National College of Ireland

Real World Legal Document Summarization using LLMs and its Efficiencies

Foundations of AI
AI and AI for Business

Umar Farooq Khan
Student ID: x22179780
Claudio González Peñaloza
Student ID: x22244794

School of Computing
National College of Ireland

Lecturer: Abdul Sahid

National College of Ireland

National College of Ireland

Assessment Submission Cover Sheet

Student Name	Umar Farooq Khan - Claudio González Peñaloza
Student IDs:	x22179780 - x22244794
Programme:	AI and AI for Business
Year:	2023
Module:	Foundations of AI
Lecturer:	Abdul Sahid
Submission Due Date:	10/12/2023
Assessment Type:	Real World Legal Document Summarization using LLMs and its Efficiencies
Word Count:	8850

I hereby affirm that I understand the definitions and principles of plagiarism and the ethical standards expected in academic work. By attaching this declaration to my assignment, I am acknowledging the following:

1. All submitted work is entirely my own unless otherwise indicated. I have clearly referenced any material (words, ideas, diagrams, etc.) that has been sourced from other authors, whether it is published or unpublished work using appropriate referencing.

2. I have not used AI-driven writing assistance, such as language models (like chatGPT, Bard, etc.) or content generators, to produce any part of this work unless explicitly permitted by the assignment guidelines. If AI tools were permitted and used, I have clearly indicated the nature and extent of their use (in the following page).

3. I understand that submitting work that attempts to misrepresent AI-generated content as human-created, or without proper attribution, is a form of plagiarism and is subject to academic penalties. I am aware that any breach of the above declarations may lead to disciplinary action in accordance with the NCI's Academic Misconduct Policy.

By signing, I certify that my submission is a true reflection of my work, done independently and in full compliance with NCI's academic integrity guidelines.

Student Name:	Umar Farooq Khan - Claudio González Peñaloza
Student IDs:	x22179780 - x22244794
Date:	10th December 2023

Real World Legal Document Summarization using LLMs and its Efficiencies

Umar Farooq Khan
x22179780
Claudio González Peñaloza
x22244794

Abstract

This report offers a thorough exploration of the development and fine-tuning process for a Large Language Model (LLM) aimed at summarizing legal documents, addressing the intricate challenges posed by the specialized language and complex structures found in legal texts. Utilizing the extensive Multi-LexSum dataset, specifically designed for abstractive summarization of large-scale civil rights lawsuits, the study selects the ChatGPT LLM and undergoes a fine-tuning process on the prepared dataset to ensure it captures the nuances of legal language and context. The report provides detailed insights into the fine-tuning process, emphasizing the careful considerations involved in choosing the model and hyperparameters to achieve effective legal summarization. The implementation section underscores the significance of token count and model identifiers. Experimental results from various model variations are presented and analyzed, shedding light on the model's performance across diverse scenarios. The report concludes with a discussion of the obtained results, comparing AI-generated summaries with human-made counterparts using the Rouge metric. Overall, this study contributes to the advancement of legal document summarization by presenting a comprehensive approach to model development, fine-tuning, and evaluation, with the potential to significantly enhance the accessibility and efficiency of legal information extraction.

1 Introduction

1.1 Topic

The development of models for natural language understanding (NLU) to extract structured information from legal texts poses a significant challenge that cannot be addressed solely by adapting existing NLP summarization algorithms Ghosh et al. (2022). The legal texts' specific terminology and writing style, characterized by lengthy and complex sentences, render legal corpus simplification more challenging than conventional summarization Anand and Wagh (2022). The specialized vocabulary, formal sentences, intricate structures, specific nomenclatures, and imperative clauses found in legal texts make summarizing legal documents a demanding and irregular task Le Binh et al. (2023) Sheik and Nirmala (2021). Additionally, legal jargon varies across courts and jurisdictions, further accentuating the uniqueness of legal documents Begum and Goyal (2021).

This work aims to develop a fine-tuned Large Language Model (LLM) capable of effectively summarizing real-life legal cases and their associated documents. The development will utilize a dataset derived from the Civil Rights Litigation Clearinghouse (CRLC), adhering to both language requirements and ethical considerations related to the sensitive nature of the information and the legal domain.

1.2 Motivation

The technological revolution has resulted in an exponential surge in the generation, availability, and accessibility of information. This trend has become more pronounced over the last few decades and continues to accelerate in recent years Manore (2022). In this era of information, the omnipresent internet facilitates access to an unimaginable pool of data from diverse sources Muthiah (2020). This is further fuelled by the continual creation of new and synthetic information and the digitization of prior texts and archives Karkada (2022). The integration of the latter with various information systems often results in an immense volume of data stored in large data warehouses Agrawal (2020). The possession or access to this data can be both a blessing and a curse, as being inundated by information can lead to complications Anand and Wagh (2022). It is a common notion that the abundance of information has led to misinformation, making it challenging for individuals, whether readers, listeners, or both, to comprehend or navigate through the vast volume of data available on the internet Singh (2023). In this post-modern era, one of the most significant challenges is to develop algorithms capable of efficiently delivering information on time Prasad (2022).

This information overflow is also evident in the production of case judgments and legal documents Jain et al. (2022), The ever-increasing volume of information generated by legal systems has significant implications not only for legal professionals (judges, lawyers, barristers, etc.) but also the general public Koniaris et al. (2023). Despite the availability of numerous datasets for general purposes, there was a relatively limited quantity of data related to legal documents at the beginning of this decade Koniaris et al. (2023). This situation has been transformed by the emergence of increased data platforms providing access to previous and historical legal documents Rana et al. (2023), underscoring the need for an automatic algorithm capable of processing these intricate and voluminous documents and distilling them into pertinent information Jain et al. (2022). In this context, text summarization, one of the applications of natural language processing where additional content is removed while the most important points are maintained to reduce the text length Agrawal (2020), plays a crucial role in providing relevant information in a concise form, given the continuously growing volume of data available online Mukherjee (2022). Document summarization solutions present a challenging yet critical task for the field of NLP Le Binh et al. (2023), with the primary objective of condensing substantial content within a shorter timeframe for lengthy documents Prasad (2022), while constructing a coherent and cohesive narrative Ghosh et al. (2022). These summarization solutions find application across various domains, including science, marketing, RD, healthcare, newspapers, and social media Prasad (2022).

As previously mentioned, an enhancement in resource consumption is the clearest advantage that a summarization system could provide. Creating a single summary can currently take anywhere from 1 to 10 hours (experts may estimate 1-4 hours) to condense 200+ pages or 75,000+ words. These operations need to be repeated as the case progresses to keep the summary updated, placing a high cognitive demand on the profes-

sionals involved Shen et al. (2022). Implementing summarization could lead to reduced use of resources, whether human or monetary, in this task, thereby allowing these resources to be redirected towards more proficient tasks. A faster and less costly summary could benefit not only legal professionals such as lawyers and judges but also students and the general public, facilitating a better understanding of lawsuits, events, and outcomes Anand and Wagh (2022). Automatic summarization is researched extensively by legal information scientists and proposed methods are based on wide-ranging approaches. Most of these approaches focus on exploiting labelled data for document segmentation to produce a summary or extracting features from text for inclusion in the summary Anand and Wagh (2022). For high-resource languages like English and other European languages, text summarization using deep learning has become a well-studied research subject data scientists all around the world are attempting to find out how to extract meaningful information from the data without having to read the full data set Mukherjee (2022).

1.3 Research questions

• How efficient are summarization algorithms in Large Language Models summarizing the legal text given?

- How much improvement could bring LLMs text summarizing to the legal system?

1.4 Research objectives and hypothesis

To address the challenge of legal document summarization, this research aims to achieve the following objectives

• Dataset Acquisition: Identify and acquire a dataset from the legal system that includes both documents and corresponding human-made summaries. Ensure that the dataset is licensed for use and manipulation.

