

# Legal Summarizer using Deep Learning

**Anuj Chandak**

*Department of Computer Engineering  
NMIMS MPSTME  
Mumbai, India  
anuj.chandak73@nmims.in*

**Karan Dave**

*Department of Computer Engineering  
NMIMS MPSTME  
Mumbai, India  
karan.dave31@nmims.in*

**Kaustub Mule**

*Department of Computer Engineering  
NMIMS MPSTME  
Mumbai, India  
kaustub.mule07@nmims.in*

**Soham Agarwal**

*Department of Computer Engineering  
NMIMS MPSTME  
Mumbai, India  
soham.agarwal159@nmims.in*

**Abhay Kolhe**

*Department of Computer Engineering  
NMIMS MPSTME  
Mumbai, India  
abhay.kolhe@nmims.edu*

**Manisha Tiwari**

*Department of Computer Engineering  
NMIMS MPSTME  
Mumbai, India  
manisha.tiwari@nmims.edu*

**Abstract**—The increasing complexity and volume of legal documents make it difficult and time-consuming to analyze the texts manually. In this paper, we propose an AI-driven Legal Text Summarization system that uses Deep Learning and Natural Language Processing (NLP) technologies to produce concise coherent summaries that are automatically developed from lengthy legal texts. Our proposed system implements and compares summarization approaches of two different categories: extractive approaches (TextRank, LexRank, and BERT) that extract the most informative sentences from source documents, and abstractive approaches (fine-tuned T5, Pegasus, BART, and LSTM) that produce human-like rephrased summaries. Our experimental results also show that transformer-based abstractive models significantly outperform traditional extractive and recurrent approaches with respect to semantic retention, fluency, and readability. In conclusion, the results of our experiments show the effectiveness and applicability of domain-adapted summarization systems to enhance the accessibility and efficiency of legal document analyzes for practitioners and researchers.

**Index Terms**—Legal Text Summarization, Natural language processing, Deep learning, Long Short-Term Memory, Bidirectional Encoder Representations from Transformers

## I. INTRODUCTION

The legal discipline alone generates volumes of text on a daily basis such as legislative bills, court opinions, contracts, and regulation documents, like never before. Academics and legal workers are commonly tasked with perusing and interpreting texts of substantial size to find pertinent answers, results, and precedents, which is a tedious exercise that is more likely to suffer adverse effects of cognitive bias. This issue throws light on the importance of having computerized summarization systems which will be effective in synthesizing briefs of the most significant information without compromising on the legal context and accuracy.

The problem of legal text summarization differs to the problem of general-domain summarization. To start with, legal texts are not similar to other text types because of their use of specialized vocabulary, use of complex sentences, presentation in a hierarchical way, cross-referencing, and accurate semantics as slight modifications to legal texts can alter its meaning.

Recent developments in the field of summarization in the past decade have seen the shift of extractive, graph-based algorithms (TextRank, LexRank) to more advanced models, such as RNNs and LSTMs. Nevertheless, all these approaches are majorly limiting. Graphs cannot be used to provide the semantic and contextual richness of legal text. In the same manner, RNNs and LSTMs also performed better in sequences, but they cannot handle long dependencies and vanishing gradients. The emergence of transformer-based architectures, such as BERT, alleviated most of these problems through self-attention to learn more about contextual relationships.

Nevertheless, they usually learn to general-domain corpora (such as news articles) and are thus inaccurate and slow on domain-specific legal text, where accuracy and context are essential. This paper fills this gap by proposing a Legal Text Summarization framework that contrasts between extractive and abstractive models on the basis of U.S. congressional bills of the BillSum dataset. We compare extractive models (TextRank, LexRank, BERT) to abstractive models (fine-tuned T5, Pegasus, BART and LSTM).

The rest of this paper will be organized in the following way: Section II will review the related literature, Section III will describe the methodology used, Section IV will discuss the results, and Section V will be the conclusion of the paper.

