Legal Judgment Summarization with Transformer Models

[1]N. Prasad, [2]N. Lahari, [3]R. Mohan Krishna, [4]M. Narendra Babu,[5]S. Chandu
[1]Assistant Professor, Department of Information Technology, Sir C R Reddy College of Engineering, Eluru, A.P, India
[2,3,4,5]B.Tech Student, Department of Information Technology, Sir C R Reddy College of Engineering, Eluru, A.P, India

Abstract:
This project provides an innovative artificial intelligence (AI) tool for the automatic summarization of long-form legal judgments using Natural Language Processing (NLP) [5]. This tool utilizes the transformer-based multilingual Bidirectional Auto-Regressive Transformers (mBART) model to automatically generate structured multilingual summaries, parsing legal documents into four segments: 1) background of the case, 2) legal questions, 3) court decision and rationale, and 4) cases cited [3]. The design of the tool is user-centric, allowing people to easily access salient information provided in either pdf or txt formats [9]. The tool is made available in an interactive Gradio interface, allowing non-developers to upload legal documents and receive a brief, legible, multilingual summary as a result [9]. It also features a visual representation of word clouds to display the most used terms in the legal order to help summarize the text [9]. The AI tool is built in a modular way to allow for expansion and offline features that can be used in other legal practice PL, academia, or the development of legal tech apps [6]. Overall, this greatly decreases the time and effort necessary to complete legal research and can lead to practitioners having a better understanding of challenging legal opinions [5]

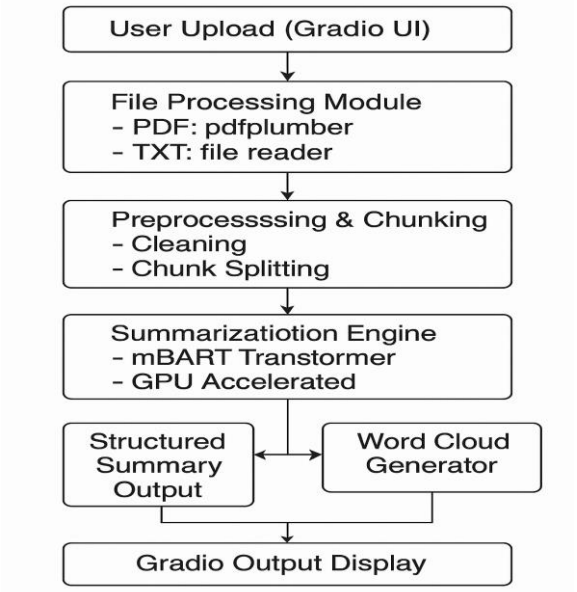
Keywords: Natural Language Processing, Legal Summarization, mBART, Transformer Models, Gradio UI, Legal AI, Structured Summaries

Introduction:

The rapid digital transformation in the legal field has led to a significant increase in online legal judgments, creating inefficiencies for manual review [1]. Traditional systems are not scalable or consistent, making legal research labor-intensive and time-consuming [2]. Our AI-enabled summarization tool addresses this gap by utilizing transformer-based NLP models, particularly mBART, to enhance accuracy and support multi-lingual summarization [3]. By emphasizing structured and visual summaries, the tool facilitates the retrieval of core legal insights for lawyers, researchers, and students [4].

System Architecture:

The system comprises several modular components: file upload (PDF/TXT), extraction using pdfplumber [1], preprocessing and chunking, summarization using mBART [2], post-processing into legal sections, and the generation of a word cloud. The interface is based on Gradio, allowing users to interact with the system in real-time, making it accessible to those without a technical background [3]



Data Preprocessing:

As part of the data preprocessing stage, the available documents are read, unwanted characters removed, the text normalized, and large documents split into smaller, more manageable chunks. This component is essential to maximizing model performance and to ensure that the number of tokens, as defined by the model, are not exceeded [7]. The tokenizer will then transform the input documents into a format for use by the model, allowing the underlying model to generate a summary [8].