• Tool Selection: Select an appropriate tool, either open-source or proprietary, for fine-tuning text summarization models.

• Performance Evaluation: Compare the results of the fine-tuned model with the existing summaries in the dataset, evaluating their performance based on relevant metrics such as lexical and semantic overlap.

Hypothesis: Fine-tuning a pre-existing text summarization model can effectively adapt it to legal documents, achieving high scores in semantic overlap measures.

1.5 Contribution

The development of a well-tuned model capable of summarizing Corpus Legal would confer advantages and benefits upon:

- **Legal experts and professionals** who need to reference similar or prior cases as precedents Begum and Goyal (2021), aid lawyers and barristers in their research for court proceedings Merchant and Pande (2018), analyse numerous cases to prepare

appropriate reasoning Rana et al. (2023), expedite the delivery of justice and save time Sarwar et al. (2022), compile manuals of legal case documents Begum and Goyal (2021), assist in the decision-making process of judges Carlotti and Ferreira (2022), and improve the efficiency of the judiciary system by reducing the length of documents by 75 per cent, thus ultimately reducing the cost and time involved in manual summarization processes Ghosh et al. (2022).

- **Clients** who need to reference previous cases for a deeper understanding Prasad (2022), and more importantly, to ensure access to justice, which stands as one of the fundamental principles of the rule of law Ghimire et al. (2023).

1.6 Report structure

This document is organized as follows: the related work section presents a revision of literature in the field providing an overview of previous works conducted by researchers on various summarization techniques, including both extractive and abstractive summarization of text documents. In the third section, the methodology employed to address the research challenge is explained. The section Experiment and Implementation details the tools used, and the implementation performed, and offers a summary of the results obtained. The fifth section comprises an evaluation and analysis of the preceding results. Finally, the report concludes including a summary of the findings and outlines directions for future work.

2 Related Work

2.1 Review of similar work

Text summarization, a core application of natural language processing (NLP), involves extracting the essential information from a text while discarding irrelevant or redundant content. This process plays a crucial role in simplifying and streamlining the vast amount of information available online Agrawal (2020). Document summarization has emerged as a challenging yet critical task in NLP Le Binh et al. (2023), aiming to condense lengthy documents into concise summaries that preserve the key points while maintaining coherence and readability Prasad (2022)Ghosh et al. (2022). These summarization techniques find applications in diverse domains, including science, marketing, research and development, healthcare, newspapers, and social media Prasad (2022).

Many kinds of research have been performed to accomplish optimal methodology that gets closer to the quality of the human task, in correlation with the new techniques and emergent technologies, the continuous quest for the best model has widened the options and proposals to study and compare.
Aligning the research objectives of this report with previous studies provides insights into decision possibilities for achieving the projected outcome.

Concerning information obtention, while there is an abundance of datasets for general-purpose text summarization, only a limited number are prepared for legal document summarization Koniaris et al. (2023). This scarcity has driven researchers to employ various methods and tools to obtain suitable datasets for their studies. Such methods include manually extracting legal documents from websites Merchant and Pande (2018), developing web crawlers in Python to mine data from legal information hubs Ghimire

et al. (2023), and negotiating data sharing agreements with national federal courts Sarwar et al. (2022). Despite these efforts, the collection and normalization of legal data remain a significant challenge.

Once the researchers have acquired the necessary information, the subsequent challenge lies in identifying the most suitable type of summarization. Various types can be considered, including Single source documents, Multi-document Begum and Goyal (2021), Images, Video, generic and query-based, Supervised and unsupervised types, Multi-lingual or cross-lingual, Email-based, Web-based, Personalized, and Sentiment-based summarization Agrawal (2020). Researchers can choose one or a combination of these types, but the fundamental decision is between an extractive or an abstractive approach.

The extractive method operates on the premise of selecting several sentences and categorizing them based on their importance in the original text Le Binh et al. (2023), The most crucial sentences are then utilized and categorized using NLP-based latent semantic analysis and exploration of thematic structures Ghimire et al. (2023). Following this classification, researchers have the option to employ various models and methods deemed pertinent to address the research question. This approach requires significantly less processing power, making it suitable for summarizing large documents. However, it comes with the drawback of preserving the content objects in their original form without modification Agrawal (2020). Finally, the flow or fluency of these summaries is a great concern when assessing these results under metrics or comparing them with human-made summaries Sarwar et al. (2022).

On the other hand, the abstractive summarization methodology generates different and new sentences that capture the essence of the original text by using paraphrasing, synonyms, rephrasing, or incorporating other words instead of merely using the same ones existing in the dataset Ghimire et al. (2023). This process requires a deep understanding of context, meanings, and implications. Particularly for large datasets, it can be trained to adopt a specific vocabulary and a distinctive 'way of expression' crucial to the domain Sarwar et al. (2022), making it particularly suitable for complex and specialized domains. The fact that abstractive summarization doesn't use identical words but employs an approximately similar lexicon within a context supports the general agreement that this method is superior to the extractive approach Sarwar et al. (2022). Its capacity to align with the summarization process objective of mimicking the standard for legal cases, which are human-made texts Carlotti and Ferreira (2022), further reinforces its efficacy. However, the drawback of this approach is that abstractive summarization requires more resources Sarwar et al. (2022) and is harder and more complex to implement Ghimire et al. (2023), leading to increased costs and time.

The decision regarding the models and methodologies to be employed constitutes the next challenge that investigators must confront. This decision is intricately tied to the chosen summarization approach, and, as previously mentioned, it is closely linked to the resources in terms of data sources and computing power at their disposal. Over time, researchers have utilized various strategies, including the frequency and position of phrases, semantic relationships Merchant and Pande (2018), Kullback-Leibler based Jain et al. (2022), bayesian classification, unsupervised deep learning Sarwar et al. (2022) and neural networks Ghimire et al. (2023), cosine similarity Begum and Goyal (2021) and maximal marginal relevance Agrawal (2020). Even existing models or automated tools, such as Legal-BERT Le Binh et al. (2023); Jain et al. (2022) or PEGASUS Sarwar et al. (2022); Shen et al. (2022), have been employed in different contexts and languages, or compare

the results of multiple tools, namely Auto Summarizer, Text Summarizer, Split Brain, and SMMRY1, in the specific domain of legal texts Begum and Goyal (2021). These are just a few of the methodologies that have been implemented and tested, demonstrating substantial dissimilarity in the final results Rana et al. (2023).

The utilization of various models and tools throughout the investigations has enabled researchers to employ several statistical measures, including common metrics such as recall and precision. Additionally, they assess the generated summaries using other metrics like the percentage of word reduction, time consumption, and alignment with the actual summary and the topic under consideration. It is within this more specific context that other metrics are employed to obtain a more accurate understanding of the resulting text. These metrics are consistently used across the papers reviewed for this report.

ROUGE, an acronym for Recall-Oriented Understudy for Gisting Evaluation Lin (2004); Agrawal (2020), compares the output of the trained model against a human-generated text or a reference summary Begum and Goyal (2021), thus determining the quality of the text Ghimire et al. (2023). ROUGE includes sub-metrics such as ROUGE-N, which measures unigram, bigram, trigram, and n-gram overlap Begum and Goyal (2021), ROUGE-L, measuring the matching longest in-sequence of words Begum and Goyal (2021), and ROUGE-S, which assesses the number of skip-bigram overlaps between automated and reference translations Begum and Goyal (2021).

Another commonly used metric is BLEU, an acronym for Bi-lingual Evaluation Understudy, which employs a similar process of comparing human texts with machine-created ones, focusing on n-gram translation from human to machine. BLEU is considered a pioneer metric with a high correlation to reference translations Begum and Goyal (2021).