## II. LITERATURE REVIEW

The practice of summarizing legal documents has been of increasing interest in the last few years as the legal industry is beginning to see the demand for quick, effective retrieval of information from legal documents. Most of the studies we've seen thus far can be categorized broadly into three connected categories: [1] domain adaptation of NLP models to the unique structure of legal documents and semantics of legal concepts, [2] retrieval-augmented generation to encourage factual grounding and alleviate hallucinations, [3] methods of addressing low-resource and long-document issues and [4] comparative approaches contrasting extractive, abstractive, and hybrid models.. This research all contributes to our comparative

exploration of deep learning models on the BillSum dataset, in particular Legal-BERT against traditional RNN/LSTM and transformer baselines.

#### A. Domain Adaptation with Legal-Aware Encoders

Legal documents are syntactically complicated and include lengthy sentences, instances of statutory cross-reference, and relatively uncommon legal phrases, all of which can impact the performance of general-purpose NLP models. To address these issues, previous research has developed domain-adapted encoders that are pretrained on legal data. Legal-BERT was pretrained using legal publications such as statutes, contracts, and case law, capturing the nuances of 'meaning that are common, but perhaps not evident, in the legal domain compared to vanilla BERT'. Furniturewala et al. [2] showed that finetuning Legal-BERT for relevance classification outperformed generic transformers and more traditional classifiers based on machine learning. They also presented evidence that appended statistical features, like TF-IDF, to Legal-BERT embeddings were a way to find improved performance and robustness in low-data scenarios. Overall, their studies affirmed absence of domain adaptation from legal to legal provided improvements of both accuracy and faithfulness in generated summaries, and established a benchmark for downstream studies focused in the legal space with their use of Legal-BERT.

#### B. Retrieval-Augmented Generation for Factual Grounding

A well-known issue affecting abstractive summarization is hallucination; that is, the model creates a statement that the source text does not support. Fortunately, retrieval-augmented generation (RAG) has been explored as a way to ground abstractive outputs in reliable source text. Mukund and Easwarakumar [1] introduced Dynamic Legal RAG, which uses Legal Named Entity Recognition (NER) to modify an entity-aware BM25 retriever to find and extract text snippets containing the most relevant provisions, clauses or case cites. These text snippets are then integrated into the generative model to support factual consistency and coherence. Their testing showed that decoder-only LLMs conditioned on the retrieved chunks showed strong BERTScore performance and less factual drift. A major advantage is that this approach is particularly useful for legal legislative text such as U.S. congressional bills or similar types of documentation, which often cite previous statutes and amendments. When a generation is conditioned on a retrieved legal passage, RAG approaches enhance transformers like Legal-BERT by strengthening reliability and legal validity, or correctness to the text.

#### C. Low-Resource, Long Documents and Evaluation Strategies

Taking into account the limited availability of annotated training data and the large length of legal documents, legal summarization presents its own unique challenges. The CLSum dataset [3] overcame these challenges by creating a large-scale multi-jurisdiction corpus and analyzing data augmentation based on LLMs. Furthermore, they proposed LTScore- a domain-specific scoring metric that favors the preservation

of legal terms in the original text to enhance the assessment of quality in the legal domain. In relation to modeling, chunking with overlapping windows, sparse attention (e.g., Longformer, BigBird), and two-stage pipelines (i.e. content selection then abstraction) have been efficient in processing long inputs, such as congressional bills. Parameter-efficient finetuning approaches such as LoRA or adapter layers have also enabled the training of these models under resource constraints while not losing performance. Finally, despite ROUGE often being talked about as a metric for evaluation, researchers are quick to advocate to add semantic metrics (e.g., BERTScore) and domain-aware metrics (e.g., LTScore) along with human evaluation to ensure fluency and statutory correctness.