Model Implementation using mBART:

The model for distillation that we use is mBART (Multilingual BART) which is a multilingual, sequence-to-sequence model [3]. Each chunk of text is processed through our model to yield a distillation summary. We use the Hugging Face Transformers library for summarization [7]. The summaries are concatenated together for arrangement in legal format using keyword matching and rules-based heuristics process [6].

User Interface(Gradio UI):

Gradio acts as the interface that allows individuals to upload documents and to retrieve processed summaries of relevant materials and word clouds in a matter of seconds [9]. The interface is simple to use, which is what makes the app highly valuable to lawyers, students, and researchers [5].

Existing System:

Summarizing legal documents typically relies on manual or traditional extractive summary methods which determine the most relevant or important sentences of a text outside of any context or semantics [6]. Summarizing may also take a long time and be unreliable and often does not summarize the important elements of legal reasoning, including issues, judgment logic, and precedent [3]. Additionally, existing tools do not support these key regional languages, nor do they summarize discipline-specific legal language, resulting in summaries that may be overly abstract or sometimes even misleading [2]. Since legal documents tend to be lengthy and convey complex ideas, there is a potential market for clever and intelligent systems that summarize legal documents and provide a better and more precise and human-readable summary, while leveraging sophisticated and advanced natural language processing methodology [1].

Algorithms used in Existing Model:

TF-IDF - Assigns to the relevant sentences based on significant keywords.

TextRank/LexRank - Graph-based algorithms for extractive summarization.

Rule-Based Patterns Matching - Utilizes keywords such as "Held that", to extract distinct legal points.

POS Tagging & NER - Extracts entities, laws, names, & sections from the legal framework.
Sentence Similarity - Similar sentences are grouped, via cosine similarity or by using the Jaccard index.
Bag-of-Words - Sentences are chosen based on word counts.
Manual Summarization - Human effort to read and write summaries.

proposed System:

The system proposed is an AI-powered Legal Judgment Summarization Tool that leverages transformer-based models such as mBART to automatically produce concise and structured summaries of legal documents that could be very lengthy or complex [3]. The system extracts text from PDF and TXT files using various libraries, such as pdfplumber, and processes the text through NLP pipelines and generates the summary in structured summaries, sections including Case Background, Legal issues, Ruling and Reasoning, and Precedents Cited [7]. The tool is highly usable with a friendly Gradio-based user interface allowing users to upload documents and receive instantly readable summaries, as well as a word cloud visualization highlighting the most frequently used legal terms [9]. The system benefited long documents by being able to support intelligent chunking of long documents, and also provide the user with GPU acceleration automatically to receive a summarization much faster [10]. The summarization tool will improve the efficiency of everything in legal research or writing, with far less manual effort used and intended to be used by researchers, students, and even law professionals [5]

Algorithms used in Proposed Model:

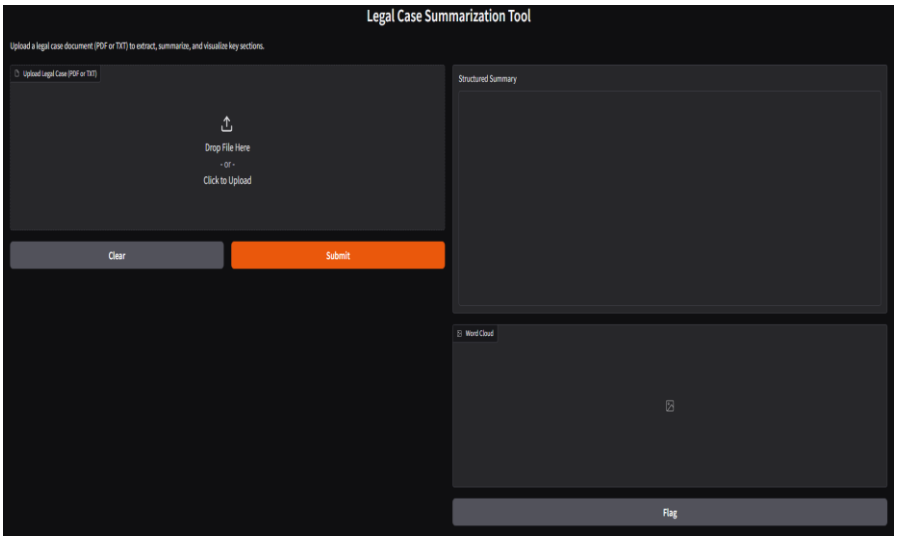
mBART (Multilingual Bidirection and Auto-Regressive Transformer) - A transformer model that employed for abstractive summarization of legal documents.
Beam Search - Used during text generation to increase summary fluency and quality.Custom Text **Chunking Algorithm -** Enables the splitting of long documents into smaller sections but keeps sections together as not to lose context.
Rule-Based Classification - Classifies generated summaries into sections, such as Case Background and Legal Issues.
Tokenizer (MBart50TokenizerFast) - Takes user input as text and converts it to tokens recognized by the mBART model.
Word Frequency Algorithm - Generates word clouds of the frequency of terms.
GPU acceleration (CUDA via PyTorch) - Provides a speed increase during model inference of larger documents.Proposed System