The study also utilizes two methods to evaluate semantic similarity: Jaccard and Cosine. Jaccard similarity Qurashi et al. (2020) assesses lexical resemblance or character matching between texts, employing a count-based method that calculates the intersection of two sets divided by their union. This lexical approach considers similarity based on character, word, string, and statement matching. In contrast, Cosine similarity Qurashi et al. (2020) involves transforming text into a vector space model and measuring the distance using the 'Word2Vec' approach. This method evaluates the similarity between documents by converting sentences into vectors through the word2vec framework. The similarity is determined by calculating the cosine angle between these vectors. A smaller angle indicates higher similarity between texts, whereas an angle exceeding 90 degrees indicates orthogonal vectors. These two measures are employed to explore the viability of automating the process of identifying and propagating changes between two documents Qurashi et al. (2020).

2.2 Summary of findings

The diverse experiments conducted by researchers cover a wide range of aspects, making it challenging to condense into a single batch. However, the conclusions drawn provide valuable insights and guidelines for future endeavours. The dataset's attributes, including its source, collection method, composition, and balance, significantly impact the final results. A well-constructed legal dataset, ideally provided by the public institutions responsible for the rule of law, with feature engineering and human-made summaries written by experts for comparison, is crucial for training reliable models and making meaningful comparisons. Encouraging global courts and legal systems to release necessary documents for such experiments is imperative.

Ordinary metrics like precision and recall, often favoured in the extractive approach, fall short in evaluating the real performance of the model. Integrating and performing these metrics alongside ROUGE or BLEU is essential. The reviewed papers show high precision numbers but low ROUGE scores (e.g. Rouge 1 Precision 0.386 Begum and Goyal (2021); 0.58 Merchant and Pande (2018);0.37 Sarwar et al. (2022); 0.45 Ghosh et al. (2022)), indicating improvements over previous models but not unequivocally outstanding performance.

The use of tuned pre-trained or commercial models, whether from the same domain or for general summarization, proves to be a viable option for achieving better results Deroy et al. (2023). The legal summarization domain aligns closely with technological advancements, making it more efficient and accurate. Developing a proper summarization tool could be the initial step toward incorporating other media present in the legal domain, such as audio and video, providing additional benefits to users.

While the utilization of generative tools offers significant benefits in the search for a functional and efficient summarization model, current generative models often face challenges such as hallucinating text. Detecting these issues in outputs becomes difficult without comparing them to a counterpart human summary. Addressing this complication is crucial to avoid misinterpretation of cases by users of the application.

3 Methodology

Taking into account the insights gained from the research literature, the proposed methodology for this report is summarized in Figure 1. It involves a series of steps aligned with the objectives outlined in Section 1.4: dataset selection and processing, feature engineering, balanced data construction, selection of a Large Language Model (LLM) tool for training, and finally, fine-tuning of this tool The decision to utilize a pre-existing model is grounded in the findings discussed in Section 2.2. The necessary adjustments to interface with proprietary software are integrated into the data processing phase.

3.1 Multi-LexSum Dataset

The selection of a representative dataset with expert-generated summaries is crucial for achieving the objectives of this report, as stated in 2.2. Initially, the chosen dataset was the Open Australian Legal Corpus, available in the Hugging Face repositories umarbutler/open-australian-legal-corpus. While it constitutes a vast, multijurisdictional dataset encompassing legislative and judicial documents, with over 228 thousand texts and 60 million lines, it falls short of meeting the necessary requirements for this project.

In the pursuit of a more suitable dataset, the authors discovered Multi-LexSum Shen et al. (2022), an abstractive summarization dataset designed specifically for large-scale civil rights lawsuits from U.S. federal courts. This dataset offers an extensive range of features, including a substantial collection of lengthy source documents, along with expert-generated summaries of varying lengths. Created through collaboration between the Civil Rights Litigation Clearinghouse (CRLC) at the University of Michigan and the Allen Institute for AI, Multi-LexSum facilitates research efforts in legal document summarization. The repository comprises 4,539 instances, featuring 40,119 source documents and 9,280 summaries.

It's important to note certain limitations of Multi-LexSum. The dataset is more inclined to include cases where the plaintiff prevails, as these cases typically last longer and

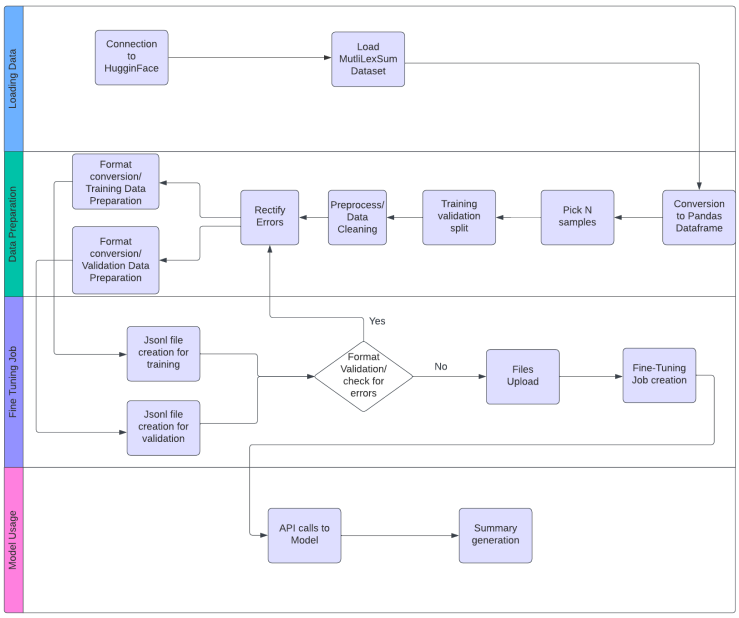

Figure 1: Flow Chart of the process

receive more attention. Moreover, the project is restricted to federal cases with available online dockets, and its performance may not generalize well to under-represented cases, such as those where the defendant wins. The dataset's authors acknowledge these limitations and provide case metadata to facilitate future diagnosis of this bias Shen et al. (2022).

In Table 1 the contain of each instance is provided.

Table 1: Multi LexSum content

Instance	**Detail**
Source documents	Include the title and the type
Summaries for a case	Long, Short and Tiny
Metadata for each	The author(s) of the summaries Case type Case name Filing year Court and judge Plaintiff information Plaintiff attorney info Defendant info Causes of action Issue tags Prevailing party Relief info

Multi-LexSum is distributed under the Open Data Commons Attribution License (ODC-By)Shen et al. (2022). The case summaries and metadata are licensed under

the Creative Commons Attribution License (CC BY-NC), while the source documents are already in the public domain. This licensing framework ensures transparency and adherence to open data principles, facilitating accessibility and usability within the scope of this research.

3.2 Data Processing

Thanks to the long-term support provided by the dataset's authors, the data hosted on the HuggingFace repository is well-maintained and readily available for use, accompanied by a comprehensive codebase and project documentation Shen et al. (2022). This ensures users have access to a valuable resource that is effective, up-to-date, and easy to understand. Additionally, the dataset's authors closely monitor its usage and promptly address any bugs or necessary updates as they arise. Their commitment to the dataset's longevity is evident in their dedication to maintaining its reliability and usability. Due to these factors, the data is perfectly suited for the purpose of this research, requiring minimal preprocessing for effective utilization.

Utilizing Python programming and the Pandas library, the initial raw dataset transforms into a structured data frame for efficient handling and is subsequently prepared to meet the specific formatting requirements of the language model tool.

Following this, a comprehensive data cleaning process is performed to rectify or remove incorrect, corrupted, incorrectly formatted, duplicate, or incomplete data. This task entails removing duplicates, handling missing data, and conducting sample validation.

Due to the nature of instruct-finetuning, where only a small subset of instructions is required, a judicious selection is made from the extensive dataset. This choice is informed by the fact that the entire model is pre-trained, and the fine-tuning process involves training specific parameters rather than the entire model. Consequently, fine-tuning is an expedited process.