#### D. Extractive vs. Abstractive Approaches and Hybrid Models

Legal summarization methods can broadly be categorized into extractive and abstractive approaches, both of which exhibit strengths and weaknesses. Extractive approaches, involving ranking and selecting only those sentences most relevant to the summary, are prized for their factual accuracy and trustworthiness. Furniturewala et al. [2] illustrated that classification of sentences combined with role labeling resulted in summaries that had remarkable factual coverage; however, the sentences themselves were at times associated repetitively or haphazardly. Abstractive models such as BART, T5, and Legal-PEGASUS produce fluent and concise summaries that arguably capture a more comprehensive meaning from legal texts, although they are still susceptible to hallucination. Hybrid approaches to legal summarization have been proposed as a way to combine advantages of both: first extracting candidate content, and afterwards generating or aggregating this content in an abstraction. Mukund and Easwarakumar [1] showed how retrieval-augmented abstractive models further advance the conceptual advantages of hybrid approaches, allowing generated content to be grounded in retrieved provisions while delivering readability and legal fidelity. Hybrid approaches are gaining popularity in legal summarization to support the broader purpose of legal summarization by both precision in factual content and interpretability.

### III. METHODOLOGY

In designing our methodology, we aim to develop and assess deep learning models to facilitate automated summarization of legal documents in a systematic way. Implementation of this research occurred in two primary phases: (1) NLP-based preprocessing and exploratory analysis, and (2) model-based summarization using extractive and abstractive techniques. The experiments were conducted using the BillSum dataset from HuggingFace, which consists of U.S. Congressional bills and their associated summaries written by humans. Throughout our research, data was cleaned, features extracted and applied, and the models evaluated in efforts to assess the approaches for legal summarization. In general, we found that transformer-based approaches were advantageous to traditional sequence models.

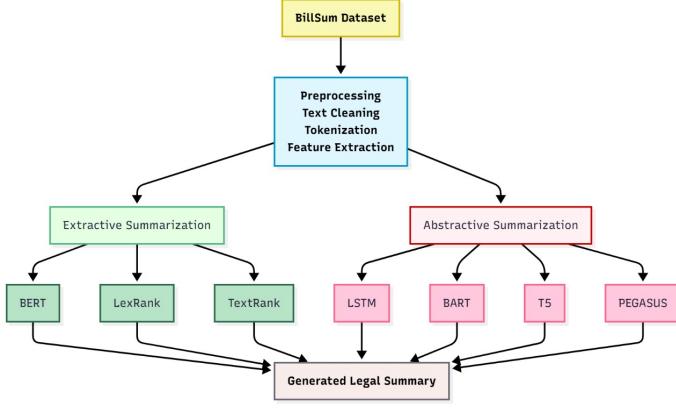


Fig. 1. Data Flow Diagram of the Legal Text Summarizer

#### A. NLP Pre-Processing and Analysis

The first phase of implementation involved preparing the BillSum dataset for deep learning based summarization. Due to the nature of legal text being formal and complex, it was important to conduct pre-processing to improve the quality of the data as well as the performance of the models. The text underwent pre-processing steps that included cleaning the text by removing punctuation, stop words, URLs, and HTML tags and converting the text to lower case for uniformity. Lemmatization was included to reduce all of the words to their standardized forms, and TF-IDF vectorization was performed to convert the text into numeric representations capturing the importance of the terms within the context of the documents. Part of speech tagging and named entity recognition (NER) were also applied to identify grammatical structures as well as extract legal entities (e.g., acts, sections, organizations). Pre-processing outlined above was important because it ensured that the post-processed text preserved linguistic integrity and contextual integrity, which would in turn assist the summarization models in producing more accurate and legally coherent summaries.

#### B. Extractive Summarization

Extractive summarization seeks to point out and draw out the most significant sentences in the source text without distorting the words. The method is specifically effective in situations when legal integrity and factual accuracy have to be ensured strictly. This category implemented three models namely BERT, TextRank and LexRank.

