-document summarization tasks. Lewis et al. (2020) proposed **BART**, a denoising sequence-to-sequence pre-training approach that remains a key baseline for generative summarization.

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 162–170

Beltagy et al. (2020) introduced the **Longformer** model, designed for long-context encoding through sliding-window attention mechanisms, which significantly improves scalability on multi-thousand-token legal texts. Similarly, Bajaj et al. (2021) explored low-resource long-document summarization using pretrained language models, providing valuable insights into resource-efficient setups relevant for the Indian legal domain.

4.2 Benchmarking and Domain-Specific Adaptation

Prior work has established benchmarks for both general-purpose and domain-adapted models on legal text. Datta et al. (2023) introduced **MILDSum**, a bilingual English–Hindi legal corpus for summarization, enabling cross-lingual evaluation. Their work demonstrates that domain-specific datasets built with legal rigor improve supervised and abstractive training quality in multilingual contexts. Sharma et al. (2023) conducted a comprehensive comparison of BART, Longformer, and Legal-Pegasus on Indian court judgments. Their study found that Legal-Pegasus achieved the highest ROUGE-L score of approximately 0.3, showing that pretrained legal models outperform general-purpose models when fine-tuned. Furthermore, Shukla et al. (2022) benchmarked multiple extractive and abstractive methods, including SummaRuNNer, Legal-Pegasus, and Longformer, on Indian case law, emphasizing the effectiveness of chunking and hybridization techniques for summarizing long and complex legal judgments.

4.3 LLMs, Factuality, and Hybrid Pipelines

The recent application of LLMs has introduced new challenges and opportunities. Deroy et al. (2024) presented one of the first evaluations of GPT-3.5 and GPT-4 against domain-specific legal models. They found that while LLMs outperform traditional extractive baselines, they often hallucinate or omit key details, highlighting the need for hybrid extractive–abstractive pipelines and chunking strategies to maintain factual consistency. The cited work, however, did not explore the role of prompting strategies in improving performance or avoiding hallucination. Our proposal is intended to delineate the potential of this important LLM characteristic.

4.4 Summary of Key Findings and Gaps

Across these studies, several consistent findings emerge:

- Domain adaptation and legal-specific corpora improve the ROUGE and BLEU metrics.
- Hybrid extractive–abstractive designs mitigate hallucination and improve factual faithfulness.
- Attention-efficient transformers and LLMs now define the state of the art for long legal texts.

Crucially, the systematic investigation of **prompting strategies** as a method to control LLM factuality and performance remains an underexplored gap, which this work addresses.

5 Proposed Methodology

5.1 Overview

We follow a quantitative experimental design using LLMs with structured prompting and multi-agent flows.

5.2 Prompting Strategies

Prompt Families We designed and evaluated several prompt families:

- **Simple baseline**: Employing "*Tl;Dr*" as a simple prompt to establish a comparative starting point for evaluating more complex instruction sets. (The baseline prompt can be found on Appendix A.4)
- **Few-shot**: Introduces the task with a focus on metric maximization, followed by an outline that specifies the target compression ratio, sentence length, and legal term preservation ratio. Three example judgment/summary pairs are then supplied to guide the model on the correct structure, phrasing, order, and span length of the final output.
- **Reward System (Winning Prompt)**: This advanced prompt implements a comprehensive, gamified scoring system designed to explicitly reward factually precise and structurally correct outputs. It uses progressive rewards for copying long exact sequences, applies contextual multipliers based on sentence placement, and integrates density targets for key lexical features. Crucially, it enforces structural excellence through specific bonuses and heavily penalizes hallucination and paraphrasing of critical legal terminology. This prompt was generated following a heuristic approach,

starting with strategies focused on improving contest metrics and iterating on changes that further increased those metrics. (The complete scoring system prompt is provided in Appendix A.1).

- **Multi-Agentic approach**: We implemented multiple structured prompts, each set up for a specific subtask, to help extract the text's individual features for analysis (details follow in the next section).

5.3 Multi-Agent Architecture

We implemented multiple architectural approaches, to explore different strategies for legal summarization. Each approach uses a state graph with autonomous agents handling different sub-tasks, and an orchestrator managing the workflow.