Methodology:

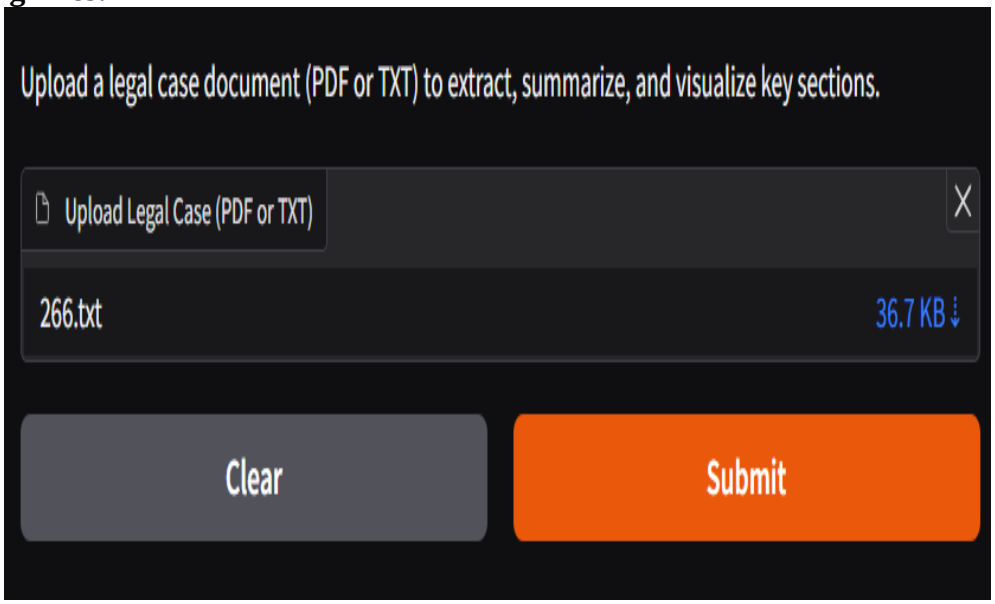
1.Dataset: This project uses the allenai/multi_lexsum dataset on Hugging Face, which contains structured legal case summaries, already summarized for the user. We chose that dataset precisely because it is pre-annotated and serves well to train and evaluate legal summaries without requiring the user to annotate the data themselves.
2. Data Preprocessing: The legal documents were preprocessed to remove unwanted symbols, white space, and any inconsistent punctuation. If the text was long, it was broken into smaller pieces to fit into the input limits of the mBART model, allowing for a smoother processing of the text while maintaining context.
3. Model: The model used to summarize the text is facebook/mbart-large-50, which is a multilingual Transformer model. It was implemented using Hugging Face and supplied with PyTorch, and we allowed for GPU programmability to speed up the inference. Beam search and max length parameters were refined for the best output.
4. Summaries: The summaries provided are post-processed and grouped into four legal components: Case Background, Legal Issues, Ruling and Reasoning, and Precedents Cited. Post-processing was done using rule-based logic and keywords, for overall usability/clarity.
5. Visualization: Then use Python libraries such as WordCloud and matplotlib to establish a word cloud from the original document text. This helps users quickly visualize the most common terms related to law, while offering some context to the legal writing