1) *BERT (Bidirectional Encoder Representations from Transformers)*: Extractive summarization BERT-based summarization leverages semantic summary based on deep bidirectional contextual embeddings to find best sentences that reflect the semantic content of a document. In contrast to such classic frequency-based or similarity-based lists as TextRank or LexRank, BERT understands the meaning of the words in the complete left and right context, providing a deeper insight into the significance of the sentences. The model categorizes every sentence by the contextual appropriateness of the sentence to

the whole document and picks out the most representative ones to build the final summary. This semantic-based solution is able to fill the gap between statistical ranking and real contextual understanding and BERT is especially helpful in legal documents where accuracy, terms, and logical connections hold the key.

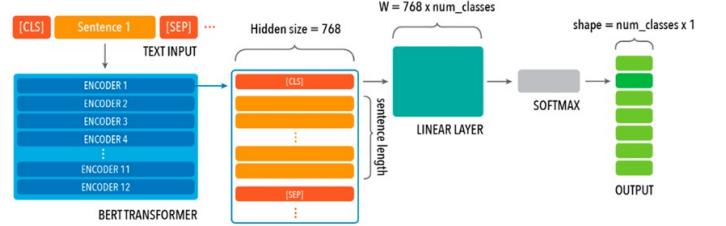


Fig. 2. BERT Architecture

2) *TextRank*: TextRank is a ranking algorithm, a graph algorithm based on the PageRank algorithm of Google, which is used to rank the most suitable sentences in a text based on similarity relationships. Sentences are the nodes and the lexical overlap or the cosine similarity is the weight of the edge. The algorithm splits the sentences into coherent summaries by calculating the importance of the sentences based on their relations and the final results are the sentences that are ranked the highest. TextRank is straightforward and does well at general-purpose summarization, but because it depends on surface-similarity, it does not do so well at identifying the more advanced semantic or contextual relationships present in complex legal text.

3) *LexRank*: LexRank is an extension of TextRank present with the use of cosine similarity alongside eigenvector centrality to rank sentences in a document graph. It assesses the importance of sentences globally by examining the connectivity patterns in the sentence so that the model can be able to determine the importance of a sentence that is more locally significant and yet central to the entire text. It makes LexRank more appropriate to longer and more complex documents in law because it guarantees coverage of the topic, factual consistency, and logicality. Its graphical form ensures that it is interpretable and it also has better performance compared to purely statistical methods.

#### C. Abstractive Summarization

Abstractive summarization is concerned with the creation of new sentences which set part of the original document in a parsimonious and lucid way. It is not like extractive methods and therefore the model must be competent in context and ability to paraphrase and flow of the text. Four architectures LSTM, fine-tuned T5, Pegasus, and BART were provided and compared.

1) *LSTM (Long Short-Term Memory)*: LSTM-based encoder-decoder model is our baseline architecture of abstractive summarization, which is the conventional recurrent neural network model of sequence-to-sequence learning. The architecture is made up of two major parts, an encoder which works with the

input bill text in sequence to construct a compact representation of the whole document, and a decoder which produces a summary token by token according to this encoded form of the document. Even the LSTM cells are specially made to handle the vanishing gradient issue of the standard recurrent neural networks by using a complex gating mechanism that includes input gates, forget gates, and output gates that control the flow of information. The gates allow the model to selectively store significant data and forget superfluous information over long sequences and thus LSTMs are theoretically viable to learn long-term dependencies in long texts.