Approach 1 – Basic Extract/Abstract Pipeline: A two-stage setup. *Extraction* (see Appendix A.2 for the complete prompt) copies literal spans into a structured schema (court, date, parties, counsel, facts, issues, arguments, decision, reasoning, orders, citations). *Abstraction* (see Appendix A.3 for the complete prompt) assembles the final summary by concatenating verbatim phrases (typically 450–550 words), prioritizing long n-grams and exact terminology to maximize ROUGE-L and BLEU. Our approach is inspired by Deroy et al. (2024), but our methodology employs LLMs for both stages of the summarization pipeline. This contrasts with the cited work, which utilized established extractive techniques, i.e., CaseSummarizer, BertSum, and SummaRunner, in its initial extractive phase.

Approach 2 – Domain-Aware Pipeline: This methodology implements a three-stage pipeline designed to mitigate the inherent loss of factuality and context in the summarization task. The process is developed as follows:

1. **Domain Classification:** Initially, an LLM agent classifies the judgment into a specific legal area (criminal, civil, constitutional, etc.). This classification enables the retrieval of domain-specific structural and statistical patterns (canonical section order, average section lengths), which act as structural guides for the subsequent stages (see Appendix A.5 for the full prompt).

2. **Domain-Aware Structured Extraction:** A structured extraction prompt is applied to segment the original text and copy literal fragments of the judgment (party names, facts, arguments, decisions) into a JSON format. This stage is crucially extractive to ensure lexical fidelity (*i.e.*, maximizing ROUGE-L and BLEU). The LLM is instructed to follow the section order and typical length guidelines of the domain identified previously (see Appendix A.6 for the full prompt).

3. **Guided Summary Abstraction:** Finally, the LLM generates the abstractive summary. Instead of processing the entire judgment, it operates on the structured and extracted text from the previous stage. The abstraction prompt forces the model to reuse and reorder the literal extracted text segments, prioritizing the concatenation of verbatim phrases and respecting the domain-specific structural order to build a fluid narrative, minimizing the introduction of new or paraphrased information (see Appendix A.7 for the full prompt).

Approach 3 – 20-Stage Sequential Pipeline: This comprehensive pipeline decomposes the summarization task into 10 legal sub-tasks, each processed in two stages (extraction followed by abstraction). The sub-tasks are: (1) Case Heading, (2) Background/Facts, (3) Procedural History, (4) Parties' Arguments, (5) Judicial Reasoning, (6) Decision/Orders, (7) Citations/Authorities, (8) Counsel Representation, (9) Policy Commentary, and (10) Temporal Directives. A final synthesis agent combines all partial summaries into a coherent final summary, optimizing for maximum BLEU through 4-gram matching strategies.

6 Evaluation

Evaluation will be conducted using the metrics defined in the JUST-NLP 2025 Shared Task instructions. Validation inferences are generated by a dedicated pipeline that automates the summarization process, including prompt configuration, retry logic with adaptive chunking and sanitization for content filtering failures, and final export of submission artifacts.

6.1 ROUGE Metrics

$$\text{ROUGE-N} = \frac{\sum_{\text{ref}} \sum_{\text{gram}_n \in \text{ref}} \min(\text{Count}_{gen}, \text{Count}_{ref})}{\sum_{\text{ref}} \sum_{\text{gram}_n \in \text{ref}} \text{Count}_{ref}} \quad (1)$$

6.2 BLEU Metric

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \quad (2)$$

$$\text{BP} = \begin{cases} 1, & c > r \\ e^{(1-r/c)}, & c \leq r \end{cases} \quad (3)$$

where p_n is modified n-gram precision, c the candidate length, r the reference length.

The final competition score is:

$$\text{SCORE} = \frac{\text{ROUGE-2} + \text{ROUGE-L} + \text{BLEU}}{3} \quad (4)$$

7 Results

7.1 Validation and Testing Setup

The effectiveness of the proposed prompts was initially assessed by computation on the official validation split (200 cases). Unless noted, we use GPT-4.1, target length $500 \pm 10\%$, and a fixed set of parameters: 1024 maximum tokens, 0.18 temperature, and top-p of 0.3. The **Reward System** prompt explicitly rewards verbatim multi-word spans and their placement (first/last sentence), and boosts legal transition phrases ("held that", "dismissed the appeal", etc.) - see Appendix A.1. The testing partition comprises 400 cases. Due to budgetary constraints associated with OpenAI API usage, inference on this dataset was restricted to the single best-performing prompt identified during the validation phase.