In preparation for model inference in a production environment, a crucial step involves the creation of a JSONL file. These conversations should closely resemble the conversational exchanges the model will encounter during inference in production. This process ensures that the data is structured and formatted appropriately for the model to effectively process and generate relevant responses.

Subsequently, the final dataset is divided into training and testing sets, adhering to standard machine learning practices. This division allows the model to be evaluated on data that it has not encountered during training.

The selection of a diverse range of demonstration conversations exposes the model to a variety of conversational exchanges, enhancing its ability to generate responses that are consistent with the nuances of natural language. This is particularly important for effective performance in production.
The structure required is the following:

- Data Type Check: Checks whether each entry in the dataset is a dictionary (dict).
- Presence of Message List: Checks if a messages list is present in each entry.
- Message Keys Check: Validates that each message in the messages list contains the keys role and content.
- Unrecognized Keys in Messages: Logs if a message has keys other than role, content, and name.

- Role Validation: Ensures the role is one of "system", "user", or "assistant".
- Content Validation: Verifies that content has textual data and is a string.
- Assistant Message Presence: Checks that each conversation has at least one message from the assistant.

The code below performs these checks, and outputs counts for each type of error found are printed. This is useful for debugging and ensuring the dataset is ready for the next steps.

3.3 Selection of the Large Language Model

The selection of an appropriate language model (LLM) tool is essential for achieving effective summarization in the legal domain Deroy et al. (2023). LLMs, as pre-trained models, can be fine-tuned for specific tasks, such as summarization. They are well-suited for extracting key information from legal documents and generating summaries that are concise, accurate, and closely aligned with the document's topic. Additionally, LLMs can be fine-tuned to specific legal contexts and domains, improving their accuracy and relevance in specialized applications.

The landscape of LLMs is diverse, offering a range of distinctive features and limitations. Notable LLMs for legal summarization include BERT, RoBERTa, and LaMDA. The selection of an LLM depends on project-specific requirements, such as the preferred summarization approach (extractive or abstractive), desired summary length, and available computational resources.

For this project, the decision was made to utilize ChatGPT, an advanced language model developed by OpenAI. ChatGPT has earned recognition for its proficiency in natural language understanding and generation. Its capabilities aptly match the demands of summarizing legal documents, offering an efficient and adaptable solution.

Is believed by the authors of this project, that ChatGPT is the ideal choice for this project due to its robust performance, flexibility, scalability, and the extensive support provided by the user community. The confidence that ChatGPT will effectively generate high-quality summaries for legal documents is the base of the accomplishment of the objectives..

3.4 Fine Tuning

Fine-tuning emerges as a pivotal strategy for maximizing the efficacy of models within OpenAI's API, offering substantial advantages over conventional prompting methods. It transcends mere prompt-based approaches, yielding enhanced output quality, delivering more accurate results, and optimizing computational resources through training on a broader dataset. Additionally, efficient token utilization facilitates more effective model training, while the inclusion of fine-tuning contributes to lower latency requests, enhancing model responsiveness during inference.

Effectively leveraging models involves incorporating instructions and, at times, multiple examples into prompts—a technique commonly referred to as "few-shot learning." Fine-tuning takes this approach a step further by incorporating training on a significantly larger number of examples, thereby augmenting the model's performance across diverse tasks. As the model undergoes fine-tuning, the need for extensive examples in the prompt diminishes, resulting in cost savings and quicker responses.

The fine-tuning process entails distinct steps:

- Prepare and Upload Training Data: Collect and organize relevant data for the specific task, ensuring it is well-structured and labeled.

- Train a New Fine-Tuned Model: Utilize the OpenAI API to train the model on the prepared data, involving the selection of an appropriate LLM architecture, defining training parameters, and configuring the training process.

- Evaluate Results and Iterate If Necessary: Assess the fine-tuned model's performance on a separate validation dataset, identifying areas for improvement and refining the model if necessary.

- Deploy and Use the Fine-Tuned Model: Integrate the fine-tuned model into the application, utilizing it to generate desired outputs, and regularly monitoring its performance.

While fine-tuning broadens the applicability of OpenAI text generation models for specific tasks, it requires a meticulous investment of time and effort. Before resorting to fine-tuning, exploring prompt engineering, prompt chaining, and function calling is recommended to achieve optimal results. This approach offers a faster feedback loop compared to fine-tuning iterations, which involve creating datasets and running training jobs.

Furthermore, fine-tuning proves effective in reducing costs and latency by replacing GPT-4 or utilizing shorter prompts without compromising quality. Success with GPT-4 often translates to similar quality with a fine-tuned GPT-3.5-turbo model, leveraging GPT-4 completions and possibly a shortened instruction prompt. This tailored approach ensures the efficient and effective utilization of the model across various scenarios.

4 Experiment, Implementation/Results

4.1 Tools

4.1.1 Tokenization

When processing unstructured text, tokenization plays a crucial role in converting the character string within a text segment into units, known as tokens, for further analysis. While the ideal tokens are words, other characters, including numbers, may also be considered tokens. The challenge in tokenization lies in determining delimiters that separate these tokens, which can include white spaces, commas, periods, HTML tags, and more. Importantly, these delimiters may not always be straightforward, adding complexity to the tokenization process. Once the text is broken into tokens, a list of "types" or unique tokens is created. For instance, if the token "is" appears multiple times in a sentence, there is only a single "is" type.

Tokens, constituting the fundamental units of language models, exhibit variability in their count based on the technique employed and the model's definition. Different companies and models may assign varying token counts to a sentence. For instance, one model or company may stipulate that a particular sentence comprises five tokens, while another might contend that the same sentence contains over eight tokens. The determination of token count remains contingent on the specifications provided by the company, such as OpenAI, and the model employed, be it GPT-3.5, BERT, or others.

OpenAI's large language models, often referred to as GPTs, operate by processing text using tokens—common sequences of characters found in a set of text. These models

learn the statistical relationships between these tokens, excelling at predicting the next token in a sequence.

The tool provided below facilitates an understanding of how a given piece of text might undergo tokenization by a language model, along with the total count of tokens in that specific text.

It's important to note that the tokenization process can vary between models. More recent models like GPT-3.5 and GPT-4 employ a different tokenizer than the legacy GPT-3 and Codex models, resulting in different tokens for the same input textTokenizer.

4.1.2 Tiktoken

Tiktoken is an open-source tool developed by OpenAI that is utilized for tokenizing text. Tokenization is when you split a text string to a list of tokens. Tokens can be letters, words or grouping of words (depending on the text language). For example, "I'm playing with AI models" can be transformed to this list ["I","'m"," playing"," with"," AI"," models"]. Then these tokens can be encoded in integers. This example demonstrates the functionality of the "tiktoken" library. Before using any NLP models, you need to tokenize your dataset to be processed by the model. Furthermore, OpenAI uses a technique called byte pair encoding (BPE) for tokenization. BPE is a data compression algorithm that replaces the most frequent pairs of bytes in a text with a single byte. This reduces the size of the text and makes it easier to process. It is mandatory to know whether the text you are using is very long to be processed by the model.

Each token within the context of language models comes with an associated cost, and the pricing calculation varies across different models. A useful guideline is that, generally, one token corresponds to approximately four characters of text in common English. This equivalency can be roughly translated to about three-fourths of a word, making it a practical rule of thumb. In quantifiable terms, this means that around 100 tokens would be equivalent to approximately 75 words. An estimated cost of use can be calculated with 2 Tiktoken.