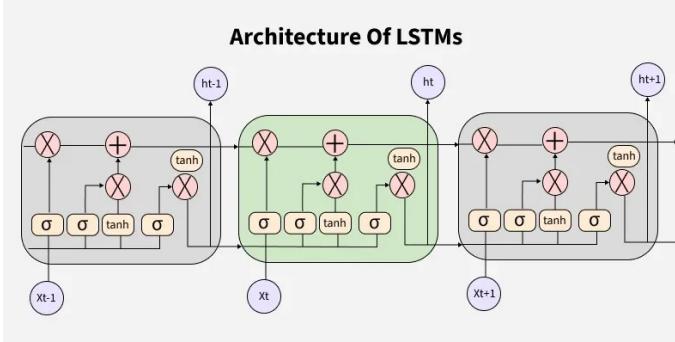


Fig. 3. LSTM Architecture

Nevertheless, in spite of such architectural improvements, LSTM models have serious shortcomings when used with very long legal documents like congressional bills. The sequentiality of the recurrent process implies that information needs to flow through thousands of time steps, and despite the gating mechanism, the model has difficulties in preserving detailed representations of content at the start of thousand-token documents to the end.

Also, the attention mechanism, though useful, is capable of countering the underlying limitation that all input information has to flow through the recurrent hidden states, only partially. Practically, we found that LSTM-based summaries tend to overlook nuanced contextual associations, are limited in their ability to relate widely-distinct passages of bills, and in other cases are capable of producing summaries in which the second half of documents receive disproportionately high attention. More so, sequential processing requirement also renders training and inference computationally costly with limited possibilities of parallelization relative to transformer-based models.

2) *Fine-tuned PEGASUS (Pre-training with Extracted Gap-sentences for Abstractive Summarization)*: Pegasus is trained on gap-sentence generation and abstractive summarization tasks on large-scale summarization data. The model is trained to produce brief summaries by extracting the most informative sections of the text by fine-tuning Pegasus on the BillSum dataset. It is also very effective in the process of capturing complex legal semantics and generating summaries that are close to human standards.

The Pegasus architecture is based on the traditional transformer encoder-decoder architecture, which has several encoder

stack and decoder stack layers with self-attention and feed-forward neural networks. The input document is then processed by the encoder which implements the self-attention mechanisms which calculate the relationship between all pairs of tokens at once allowing at least parallel processing and long-range dependencies to be captured without the bottleneck of recurrent models. The input is then transformed by each encoder layer by multi-head self-attention—enabling the model to attend to several representation subspaces at the same time—then by position-wise feed-forward networks and by residual connections with layer normalization. The decoder also uses the same self-attention as the encoder generated tokens and cross-attention, with causal masking to avoid attending future positions during autoregressive generation.

The difference between Pegasus and other transformer based models is that it uses pre-training methodology, and it is optimized on summarization. By tuning Pegasus on the BillSum data, we both use this pre-trained functionality and specialise the model to the particular language features, lexical norms, and formal attributes of legal texts. In the fine-tuning phase, the model gets to know how to identify indicators of importance that are specific to the law, like statutory definitions, operative clauses, amounts of appropriation and mechanisms of enforcement, that can vary significantly as compared with indicators of importance in the pre-training data of the model.

Pegasus showed great performance in our experiments in capturing intricate legal semantics and creating summaries of near-human quality. The model has been shown to have good abilities in detecting the essential provisions of long bills, giving the relative weight of different sections, the overall multi-topic legislation, and provide abstractive summaries that paraphrase rather than merely textually extract.

3) *Fine-tuning BART (Bidirectional and Auto-Regressive Transformer)*: The combination of bidirectional and autoregressive transformers is in a way that BART is quite effective in text generation tasks. The model was optimized to recreate masked or corrupted text sequences to coherent summaries.

The BART model is composed of bidirectional encoder architecturally analogous to BERT and an autoregressive decoder that is architecturally analogous to GPT. With bidirectional attention to self-attention, the encoder processes all inputs in the input sequence where each token takes care of all other tokens in the input without directional restrictions. This bidirectionality is essential in contextual interpretation in that the model is capable of exploiting both left and right context to create representations of each token. The decoder produces output sequences in the autoregressive manner of the left-to-right generation, taking into consideration not only the previously generated tokens with the help of the masked self-attention but the encoder outputs as well with the help of the cross-attention mechanisms. This encoder-decoder architecture simply fits the task of summarizing full text, in which the entire input document needs to be properly understood (by the encoder) and then a summary version of the document has to be constructed (by the decoder).