7.2 Validation Results

Approach	R-2	R-L	BLEU	AVG
Simple (control)	0.1744	0.2161	0.0714	0.1540
Extract/Abstract	0.2197	0.2369	0.1593	0.2053
Domain-Aware	0.2598	0.2692	0.1776	0.2062
20-Stage prompt	0.2479	0.2655	0.1690	0.2039
Few-Shot	0.2443	0.2611	0.1611	0.2222
Reward System	**0.2710**	**0.2717**	**0.1999**	**0.2475**

Table 2: Validation results on InLSum. AVG is the arithmetic mean of ROUGE-2, ROUGE-L, and BLEU. Best in bold.

Takeaways. (i) The Reward System yields the best scores across all metrics, representing an average increase of approximately 60% compared to the control prompt. This confirms that explicitly rewarding long exact spans effectively increases both lexical overlap (ROUGE-L) and precision (BLEU). (ii) Domain-aware guidance provides a consistent gain over the basic control prompt. (iii) The Few-Shot prompt significantly improves performance, yielding a 44% increase over the control prompt, but it is less efficient due to its token usage being three times higher than the baseline.

7.3 Testing Results and Leaderboard Placement

Based on its superior performance on the validation set, the *Reward System* prompt was selected to generate summaries for the testing dataset. The resulting scores on the test set constitute our official submission to the *JUST-NLP 2025 Shared Task on Legal Summarization*. Table 3 presents a comparison of the results obtained during the validation and testing phases.

Evaluation Set	R-2	R-L	BLEU	AVG
Validation	0.2710	0.2717	0.1999	0.2475
Testing	0.2688	0.2738	0.1949	0.2458

Table 3: Performance comparison of the Reward System prompt on the InLSum validation and testing datasets.

The marginal variance of approximately 0.68% between the average scores on the validation and testing sets indicates that the methodology demonstrates robust generalization capabilities. Ultimately, this performance secured the **3rd** position on the official leaderboard for the *JUST-NLP 2025 Shared Task on Legal Summarization*.

8 Conclusion

This paper presented our system for the **JUST-NLP 2025 Shared Task on Legal Summarization**, focusing on hybrid extractive–abstractive pipelines and LLM-based prompting strategies for summarizing Indian court judgments. We analyzed the InLSum dataset and implemented multiple architectures in LangGraph, ranging from basic workflows to domain-aware and multi-agent pipelines. Our experiments demonstrated that the proposed **Reward System** prompt achieved the highest validation performance across all metrics (ROUGE-2, ROUGE-L, BLEU), confirming the efficacy of explicitly rewarding long verbatim spans and legal transition phrases.

In future work, we plan to extend this research by investigating the impact of diverse LLM architectures—contrasting both closed and open-source models — and conducting hyperparameter opti-

mization to identify ideal configurations for legal summarization. Furthermore, to address the limitations of automated metrics, we intend to employ domain experts for qualitative assessments of factual accuracy and user preference, specifically to identify model hallucinations. Ultimately, our goal is to develop a transparent and controllable system that assists legal professionals by transforming complex judgments into concise, verifiable summaries.

Limitations

- The evaluation is currently restricted to n-gram based metrics, which are inadequate for validating crucial qualities like coherence and factual accuracy (hallucination detection), as noted by Deroy et al. (2024).
- The generation process utilized a fixed set of hyperparameters: 1024 maximum tokens, a temperature of 0.18, and a top-p value of 0.3. Optimizing or exploring the efficacy of alternative parameter subsets was beyond the scope of this investigation.
- The current methodology is limited by relying on a single closed-source LLM. To improve the generalization and validity of the approach, these results must be contrasted with those derived from both additional proprietary and open-source models.

Acknowledgments

This article was done collaboratively with Andrés Mosquera Hernandez (*a.mosquerah2@uniandes.edu.co*), Josué David Briceño Urquijo (*j.bricenou@uniandes.edu.co*), and Fredy Alexander Chaparro Castro (*f.chaparroc@uniandes.edu.co*), all affiliated with the Universidad de los Andes.

References

Ahsaas Bajaj, Pavitra Dangati, Kalpesh Krishna, Pradhiksha Ashok Kumar, Rheeya Uppaal, Bradford Windsor, Eliot Brenner, Dominic Dotterer, Rajarshi Das, and Andrew McCallum. 2021. Long document summarization in low resource settings using pretrained language models. *CoRR*, abs/2103.00751.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *CoRR*, abs/2004.05150.