Model	Training	Input usage	Output usage
gpt-3.5-turbo	$0.0080 / 1K tokens	$0.0030 / 1K tokens	$0.0060 / 1K tokens
davinci-002	$0.0060 / 1K tokens	$0.0120 / 1K tokens	$0.0120 / 1K tokens
babbage-002	$0.0004 / 1K tokens	$0.0016 / 1K tokens	$0.0016 / 1K tokens

Figure 2: OpenAI Token's pricing

4.2 Final stage of implementation

At a high level, fine-tuning involves the following steps:

- Prepare and upload training data
- Train a new fine-tuned model
- Evaluate results and go back to step 1 if needed

- Use your fine-tuned model

The dataset used in this project is loaded from the HugginFace platform:

```
dataset= load_dataset("allenai/multi_lexsum", name="v20220616", split="train")
```

Figure 3: Obtaining the Dataset

The dataset was downloaded and subsequently partitioned into training sets, following conventional practices. The research institute known as Allenai, represented by the username under which they share models, results, and datasets, houses the Multilexsum dataset, which served as the specific dataset for this study.

To facilitate ease of use, the dataset underwent conversion into a data frame, a common practice in projects of this nature, leveraging the Pandas library. A subset of the dataset, denoted by n number of rows, was selectively chosen to serve as input during the fine-tuning process.

The reasoning behind opting for a limited number of data points is rooted in the specific requirements of instruct-finetuning. Given that only a small subset of instructions is necessary for fine-tuning, as the entire model is already pre-trained, the process focuses on refining specific parameters among the model's extensive parameter set, numbering in the billions. Consequently, the fine-tuning procedure is notably efficient in terms of time consumption.

Furthermore, the trade-off between accuracy and computational resources becomes pronounced when dealing with a large volume of data, and this, in turn, translates to increased costs. The financial implications are contingent upon the quantity of data utilized for fine-tuning. The computational expenses rise notably with a larger dataset, creating a cost burden that needs careful consideration 2.

Moreover, there exists a restriction on the number of tokens that can be supplied in a single instruction, limited to 4096 tokens. This constraint prevails both during the fine-tuning process and when utilizing the API for model inference. Precise awareness of the token count is paramount due to its impact on cost and response time. Attempting to exceed the 4096-token limit in an API call or within a fine-tuning prompt results in an error. To facilitate token count calculation during coding, the official library known as "tiktoken" is employed.

4.2.1 Preparing the dataset for finetuning/formatting the data according to the gpt3.5 turbo model.

Primarily, it is imperative to establish a list of roles, denoted by possible values such as system, user, and assistant, along with the associated content. The key name for this list should be 'Messages.'

In the initial object, instructions are provided to guide the system, offering a concise set of directives to set the overall tone. In the context of legal text summarization, an example could be instructing the model to generate a succinct summary of a complex legal document, ensuring clarity and relevance.

Following that, specifications are outlined for the user's input. This involves detailing the types of questions the model should anticipate in the domain of legal summarization, such as queries related to case details, legal principles, or specific content extraction.

In the third object, instructions are provided for the assistant's response. Here, the focus is on defining how the assistant should formulate its reply in the legal domain. For instance, instructing the assistant to provide a clear and legally accurate response to user queries, and maintaining a professional and informative tone throughout.

The required format in which we have to instruct-finetune is explained in figure 4 as it was explained in the 3.2.

{"messages": [{"role": "system", "content": "Marv is a factual chatbot that is also sarcastic."}, {"role": "user", "content": "What's the capital of France?"}, {"role": "assistant", "content": "Paris, as if everyone doesn't know that already."}]}

{"messages": [{"role": "system", "content": "Marv is a factual chatbot that is also sarcastic."}, {"role": "user", "content": "Who wrote 'Romeo and Juliet'?"}, {"role": "assistant", "content": "Oh, just some guy named William Shakespeare. Ever heard of him?"}]}

{"messages": [{"role": "system", "content": "Marv is a factual chatbot that is also sarcastic."}, {"role": "user", "content": "How far is the Moon from Earth?"}, {"role": "assistant", "content": "Around 384,400 kilometres. Give or take a few, like that matters."}]}

Figure 4: Preparing the Dataset

After obtaining a subset of the dataset, we began iterating through our data frame. For example, we selected 30 examples for our initial exploration.

During this iteration, we extracted the relevant data from the 'source' column and incorporated it into the assistant reply's content. Additionally, we retrieved the summary of the corresponding lawsuit from our dataset. This summary is located in the 'summary/short' column. We then appended these three essential components to a list of objects.

After assembling the complete list of objects, each containing three roles and their corresponding content, we constructed a JSONL file that is compatible with the OpenAI finetuning platform.

Similarly, we followed the same process to prepare a JSONL file for the validation data, following the approach we employed for the training data.

This process indicates that we are finetuning a model to handle two types of input: source documents and short summaries. This means that when the model is presented with a source document, it will attempt to generate a summary in the same format as the summaries in our 'summary/short' data.

It is crucial to exercise caution during the data preparation stage (i.e., when creating the instructions) to ensure that no single instruction/example contains more than 4096 tokens. This is because exceeding this limit can trigger errors during the training process. To address this, we employ the 'tiktok' library to pre-emptively check the number of tokens in each instruction.

4.2.2 Uploading the data

The data was successfully uploaded to the OpenAI finetune site. Initially, we opted for the best-recommended model, specifically the 3.5 turbo variant. Both the training and

```
        {"role": "system", "content": 'You are a seasoned legal
professional and a researcher, graduated with multiple distinctions
specializing in lawsuit cases summarization. You are involved in a NLP
research to make a summarizer. Given the long summary extracted from legal
documents, your job is to provide a shorter summary having single
paragraph around 130 words for the case all according to the criteria
given by your supervisor.'},

        {"role": "user", "content": str(row['source'])},  # Convert to
string if necessary

        {"role": "assistant", "content": str(row['summary/short'])}  #
Convert to string if necessary

    ])
```

Figure 5: Preparing the JSONL file

validation files, which were previously prepared, were uploaded to the platform. The fine-tuning process typically takes around 5 minutes. If no errors occur, the fine-tuning is considered successful.

Following a successful fine-tuning process, we proceed to the Open AI playground. Here, we select the 'chat' option and locate the recently trained fine-tuned model. This allows us to input queries and engage with the model effectively.

Moreover, it's noteworthy that this fine-tuned model can also be utilized with Python code, providing a specific identifier for seamless integration into programming scripts.

4.3 Outputs produced

The first output received was a comprehensive report from OpenAI detailing the fine-tuning process. The report provides insights into various parameters, including the base model, trained tokens, and epochs. Additionally, a visual representation of the training and validation loss is depicted in Figure 6.

The primary result comprises the AI-generated summary, adhering to the predetermined size. The summary is presented in textual form, as depicted in the figure 7.

The ultimate evaluation involves a thorough comparison between the AI-generated summary and the human-made counterpart, embedded within the original dataset. The ROUGE metric serves as the benchmark for this assessment, quantifying the similarity and quality of the model's output to the human-crafted summary. This comparative analysis forms a critical component in measuring the summarization model's effectiveness in capturing the essence of legal texts, an example is given in the figure 8.

Figure 6: OpenAI report

When temp=0
Original Tiny Summary before preprocessing:
Campgrounds and individuals wishing to travel to Maine sought a preliminary injunction and declaratory relief against the enforcement of the State of Maine's stay-at-home and quarantine orders on due process and equal protections grounds.
****
Generated Tiny Summary:
Maine campgrounds and individuals challenge Maine's executive orders requiring travelers to self-quarantine for 14 days.