Fine-tuning BART with the BillSum dataset on legal sum-

marization: Summarizing general text generation capabilities, this model is used to adapt to the needs of the specific task of legislative documents. Fine-tuning The fine-tuning process learns the patterns of legal discourse, the structure of the statutes, the legal entities and provisions to consider, and produces a summary of the law that balances formality and readability. The bidirectional encoder used by BART is particularly useful when working with the complicated nested clauses and cross-references used in legal documents as it is able to look at the context in multiple directions simultaneously when encoding any given element.

BART generates better readable and sentence fluency summaries than other abstractive models and can generate grammatically correct natural flowing text that is more accessible to non-experts without losing the legal accuracy of the generated text. The model has been found to work well with very long bills, bringing together information from more than one section and producing coherent summaries that reflect the main purposes and mechanisms of the legislation. The fact that BART can strike a balance between an abstractive and a copying approach to generating content, i.e., the generation of new phrasings instead of copying the source text, and the preservation of content faithfully, places it in a well-placed situation in the context of professional legal summarization.

4) *Fine-Tuned T5 (Text-to-Text Transfer Transformer)*: T5 is a transformer model that considers all problems in NLP as text-to-text problems. The BillSum dataset has been fine-tuned on, with each text of a bill as input and the target output the summary. The unified architecture and the aid of T5 bidirectional attention allow it to produce coherent, fluent and context rich summaries, which are very much aligned to the original meaning of the legal documents.

The T5 architecture is based on the transformer encoder-decoder design but has many significant changes and training innovations. The model is fully encoder-decoder with both parts being transformer-based and self-attention mechanisms. The encoder operates in a two-way way, constructing contextualized representations of every token, based on attention to all other input tokens. The decoder produces the output text in an autoregressive manner applying both self-attention to the already generated tokens and cross-attention to encoder outputs. T5 uses relative positional encodings instead of absolute positions, which serves to make the model extend the performance to sequences of different lengths, as well as to longer documents.

The unified architecture and the ability to focus both ways in the encoder of T5 are especially effective in terms of summarizing the law. Bidirectional processing enables the model to form rich representations that take into account the complete context of every token, which is fundamental to comprehend legal texts in which meaning is frequently based on intricate associations amidst distant clauses, definitions and cross-references. The wide-ranging pre-training of the model gives strong linguistic knowledge and generation strength, whereas the text-to-text architecture allows natural specification of tasks and possible multi-task learning, through which

summarization might be added with the legal NLP tasks it might be related to.

#### IV. RESULTS AND DISCUSSION

This chapter shows a detailed discussion of the experimental findings of our comparative analysis of the deep learning models in legal text summarization on the BillSum data set.

TABLE I  
MODEL RESULTS

Model	ROUGE-1	ROUGE-2	ROUGE-L
BERT	0.47	0.22	0.24
Text Rank	0.38	0.20	0.31
Lex Rank	0.39	0.20	0.24
LSTM	0.74	0.55	0.74
PEGASUS	0.55	0.24	0.36
BART	0.35	0.14	0.21
T5	0.45	0.23	0.29

Table I displays the Recall-Oriented Understudy for Gisting Evaluation (ROUGE) scores obtained by both of the models implemented on the BillSum test set. ROUGE metrics quantify an overlap between generated summaries and reference summaries by matching n-grams, and offer standardized quantitative results on summarization quality. ROUGE-1 quantifies unigram overlap and the content coverage, ROUGE-2 quantifies the bigram overlap and the fluency and sentence level accuracy, and ROUGE-L quantifies the longest common sequence and structural similarity and sentence level coherence.