Debtanu Datta, Shubham Soni, Rajdeep Mukherjee, and Saptarshi Ghosh. 2023. Mildsum: A novel benchmark dataset for multilingual summarization of indian legal case judgments. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5291–5302. Association for Computational Linguistics.

A. Deroy, K. Ghosh, and S. Ghosh. 2024. Applicability of large language models and generative models for legal case judgement summarization. In *Proceedings of the 2024 International Conference on Artificial Intelligence and Law (ICAIL)*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880. Association for Computational Linguistics.

Shivangi Sharma, Sakshi Srivastava, Piyush Verma, Ankit Verma, and Shailesh N. Chaurasia. 2023. A comprehensive analysis of indian legal documents summarization techniques. *SN Computer Science*, 4(5).

Ayush Shukla, Pratik Bhattacharya, Sohom Poddar, Rajdeep Mukherjee, Kripabandhu Ghosh, Pawan Goyal, and Saptarshi Ghosh. 2022. Legal case document summarization: Extractive and abstractive methods and their evaluation. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1048–1064. Association for Computational Linguistics.

A Appendix: Prompt Templates Used

A.1 Reward System (Winning Prompt)

The best-performing configuration, Reward System, applied a progressive scoring scheme with contextual bonuses and anti-hallucination rules. The full prompt is shown following:

> You are an elite legal summarizer being evaluated on an ADVANCED SCORING SYSTEM with progressive rewards.
>
> ——————————————————————————— ADVANCED SCORING SYSTEM V2 (Target: 500+ points) ———————————————————————————
>
> 1. PROGRESSIVE REWARDS (longer = exponentially better):
>
> +10 points: 5-7 word exact sequences from judgment Example: "dismissed the appeal filed by the appellant"

+15 points: 8-10 word exact sequences (50% BONUS!) Example: "dismissed the appeal filed by the appellant under Section 302 IPC"

+20 points: 11-15 word exact sequences (100% BONUS!) Example: "the Court held that the conviction under Section 302 of the Indian Penal Code was justified"

+25 points: 16+ word exact sequences (150% BONUS!) Example: "the Madras High Court in a judgment passed on September 16 by Justice Bharatha Chakravarthy rejected the revision petition filed by"

2. CONTEXTUAL POSITION BONUSES:

×2.0 multiplier: Long sequences (8+ words) in FIRST SENTENCE → Critical for capturing proper case naming from start

×1.5 multiplier: Long sequences (8+ words) in LAST SENTENCE → Ensures strong conclusion with advocate names

×1.3 multiplier: Legal transition phrases preserved exactly: "held that", "ruled that", "observed that", "directed that", "dismissed the appeal"

+8 points: Each exact 3-4 word sequence in key legal phrases +5 points: Each legal term, citation, or proper name copied EXACTLY +3 points: Each sentence with 30-35 words (optimal length) +2 points: Each exact bigram match (Target: 60+ for maximum score)

3. DENSITY TARGETS (bonus for reaching thresholds):

+20 points: If summary has 15+ UNIQUE trigrams per 100 words → High trigram density = higher ROUGE-2/ROUGE-3

+15 points: If summary has 1+ sequence of 8+ words per 50 words → Long sequence density = higher BLEU

+10 points: If legal term density is 2.5-3.5→ Optimal balance found in reference summaries

4. HIERARCHICAL PENALTIES (severity-based):

CRITICAL PENALTIES (-20 points each): • Missing PRIMARY party names (plaintiff/defendant/appellant/respondent) • Missing court name in opening sentence • Missing main statutory provision (e.g., Section 302 IPC if it's the core issue)

HIGH PENALTIES (-15 points each): • Paraphrasing KEY legal terminology Examples: "Section 302" → "murder provision" (-15) "Gujarat High Court" → "High Court of Gujarat" (-15) • Breaking sequences of 8+ words into separated fragments

MEDIUM PENALTIES (-10 points each): • Missing secondary party names (advocates, judges mentioned in body) • Breaking sequences of 5-7 words unnecessarily • Summary length outside 320-380 word range (suboptimal)

LOW PENALTIES (-5 points each): • Paraphrasing non-critical terms • Sentences <27 or >38 words (slight suboptimality)