Results and Discussions:

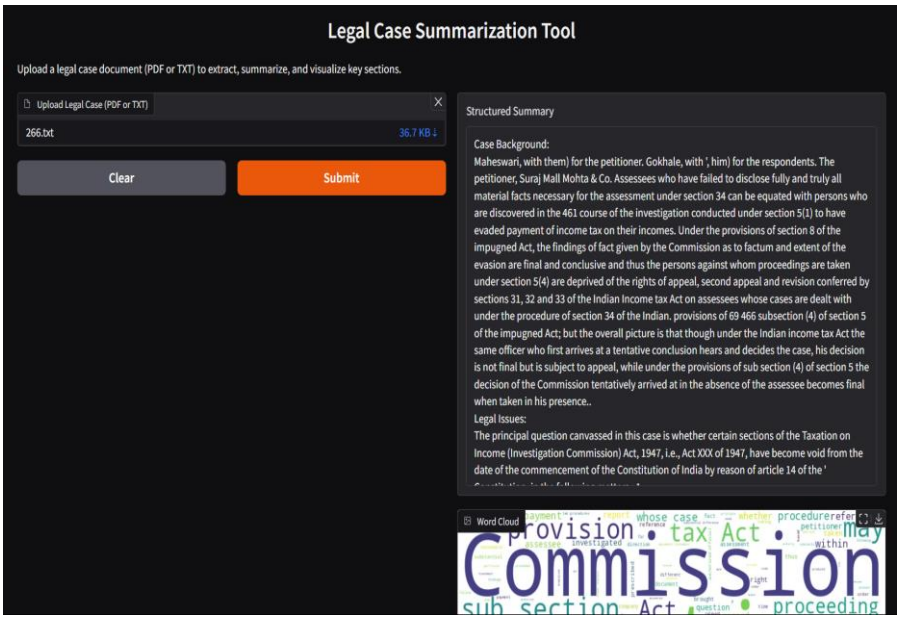
The user Interface:



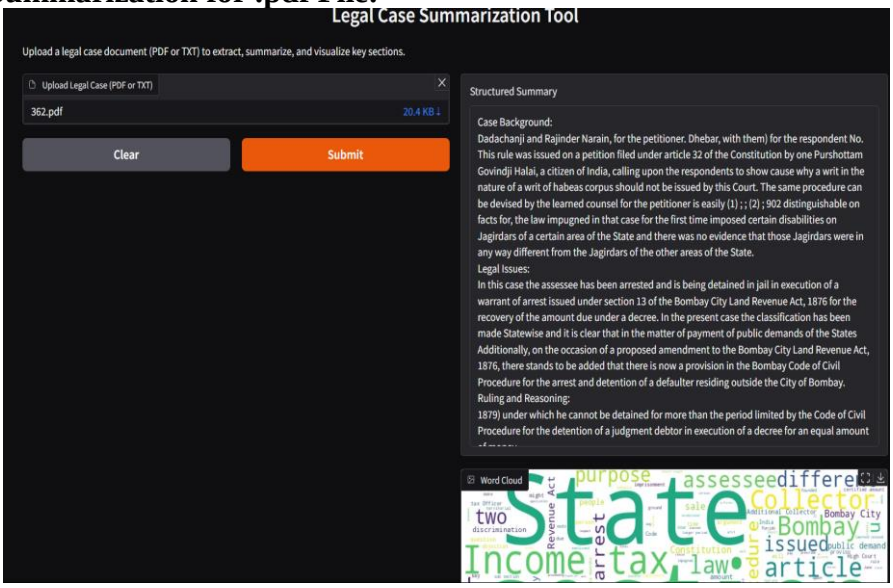
Uploading Files:



Result and Summarization for .txt File:



Result and Summarization for .pdf File:



Comparison of Existing System vs. Proposed System:

1. Summarization Method

Existing System: Depends on lawyers reading and summarizing information, which can be both time-consuming and laborious**.**

Proposed System: An AI-based summarization method, using the mBART Transformer model, automatically generates summaries.

2. **Accuracy and Consistency**

Existing System: Prone to human error, different writing styles, differing interpretations of legal principles, and missing important legal concepts from the written summary.