The LSTM-based encoder-decoder model performed better in all the assessment measures. These findings are especially impressive because LSTM is a relatively older recurrent model than the transformer-based one of our research. The ROUGE-1 score is 0.74 which means that about 74 percent of the unigrams in the reference summaries are found in the LSTM-generated summaries, that is, the content is well covered. The fact that the score of ROUGE-L is also high, at 0.74, indicates that the model is well aligned structurally with reference summaries, which does not damage the logical flow and structure of the information. The 0.55 ROUGE-2 score is lower than ROUGE-1, but nonetheless represents a significant amount of bigram overlap in the model, suggesting that not only isolated significant terms but also phrasing expressions and collocations as found in legal language are the product of the model.

The classic extractive algorithms TextRank and LexRank—performed fairly well, which is important as a benchmark. The mediocre performance of these extractive techniques confirms that such a quality of legal summarization cannot be achieved with the help of simple sentence selection. Although extractive methods have certain benefits like ensuring factual precision and computational efficiency, they cannot synthesize information across several passages, avoid repetition by paraphrasing, or reorganize information in a more coherent way, which the abstractive models can.

The intermediate performance is attained by Pegasus and there are a few reasons that can be attributed to the comparatively low performance of Pegasus in the legal field. To

begin with the pre-training of Pegasus, the pre-training mostly involved news articles, scientific papers, and general text domains of the web that are vastly different in terms of style, structure, and content features compared to the language of legislations. A typical news summary has more elements of who, what, when, and where that have straightforward event-based accounts, whereas a legal bill summary needs to cover the intricate workings of legislation, definitions, relationships between provisions, and legal implications. Fine-tuning will be constrained by this mismatch of domains between transfer.

The significant difference between the highest abstractive model and highest extractive method performance is an indication of the usefulness of generation-based methods in law summarization. The abstractive models are able to integrate data among various parts, reform complex legal texts into more easily understandable forms, remove repetition, and form coherent texts that read like natural accounts as opposed to merely joining together mined sentences.

There are however significant tradeoffs during this performance advantage. Abstractive models run the risk of factual errors or hallucinating information that was never there in the source documents or insidiously changing the legal meaning by paraphrasing legal information, which is especially worrisome in legal uses of such models, where accuracy is paramount. Extractive approaches are less versatile and coherent, but ensure that all the summary content is verbatim to the source, removing the possibility of errors in generation. In the case of critical legal uses, a compromise between accuracy and coherence between extractive identification of important content and limited abstractive generation could be provided by hybrid techniques.

In addition to the quantitative scores of ROUGE, the qualitative analysis of the generated summaries shows significant trends of the model behavior as well as gives the information about its practical usage.

Summaries produced by Pegasus are more abstractive, and the paraphrasing and reorganization of contents are more frequent than those produced by LSTM. In many instances, the summonses have been effective in expressing in more natural and accessible terms the general intent and major provisions of bills. Pegasus, however, occasionally leaves out some important technical information, or dollar amounts, or even legal terminology in favor of more general statements, which is worrying as far as legal applications are concerned, where accuracy is fundamental.

## V. CONCLUSION

The paper has discussed the concept of extractive and abstractive text summarization methods on domain-specific text, especially the law. Our goal was to compare deep learning and transformer-based models in terms of their ability to encode the semantic meaning of long and information-rich texts. The classical extractive models such as TextRank and LexRank provided a good starting point but could not understand what lies deeper in the context and legal details. Conversely, abstractive models (including BART, PEGASUS

and variants of the fine-tuned versions of BERT) proved to be better at producing coherent, fluent, and contextualized summaries. These models were able to learn legal terminology and legal relations in complicated statutes, and this was an important advance in the development of intelligent legal NLP systems. The findings highlight the idea that extractive and abstractive methods are factually accurate and more readable, respectively. Finally, the domain-specific transformer-based architectures can be refined to fine-tuning on domain specific legal corpora is a paradigm shift toward building high-fidelity context-aware systems in terms of summarization of the legal domain.