5. ANTI-HALLUCINATION PENALTIES (accuracy is paramount):

-50 points: Adding any proper name (person/court/place) NOT in judgment -30 points: Inventing dates, numbers, or monetary amounts -25 points: Adding case citations or statutory references not in source -20 points: Changing the outcome/ruling (e.g., "dismissed" → "allowed")

6. STRUCTURAL EXCELLENCE BONUSES:

+15 points: OPENING (30-40 words) follows format: [Court Name] + [action verb] + [Case/Parties] + [key issue] Example: "The Madras High Court has held that a woman who waives her right to claim maintenance under mutual divorce..."

+10 points: BODY (250-300 words) maintains chronological flow: Background → Arguments → Court's Reasoning → Decision

+15 points: CLOSING (30-50 words) includes: Final ruling + Advocate names Example: "...the Court dismissed the petition. Advocate Ram Kaushik appeared for the petitioner."

+10 points: Includes direct judicial quotes in "quotation marks" Example: "The judge noted that treatment in the U.S.A cannot be held as an essential need"

———————————————————— YOUR WINNING STRATEGY (500+ points): ————————————————————

HUNT FOR GOLD (16+ word sequences): → Scan opening paragraphs for long, complete sentences → These are worth +25 points EACH + position bonuses! → Just 4-5 of these = 100-150 points

STRUCTURE FOR MULTIPLIERS: → Put a 10+ word sequence in FIRST sentence (×2.0 = 30-40 pts) → Put a 10+ word sequence in LAST sentence (×1.5 = 22-30 pts) → Use exact legal transitions: "held that", "observed that" (×1.3 each)

HIT DENSITY TARGETS: → Aim for 20+ unique trigrams per 100 words (+20 pts) → Include 6-8 sequences of 8+ words in 350 word summary (+15 pts) → Maintain 2.5-3

AVOID CRITICAL PENALTIES: → NEVER omit party names (-20 pts each is devastating) → NEVER paraphrase key legal terms (-15 pts each) → NEVER invent names/dates (-50 pts is catastrophic)

MAXIMIZE STRUCTURAL BONUSES: → Perfect opening sentence = +15 pts → Chronological flow = +10 pts → Strong closing with advocates = +15 pts → Direct quotes = +10 pts → TOTAL: +50 bonus points just for structure!

OPTIMAL LENGTH & COMPOSITION: → 350 words (±20 words for safety) → 10-12 sentences (avg 30-32 words each) → 60+ bigrams, 20+ trigrams, 8-10 sequences of 8+ words → 8-12 proper names preserved exactly

———————————————————— MASTER EXAMPLES (Study these 500+ point summaries): ————————————————————

<example id="1"> <judgment> If a woman agrees to waive her right to claim maintenance from her husband, and opts for a divorce by mutual consent,

she cannot later demand maintenance under the Code of Criminal Procedure (CrPC), the Madras High Court has held. In a judgment passed on September 16, Justice Bharatha Chakravarthy rejected a revision petition filed by a woman challenging the decision of a family court that had refused to direct her ex-husband to pay her a monthly maintenance of 1 lakh, and to pay a lump sum amount of 5.80 crore for the medical treatment of their 35-year-old son. </judgment>

<elite_summary> If a woman agrees to waive her right to claim maintenance from her husband, and opts for a divorce by mutual consent, she cannot later demand maintenance under the Code of Criminal Procedure (CrPC), the Madras High Court has held. In a judgment passed on September 16, Justice Bharatha Chakravarthy rejected a revision petition filed by a woman challenging the decision of a family court that had refused to direct her ex-husband to pay her a monthly maintenance of 1 lakh, and to pay a lump sum amount of 5.80 crore for the medical treatment of their 35-year-old son. </elite_summary>

<v2_scoring> PROGRESSIVE REWARDS: • +25 pts × 2 (16+ word sequences): "If a woman agrees to waive her right to claim maintenance from her husband and opts for a divorce by mutual consent" (21 words), "In a judgment passed on September 16 Justice Bharatha Chakravarthy rejected a revision petition filed by a woman challenging the decision of a family court" (25 words) = +50 pts • +20 pts × 3 (11-15 word sequences) = +60 pts • +15 pts × 2 (8-10 word sequences) = +30 pts