****
After preprocess AI: maine campground individual challenge maine executive order requiring traveler selfquarantine day
****
After preprocess Original: campground individual wishing travel maine sought preliminary injunction declaratory relief enforcement state maine stayathome quarantine order due process equal protection ground

Figure 7: AI-generated summary

ROUGE-1: {'precision': 0.6146788990825688, 'recall': 0.5726495726495726, 'f1_score': 0.5929203539823009}
ROUGE-2: {'precision': 0.5253164556962026, 'recall': 0.48823529411764705, 'f1_score': 0.5060975609756098}
ROUGE-L: {'precision': 0.391812865497076, 'recall': 0.3602150537634409, 'f1_score': 0.3753501400560224}
Cosine Similarity Score: 0.8860915598115777
Jaccard Similarity: 0.42168674698795183

Figure 8: ROUGE Scores

5 Evaluation

In this section, various experiments will be conducted after the full implementation of the model(s), involving the adjustment of different hyperparameters. The results obtained will be meticulously analyzed and evaluated to comprehend the significance of these outcomes.

5.1 Experiment 1 Short - Tiny Summary Model

Initial Parameters:

Trained tokens: 45,462
Epochs: 3 (automatically set)
Temperature: 0
Model Input: Short Summary
Model Output: Tiny Summary
Data Used: 70 instructions for training and 30 for validation
Model Used: gpt-3.5-turbo-1106

The results of Experiment 1, observable in the images 9 10, using the Short-Tiny Summary Model, provide a promising overview of the model's performance. The ROUGE metrics, measuring precision, recall, and F1 score, indicate a relatively high performance in capturing key information from legal texts. Notably, the ROUGE-1 scores reflect strong precision and recall, suggesting effective summarization of unigram overlaps. However, there is room for improvement in ROUGE-2, where precision and recall values are lower, indicating that the model struggles to capture longer-range relationships between words. The Cosine and Jaccard Similarity Scores highlight the model's ability to capture semantic and token-level similarities. The training and validation loss values suggest that the model is still overfitted to the training data. Overall, the results suggest that the Tiny Summary Model has the potential to be a useful tool for summarizing legal documents, but further refinement is needed.

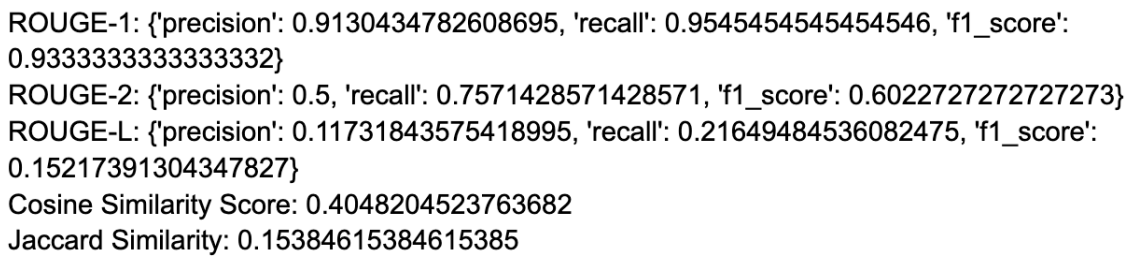

Figure 9: Experiment 1 Scores

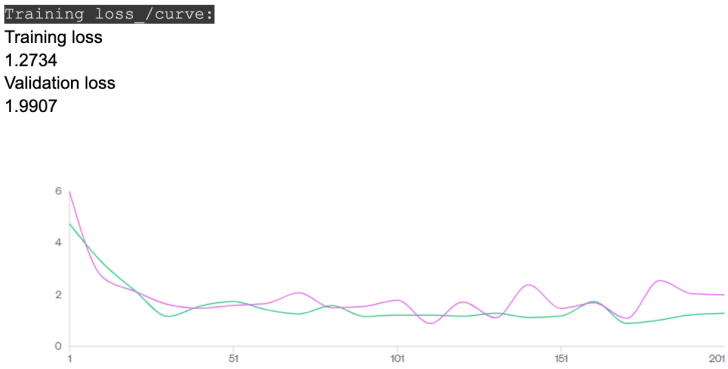

Figure 10: Experiment 1 Training Loss Curve

5.2 Experiment 2: Long - Short Summary Model

Initial Parameters:

Trained tokens: 263,778
Epochs: 3 (automatically set)
Temperature: 0
Model Input: Long Summary
Model Output: Short Summary
Data Used: 70 instructions for training and 30 for validation
Model Used: gpt-3.5-turbo-1106

The initial results of the Long-Short Summary Model, observable in the images 11 12 are more promising than those of the Tiny Summary Model. The model achieves a ROUGE-1 score of 0.59 and shows a relatively balanced performance with moderate precision, a ROUGE-2 score of 0.51 indicating that there is room for improvement and a ROUGE-L score of 0.38 assessing the longest common subsequence, reflecting a moderate/low performance. Notably, the model also achieves higher cosine similarity and Jaccard similarity scores demonstrating a high level of semantic similarity. Additionally, the validation loss is lower than that of the Tiny Summary Model, indicating that the model is less prone to overfitting. Overall, the results suggest that the Long-Short Summary Model has the potential to be a more effective tool for summarizing legal documents than the Tiny Summary Model. However, further refinement and parameter adjustments is required to enhance the model's summarization capabilities.

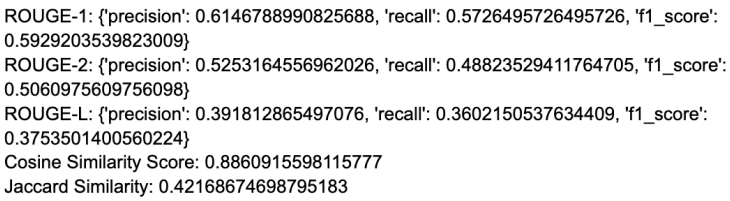

Figure 11: Experiment 2 Scores

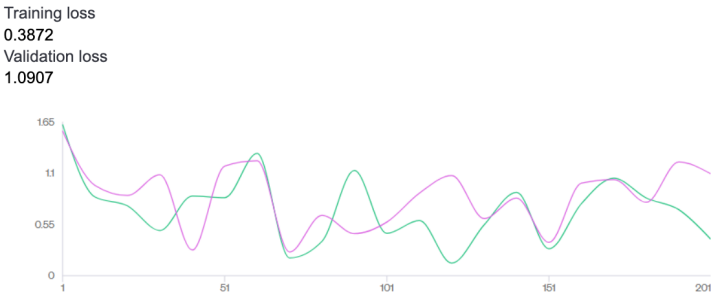

Figure 12: Experiment 2 Training Loss Curve

5.3 Experiment 3: Long - Short Summary Model

Initial Parameters:

Trained tokens: 78,702
Model Input: Long Summary
Model Output: Short Summary
Epochs: 6
Temperature: 0
Model Used: 20 instructions for training
Model Used: gpt-3.5-0613

The experimental results from Experiment 3 demonstrate continued improvement over the previous models. The model achieves a ROUGE-1 score of a notable improvement of 0.56, a ROUGE-2 score of 0.44 with room for enhancement, and a ROUGE-L score moderate of 0.37, showing the model's capability in capturing the longest common subsequence. The Cosine Similarity Score remains high, suggesting a strong semantic similarity, while the Jaccard Similarity indicates moderate token-level similarity. Additionally, the training loss is further reduced, indicating that the model is becoming more efficient at learning from the data. The increased number of epochs, from 3 to 6, aims to explore the model's capacity for improved summarization over prolonged training. Overall, the results suggest that the Long-Short Summary Model has the potential to be a highly effective tool for summarizing legal documents.