There are a number of areas that the future research should center on to improve the legal summarization systems. To start with, a combination of rhetorical role classification and legal knowledge graphs could enable models to give priority to some important sections such as the Issues or Holdings. Second, pre-training on a variety of legal corpora may also be quite effective at developing legal-specific language models. Lastly, study on model explainability and bias reduction will be most crucial to the extent that a given model produces a summary, it can be justified to a particular user in the event that the summary is not only correct but also ethically justifiable and acceptable by lawyers which will eventually prove beneficial in more adaptations.

Investigating extractive-abstractive methods hybrids would be a way to balance between accuracy and readability. Longer documents might be better served by researching such specialized architectures as hierarchical transformers or retrieval-augmented generation (RAG) systems. Lastly, the explainability of models and bias mitigation research is necessary to establish trust in legal practitioners and defendable but ethical summaries that further facilitated its use in real-world practice.

## REFERENCES

- [1] G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” *Phil. Trans. Roy. Soc. London*, vol. A247, pp. 529–551, April 1955.
- [2] J. Clerk Maxwell, *A Treatise on Electricity and Magnetism*, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
- [3] I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in *Magnetism*, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.
- [4] K. Elissa, “Title of paper if known,” unpublished.
- [5] R. Nicole, “Title of paper with only first word capitalized,” *J. Name Stand. Abbrev.*, in press.
- [6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” *IEEE Transl. J. Magn. Japan*, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].
- [7] M. Young, *The Technical Writer’s Handbook*. Mill Valley, CA: University Science, 1989.
- [8] S. Undavia, A. Meyers, and J. E. Ortega, *A comparative study of classifying legal documents with neural networks*. New York, NY: Institute of Electrical and Electronics Engineers Inc., 2018.
- [9] A. Vaswani et al., *Attention Is All You Need*. Advances in Neural Information Processing Systems, 2017.
- [10] B. Zhang, D. Xiong, and J. Su, *Accelerating Neural Transformer via an Average Attention Network*. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, 2018.

- [11] C. Y. Lin, ROUGE: A Package for Automatic Evaluation of Summaries. ACL Workshop, 2004.
- [12] K. V. Rajkumar et al., Scalable Web Data Extraction for XtreeAnalysis: Algorithms and Performance Evaluation. Bangalore, India: Second International Conference on Inventive Computing and Informatics (ICICI), 2024.
- [13] A. F. V. Hout, A. J. S. van der Meer, and W. H. M. P. van de Weerd, A systematic review of automated text summarization of legal documents: current state and the way forward. Computer Science Review, 2021.
- [14] A. Kanapala, S. Pal, and R. Pamula, Text summarization from legal documents: a survey. Artificial Intelligence Review, 2019.
- [15] V. N. Thatha, A. SudhirBabu, and D. Haritha, Privacy-preserving smart contracts for fuzzy WordNet-based document representation and clustering using regularised K-means method. International Journal of Ad Hoc and Ubiquitous Computing, 2022.
- [16] R. Mihalcea and P. Tarau, TextRank: Bringing Order into Texts. Denton, TX: University of North Texas, 2004.
- [17] R. C. Kore et al., Legal Document Summarization Using NLP and ML Techniques. Mumbai, India: Vidyalankar Institute of Technology, 2023.
- [18] S. Zahra, U. Chauhan, I., and A. I. Abidi, Legal Document Summarizer. Preprints.org: [Preprint Platform/Organization], 2025.
- [19] E. Bauer, D. Stammbach, N. Gu, and E. Ash, Legal Extractive Summarization of U.S. Court Opinions. Rome, Italy: LIRAI'23, 2023.
- [20] U. Dixit, S. Gupta, D. Yadav, and A. K. Yadav, Recent Advances in DL-based Text Summarization: A Systematic Review. ICACITE: IEEE, 2023.
- [21] P. V. Preeja, T. S. Subashini, and J. S. Jayasudha, Exploring Deep Learning Approaches for Text Summarization and Text Generation. ICUIS: IEEE, 2023.