CONTEXTUAL BONUSES: • ×2.0 (first sentence, 21 words): 25 × 2 = +50 pts • ×1.3 (legal transitions "has held"): +13 pts • +5 pts × 15 (exact terms): "Madras High Court", "Justice Bharatha Chakravarthy", "CrPC", etc. = +75 pts • +2 pts × 75 (bigrams) = +150 pts

DENSITY BONUSES: • +20 pts: 18 unique trigrams per 100 words • +15 pts: 2 sequences 8+ words per 50 words • +10 pts: 3.1% legal term density

STRUCTURAL BONUSES: • +15 pts: Perfect opening (Court + action + case) • +10 pts: Chronological flow

NO PENALTIES: Zero hallucinations, zero paraphrasing

TOTAL SCORE: 470 points

WHY ELITE: • Two 16+ word sequences in opening = Massive ROUGE-L boost • 75+ bigrams = Maximum BLEU score • Zero paraphrasing = Perfect precision • All density targets hit = Optimal n-gram distribution </v2_scoring> </example>

<example id="2"> <judgment> The Allahabad High Court held a special hearing on Sunday evening to initiate a suo motu case on the recent attack on a Uttar Pradesh Police woman officer, who was found injured on a train. A Bench of Chief Justice Pritinker Diwaker and Justice Ashutosh Srivastava took suo motu note of the incident on the basis of a WhatsApp message received by the Chief Justice. </judgment>

<elite_summary> The Allahabad High Court held a special hearing on Sunday evening to initiate a suo motu case on the recent attack on a Uttar Pradesh Police woman officer, who was found injured on a train. A Bench of Chief Justice Pritinker Diwaker and Justice Ashutosh Srivastava took suo motu note of the incident on the basis of a WhatsApp message received by the Chief Justice. </elite_summary>

<v2_scoring> PROGRESSIVE REWARDS: • +25 pts × 1 (16+ words): "The Allahabad High Court held a special hearing on Sunday evening to initiate a suo motu case on the recent attack on" (22 words) = +25 pts • +20 pts × 2 (11-15 words): "A Bench of Chief Justice Pritinker Diwaker and Justice Ashutosh Srivastava took suo motu note" (14 words) = +40 pts • +15 pts × 3 (8-10 words) = +45 pts

CONTEXTUAL BONUSES: • ×2.0 (first sentence, 22 words): 25 × 2 = +50 pts • ×1.3 (legal transition "held"): +13 pts • +5 pts × 12 terms: "Allahabad High Court", both judge names, etc. = +60 pts • +2 pts × 68 bigrams = +136 pts

DENSITY BONUSES: • +20 pts: 16 trigrams per 100 words • +15 pts: Strong 8+ word density

STRUCTURAL BONUSES: • +15 pts: Perfect opening

TOTAL: 440 points

WHY ELITE: • 22-word sequence in opening (×2.0) = Huge position bonus • All judge names preserved exactly = Zero penalties • 68 bigrams = Excellent BLEU </v2_scoring> </example>

JUDGMENT TO SUMMARIZE:

A.2 Hybrid Pipeline – Extraction Prompt

Goal: Copy literal text segments from judgments into a structured JSON schema. **Guidelines:**

- Extract full sentences for Facts, Arguments, and Reasoning.
- Preserve exact legal terminology (e.g., "appellant", "respondent").
- Keep procedural phrases ("filed a petition", "argued that").

Output Schema: `{Court, Date, Parties, Counsel, Facts, Issues, Arguments, Decision, Reasoning, Orders, Citations}`

A.3 Hybrid Pipeline – Abstraction Prompt

Objective: Produce a 450–550 word summary using only extracted text.

Constraints:

- Every word must come from extracted fields.
- No elaboration, synonyms, or paraphrasing.
- Maintain original sentence structures.

Assembly Order: Court/Date/Parties → Facts → Issues → Arguments → Decision → Reasoning → Orders.

Optimization: Maximize n-gram overlap for ROUGE-L/BLEU; minimize lexical diversity.

A.4 Simple Baseline Prompt

`Tl;Dr {text}`

A minimal baseline prompt inspired by Deroy et al. (2024), with no constraints or structure.

A.5 Domain Classification Prompt

This system prompt is used in the initial stage to classify the legal judgment into one of the predefined areas of law.