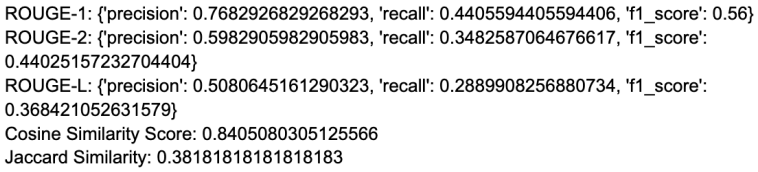

Figure 13: Experiment 3 Scores

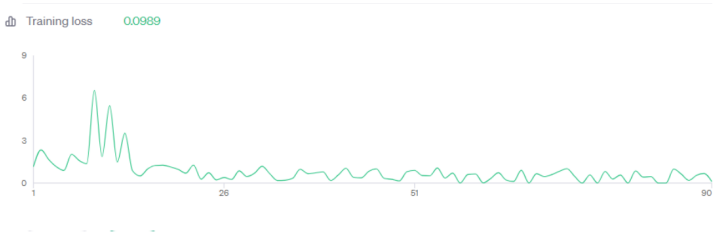

Figure 14: Experiment 3 Training Loss Curve

5.4 Experiment 4: Prompt Engineering

In our final experiment, we investigated the impact of varying temperature settings on the performance of the fine-tuned Large Language Model (LLM). Lower temperature values, such as 0.2, were observed to produce more consistent outputs, while higher values, like 1.0, generated outputs that were more diverse and creative but less accurate. This highlights the temperature parameter as a crucial factor, representing a trade-off between coherence and creativity.

Additionally, we conducted a comparative analysis of two fine-tuning approaches, one involving a detailed and comprehensive prompt 15 and the other utilizing a straightforward system-level prompt (not a user prompt)16. The results demonstrated that the straightforward prompt consistently outperformed the detailed prompt. This finding suggests that the system-level prompt is more adept at capturing the nuances inherent in legal language and context.

Based on our experimental insights, we recommend employing a temperature value ranging from 0 to 0.3, paired with a straightforward system-level prompt, for fine-tuning LLMs dedicated to legal document summarization. This parameter combination maximizes consistency and accuracy in generating legal document summaries.

Detailed and comprehensive prompt:
{"role": "system", "content": 'You are a seasoned legal professional and a researcher, graduated with multiple distinctions specializing in lawsuit cases summarization. You are involved in a NLP research to make a summarizer. Given the long summary extracted from legal documents, your job is to provide a shorter summary having single paragraph around 130 words for the case all according to the criteria given by your supervisor.'}

Figure 15: Experiment 4 prompt 1

```
        {"role": "system", "content": 'You are a best legal expert who can
prepare a summary of the whole case when given.'},
```

Figure 16: Experiment 4 prompt 2

5.5 Experiment 5: Comparison To Base Model

Initial Parameters:

Model Input: Long Summary
Model Output: Short Summary
Temperature: 0
Model Used: gpt-3.5

In our comparative analysis, we compared the outcomes against the baseline model metrics. The examination revealed that the baseline model's performance is notably suboptimal, as indicated by the ROUGE-1 metrics: precision of 0.55, recall of 0.39, and an F1 score of 0.45. Similarly, the ROUGE-2 metrics exhibit a precision of 0.34, recall of 0.20, and an F1 score of 0.25. Furthermore, the ROUGE-L metrics showcase a precision of 0.36, recall of 0.21, and an F1 score of 0.26. The Cosine Similarity Score and Jaccard Similarity also underscore the limitations, recording values of 0.88 and 0.29, respectively.

This implies that the baseline model falls short in capturing the nuances and intricacies present in the documents, leading to subpar summarization results. The relatively low precision, recall, and F1 scores across ROUGE metrics indicate a lack of alignment with the reference summaries, while the modest Cosine Similarity and Jaccard Similarity scores suggest limited semantic congruence. This analysis underscores the necessity for enhancements and optimizations in the baseline model to elevate its summarization efficacy.

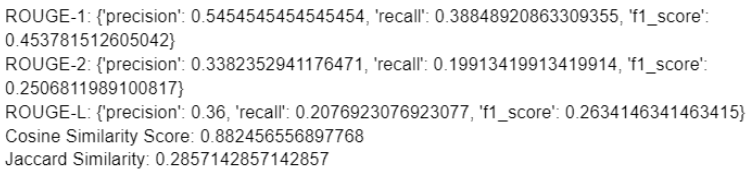

Figure 17: Experiment 5 - Basemodel Metrics

Figure 18: Experiment 5 - Basemodel Training Loss

6 Conclusion

In conclusion, this research provides valuable insights into the fine-tuning of models for legal document summarization, with a focus on the OpenAI GPT-3.5 Turbo model. The

key findings of our study underscore the significance of prompt design, hyperparameter selection, and model architecture/model variants in optimizing summarization outputs for legal texts.

We have successfully investigated the efficiency of the LLM model in summarizing lengthy documents, achieving our research objective. Although, the model is reliable giving promising accuracies and similarities according to all similarity metrics but still not reliable enough to blindly follow the summary without any domain expertise because legal matters are sensitive and hence require a human evaluation and comprehension who have a sound domain knowledge. The summaries are generated by Artificial Intelligence can save Legal expert's ample amount of time and resources, hence the cost of his working may get almost half, which might increase the flow of justice overall. It assist legal workflows but it should not be solely relied upon for the outcomes or for the understanding of lawsuits. It's important to inform users about the AI's nature. Human expert input is essential throughout the process. The dataset is a large diverse dataset of non-representative sample of all US civil lawsuits but only suitable for US as it contains data that follows their legal framework as they have their own style of understanding, writing, outcome which may not be applicable to countries that follow other principles in their legal frameworks. In future, we will train with several other countries data, to remove biasness.

Our research addresses the core argument of optimizing fine-tuning methodologies for legal document summarization using advanced Large language models. The decision to use a subset of data, carefully chosen to meet the requirements of instruct-finetuning, proved efficient in terms of time consumption and computational resources.

Our experiments with Short-Tiny and Long-Short Summary Models demonstrated varying degrees of success. The former showed promise but indicated room for improvement, while the latter, especially with an increased number of epochs, exhibited a higher potential for effective legal document summarization. We did further experimentations with different prompts revealed notable differences in performance. The more detailed and comprehensive prompt, specifying roles, background, and criteria, exhibited challenges related to ambiguity, overfitting, and potential model understanding issues. In contrast, a straightforward system-level prompt consistently outperformed, emphasizing clarity and effectiveness in capturing legal nuances.

Furthermore, Temperature settings were explored in the context of a Large Language Model (LLM), with lower values contributing to more consistent outputs, while higher values offered increased creativity at the expense of accuracy. Additionally, a comparison between detailed and straightforward prompts favored the latter, suggesting its superiority in capturing legal language nuances.

It is to be noted that the terms "High," "Moderate," and "Low" are relative within the context of the comparison between the two models. These observations are based on the provided ROUGE metrics, and the significance of these scores depends on the specific requirements and priorities of your summarization task and as per human evaluation.

In summary, our research provides valuable insights into the intricate process of fine-tuning models for legal document summarization. We recommend employing a straightforward system-level prompt with a temperature value ranging from 0 to 0.3 for optimal results. As we move forward, it is essential to consider dataset attributes, experiment with various prompts, and explore the dynamic interplay between temperature settings and prompt styles for further refinement and enhancement of summarization models in the legal domain.

Model	ROUGE-1 Precision	ROUGE-1 Recall	ROUGE-1 F1 Score	ROUGE-2 Precision	ROUGE-2 Recall	ROUGE-2 F1 Score	ROUGE-L Precision	ROUGE-L Recall	ROUGE-L F1 Score
Short-Tiny	High	High	High	Moderate	Moderate	Moderate	Low	Low	Low
Long-Short	Moderate	Moderate	Moderate	Low	Low	Low	High	High	High

Figure 19: Summary of Models

The effectiveness of fine-tuning models depends on various factors, such as type of data, craftiness of prompts, amount of data, complexity of task, overfitting risks, and model understanding and most importantly it depends on the human factor how well the summaries are interpreted by humans.