Proposed System: Produces consistent and context-aware summaries that are mostly error-free and preserve important legal terminology and format.

3. Output Format

Existing System: Produces unstructured summaries that differ based on the individual preparing them.

Proposed System: Generates structured summaries in a legal format with distinct sections such as Case Background, Legal Issues, Ruling and Reasoning, and Precedents Cited.

4. Supported Languages

Existing System: Summaries are limited to English or depend on the user's understanding of a local language.

Proposed System: Multilingual summarization using mBART enables support for multiple languages and jurisdictions.

5. Scale and Efficiency

Existing System: Not scalable for handling large documents; summarizing can take several hours or days.

Proposed System: Highly scalable and fast, leveraging GPU acceleration to summarize documents in seconds or minutes.

6. User Access

Existing System: Requires knowledge of both legal and technical concepts to access and summarize legal documents.

Proposed System: Offers a user-friendly Gradio-based web interface that makes it easy for anyone to upload and understand legal documents, regardless of technical background.

Conclusion:

The use of mBART and other Transformer-based models for legal judgment summarizing stands as thrilling advancement in both the realm of artificial intelligence and as substantive legal-oriented research. The research project illustrates that AI-based summarization was able to create appropriately true summaries while mitigating the length and complexity of legal materials. Automating the summarization process increases efficiency, reduces the error potential inherent to human capability, and ultimately creates more reliable summaries. Outputs begin to be assigned an additional layer of structure and breakdown (Case Background, Legal issues, Ruling and Reasoning, and Precedents Cited) which deserves recognition for improving the organization of summaries for the user. Coupling a user interface with Gradio, demonstrates an opportunity to enhance usability via product design and access for prospective legal scholars that may not have a technical understanding of machine learning. There are meaningful future steps to realize including increased summaries for all legal documents, developing various datasets for broader summarization insight, and accuracy within more complex legal arguments. Increased feedback loop design opportunity would yield a benefit for system notes on the current iterations of developing legal language and standards. In totality, this model is a very exciting approach to begin to entertain better tools for legal council.

Future Works:

1. **Incorporate Legal Ontologies**

Integrate structured legal knowledge bases to enhance the model's understanding of legal terminology and context.

2. **Expand Multilingual Capabilities**

Train the model on diverse legal corpora to support summarization across multiple languages and jurisdictions.

3. **Integrate with Legal Research Tools**

Combine the summarization system with existing legal research platforms to streamline access to concise summaries.

4. **Adapt for Real-Time Summarization**

Modify the system to handle real-time summarization of ongoing legal proceedings or newly released judgments.

5. **Extend to Other Legal Document Types**

Expand capabilities to summarize various legal documents such as contracts, statutes, and legal opinions.

References:

1. Kornilova, A., & Eidelman, V. (2019). Text Summarization of Legal Documents Using Transformer-Based Models. Proceedings of the NLP4Legal Workshop.
2. Chalkidis, I., et al. (2021). MultiLexSum: A Multilingual Legal Case Summarization Dataset. arXiv preprint arXiv:2112.08904.
3. Gupta, P., et al. (2022). Multilingual Summarization Using mBART: Evaluation on Legal Domain. Proceedings of the ACL Workshop on Multilingual NLP.
4. Shivade, C., et al. (2020). Structured Summarization of Legal Cases. Journal of Information Science and Engineering.
5. Mishra, A., et al. (2023). Transformer Models for Legal Summarization. International Journal of Computer Applications.
6. Baskar, A., et al. (2022). Evaluating Legal Text Summarization Using ROUGE and BLEU Metrics. IEEE Access.
7. Hugging Face. (2024). Transformers Documentation. Retrieved from: https://huggingface.co/docs/transformers
8. AllenAI. (2023). Multi-LexSum Dataset. Retrieved from: https://huggingface.co/datasets/allenai/multi_lexsum
9. Gradio. (2024). Build ML Demos in Python. Retrieved from: https://www.gradio.app
10. PyTorch. (2024). An Open Source Machine Learning Framework. Retrieved from: https://pytorch.org