> You are an expert legal domain classifier for Indian judgments.
>
> Your task: Classify the following legal judgment into ONE of these specific areas of law:
>
> legal_domains
>
> CRITICAL INSTRUCTIONS: - Read the judgment carefully and identify the primary legal domain - Consider the subject matter, legal issues, and procedural context - Return ONLY the exact area of law from the list above - If uncertain, choose the most appropriate domain based on the main legal issues
>
> LEGAL JUDGMENT TEXT: text
>
> CLASSIFICATION:

A.6 Domain-Specific Structured Extraction Prompt

This prompt guides the model to perform the extractive stage, copying literal text segments into a structured JSON format. It is dynamically populated with domain-specific information (e.g., expected section order and target lengths) derived from the analysis of the legal domain.

> You are an expert legal information extractor for Indian judgments in the domain_characteristics.get('domain', 'legal') domain.
>
> Your task: Extract structured information following the typical pattern for this legal domain.
>
> EXPECTED SECTION ORDER FOR THIS DOMAIN: most_common_order
>
> TARGET LENGTHS FOR EACH SECTION: sections_info
>
> CRITICAL INSTRUCTIONS FOR MAXIMUM ROUGE-L AND BLEU: - COPY verbatim phrases and sentences from the original text - DO NOT paraphrase or rephrase — extract exact textual segments - Preserve original wording, terminology, and sentence structure - Use literal quotes from the judgment for all fields - Maintain exact punctuation, capitalization, and legal terminology - Follow the expected section order for this domain
>
> Required JSON schema (exact keys): "ID": "<use provided id when available or empty>", "Court": "", "Date": "", "Parties": "Petitioner": [], "Respondent": [] , "Counsel": [], "ProceduralHistory": "", "Facts": [], "Issues": [], "Arguments": [], "Decision": "", "Reasoning": [], "Orders": "", "Citations": []
>
> IMPORTANT: - Output MUST be valid JSON only - Extract by COPYING literal text segments — do not summarize or paraphrase - If a field is not present, use empty string or [] - Preserve exact names, dates, legal terms from the original judgment - Follow the domain-specific section order and length guidelines
>
> LEGAL JUDGMENT TEXT: text

A.7 Domain-Specific Abstractive Summarization Prompt

This prompt guides the final abstractive stage, forcing the model to reuse the literal text extracted in the previous step and adhere to domain-specific structural and length guidelines. The goal is to maximize lexical overlap (ROUGE/BLEU) while maintaining narrative flow.

> You are an expert legal summarizer optimized for MAXIMUM ROUGE-L and BLEU scores for domain_characteristics.get('domain', 'legal') domain cases.
>
> CRITICAL OPTIMIZATION STRATEGY: - REUSE verbatim phrases and sentences from the extracted fields - COPY literal n-grams (3-5+ word sequences) from the source text - MINIMIZE paraphrasing — preserve original wording wherever possible - MAINTAIN exact legal terminology, names, dates, and citations - Build summary

by CONCATENATING and ARRANGING literal text segments - FOLLOW the domain-specific section order and length guidelines - DO NOT use section headers or titles - write as a flowing narrative

DOMAIN-SPECIFIC GUIDELINES: Expected section order: most_common_order Target lengths per section: length_guidelines

Target specifications: - Length: 500 words (±25- Sentence length: 27-32 words average - Maximize lexical overlap with reference summaries - Preserve chronological flow and judicial objectivity - Follow the typical structure for this legal domain - NO section headers, titles, or bold formatting - write as continuous text

ASSEMBLY INSTRUCTIONS: 1. Start with verbatim party names and court/date information 2. Follow the domain-specific section order: most_common_order 3. Incorporate literal sentences from Facts, Issues, Arguments 4. Copy exact Decision and Reasoning statements 5. Include verbatim Orders and Citations 6. Link segments with minimal connecting phrases (use "and", "while", "following", etc.) 7. DO NOT create new phrasings — recombine existing text 8. Respect the target lengths for each section type 9. DO NOT use any section headers, titles, or formatting - write as a natural flowing summary

Input: Use the extracted fields below and REUSE their literal text.

extracted_fields

Generate a comprehensive legal summary by REUSING and ARRANGING the literal text segments above. Write as a natural flowing narrative without any section headers or titles. Maximize word-for-word overlap while maintaining natural flow and following the domain-specific structure.