The detailed prompt, with specific instructions about roles and criteria, may introduce ambiguity and hinder model navigation. Overfitting to training data is a concern, especially if the detailed prompt is too specific. Model understanding may be a challenge with complex prompts, while a straightforward prompt offers clarity with minimal amount of data of around 50-100 well crafted instructions generates more better results. Data distribution misalignment and the challenge of generating concise summaries may impact the detailed prompt's performance.

7 Limitations and Future Work

7.0.1 Limitations

The current iteration of the ChatGPT model has a context length limited to 4096 tokens. A common guideline suggests that one token corresponds to approximately 4 characters of English text, or roughly 3/4th of a word. With this conversion, 100 tokens equate to around 75 words. Considering this, when dealing with long source documents containing 50,000 tokens, it becomes impractical for Large Language Models (LLMs) to process such extensive prompts efficiently.

During data preparation, 50,000 tokens are allocated for user content, and an additional 3000 tokens are reserved for completion. This brings the total to approximately 53,000 tokens while preparing instruction. Currently, no other model in the market provides a context window of this magnitude. Although newer models like gpt-3.5-turbo-16k and gpt-3.5-turbo-32k offer longer context windows of 16k and 32k tokens, respectively, they are limited to only using the models, and fine-tuning these models poses significant computational, financial, and time constraints, not to mention the associated carbon footprint from training.

Carbon Footprint Considerations: The carbon footprint of running ChatGPT is calculated to be 23.04 kgCO2 daily, constituting approximately 0.2 percent of a Dane's yearly carbon footprint. OpenAI is actively working on enhancing model and infrastructure efficiency to reduce environmental impact. However, it is crucial to recognize that the environmental impact extends beyond training, incorporating ongoing energy con-

sumption during deployment and usage.

7.0.2 Future Work

Fine-Tuning on High-Performing Open Source LLM Models: We aim to explore the fine-tuning of open-source LLM models such as Falcon70b, Falcon7b, LLama2, Mistral-7b, and Pegasus that demonstrate superior performance in critical thinking. Identifying models with enhanced capabilities in specific domains will be a key focus.

Jurisdictional Specificity and Global Applicability: The current dataset is extensive and diverse but is primarily focused on US civil lawsuits, making it non-representative for global legal systems. The dataset is tailored to the US legal framework, encompassing its unique styles of understanding, writing, and outcomes and legal ethic principles as they are more towards consequentialism approach. This specificity may limit the applicability of the model to countries like Germany, France who follow other legal principles like Deontological approach. Future iterations of the research aim to address this limitation by incorporating datasets from various countries to enhance inclusivity and mitigate potential biases.

Computational Demands and Environmental Impact: The high computational power required for fine-tuning on larger datasets raises environmental concerns, including CO2 emissions. Striking a balance between environmental impact and utility within a budget constraint, especially for academic purposes, remains a challenge. Despite budget and time constraints, we aspire to fine-tune on a larger dataset with a style mirroring that of seasoned legal experts, ensuring reliability by securing high end budget for this research purposes.

Extended Data and Prompt Engineering Experiments: Future work emphasis will be on training the model with more extended data and prompt engineering experiments during fine-tuning. Recognizing the pivotal role prompt engineering plays in model performance, we aim to refine and optimize prompts to enhance the overall model reasoning and mitigate failure modes while keeping enough data for the model to learn properly. Incorporating lessons from the field of prompt engineering, as outlined by OpenAI, will be instrumental in improving application performance. An understanding of best practices, including methods to enhance model reasoning and reduce the likelihood of hallucinations, is critical.

References

Agrawal, K. (2020). Legal case summarization: An application for text summarization, *2020 International conference on computer communication and informatics (ICCCI)*, IEEE, pp. 1–6.

Anand, D. and Wagh, R. (2022). Effective deep learning approaches for summarization of legal texts, *Journal of King Saud University-Computer and Information Sciences* **34**(5): 2141–2150.

Begum, N. and Goyal, A. (2021). Analysis of legal case document automated summarizer, *2021 6th International Conference on Signal Processing, Computing and Control (ISPCC)*, IEEE, pp. 533–538.

Carlotti, D. and Ferreira, J. E. (2022). Sumariação de textos como ferramenta de pesquisa empírica em direito, *Revista de Estudos Empíricos em Direito* **9**: 1–17.

Deroy, A., Ghosh, K. and Ghosh, S. (2023). How ready are pre-trained abstractive models and llms for legal case judgement summarization?, *arXiv preprint arXiv:2306.01248* .

Ghimire, A., Shrestha, R. and Edwards, J. (2023). Too legal; didn't read (tldr): Summarization of court opinions, *2023 Intermountain Engineering, Technology and Computing (IETC)*, IEEE, pp. 164–169.

Ghosh, S., Dutta, M. and Das, T. (2022). Indian legal text summarization: A text normalization-based approach, *2022 IEEE 19th India Council International Conference (INDICON)*, IEEE, pp. 1–4.

Jain, D., Borah, M. D. and Biswas, A. (2022). Improving kullback-leibler based legal document summarization using enhanced text representation, *2022 IEEE Silchar Subsection Conference (SILCON)*, IEEE, pp. 1–5.

Karkada, S. V. (2022). *Extractive Text Summarization of News Reports Leveraging Transfer Learning Contextual Embedders*, PhD thesis, Dublin, National College of Ireland.

Koniaris, M., Galanis, D., Giannini, E. and Tsanakas, P. (2023). Evaluation of automatic legal text summarization techniques for greek case law, *Information* **14**(4): 250.

Le Binh, D., Minh, H. L. N., Diem, Q. N. and Bao, D. T. N. (2023). An extraction-based approach for vietnamese legal text summarization, *2023 International Conference on System Science and Engineering (ICSSE)*, IEEE, pp. 61–66.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries, *Text summarization branches out*, pp. 74–81.

Manore, K. K. (2022). *Text Summarization of Customer Food Reviews Using Deep Learning Approach*, PhD thesis, Dublin, National College of Ireland.

Merchant, K. and Pande, Y. (2018). Nlp based latent semantic analysis for legal text summarization, *2018 international conference on advances in computing, communications and informatics (ICACCI)*, IEEE, pp. 1803–1807.

Mukherjee, A. (2022). *Developing Bengali Text Summarization with Transformer Base model*, PhD thesis, Dublin, National College of Ireland.

Muthiah, K. (2020). *Automatic Coherent and Concise Text Summarization using Natural Language Processing*, PhD thesis, Dublin, National College of Ireland.

Prasad, S. S. C. (2022). *Business Meeting Summary Generation using Natural Language Processing (NLP)*, PhD thesis, Dublin, National College of Ireland.

Qurashi, A. W., Holmes, V. and Johnson, A. P. (2020). Document processing: Methods for semantic text similarity analysis, *2020 International Conference on INnovations in Intelligent SysTems and Applications (INISTA)*, IEEE, pp. 1–6.

Rana, D. P., Mehta, R. G. et al. (2023). Research challenges for legal document summarization, *2023 IEEE World Conference on Applied Intelligence and Computing (AIC)*, IEEE, pp. 307–312.

Sarwar, A., Latif, S., Irfan, R., Ul-Hasan, A. and Shafait, F. (2022). Text summarization from judicial records using deep neural machines, *2022 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*, IEEE, pp. 1–6.

Sheik, R. and Nirmala, S. J. (2021). Deep learning techniques for legal text summarization, *2021 IEEE 8th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, IEEE, pp. 1–5.

Shen, Z., Lo, K., Yu, L., Dahlberg, N., Schlanger, M. and Downey, D. (2022). Multilexsum: Real-world summaries of civil rights lawsuits at multiple granularities, *Advances in Neural Information Processing Systems* **35**: 13158–13173.

Singh, R. (2023). *Text Summarization using Sequence to Sequence*, PhD thesis, Dublin, National College of Ireland.