Text Summarization from Judicial Records using Deep Neural Machines

By

Ayesha Sarwar

2018-NUST-MS-CS-08-275787

Supervisor

Dr. Faisal Shafait

Department of Computing

A thesis submitted in partial fulfillment of the requirements for the degree of Masters in Computer Science (MS CS)

In

School of Electrical Engineering and Computer Science,
National University of Sciences and Technology (NUST),
Islamabad, Pakistan.

(September 2021)

Approval

It is certified that the contents and form of the thesis entitled "Text Summarization from Judicial Records using Deep Neural Machines" submitted by AYESHA SARWAR have been found satisfactory for the requirement of the degree

Advisor : Prof. Dr. Faisal Shafait

Signature: Faisal

Date: 09-Nov-2021

Committee Member 1:Mr. Adnan Ul-Hasan

Signature:

Date: 09-Nov-2021

Committee Member 2:Dr. Seemab Latif

Signature:

Date: 10-Nov-2021

Committee Member 3:Dr. Rabia Irfan

Signature:

Date: 09-Nov-2021

THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS/MPhil thesis entitled "Text Summarization from Judicial Records using Deep Neural Machines" written by AYESHA SARWAR, (Registration No 00000275787), of SEECS has been vetted by the undersigned, found complete in all respects as per NUST Statutes/Regulations, is free of plagiarism, errors and mistakes and is accepted as partial fulfillment for award of MS/M Phil degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have also been incorporated in the said thesis.

Signature: Faisal

Name of Advisor: Prof. Dr. Faisal Shafait

Date: 09-Nov-2021

Signature (HOD): ____________________________

Date: ______________________________________

Signature (Dean/Principal): ___________________

Date: ______________________________________

Dedication

This work is wholeheartedly dedicated to my beloved parents who have been my source of inspiration and gave me strength during my challenging time, who continually provide their moral, spiritual, emotional, and financial support. To my sister Amna Sarwar, and my brother Fahad Sarwar who shared their word of advice and encouragement to complete this study.

And above all, the Almighty Allah who gave me the strength, peace of mind, skills and for giving me a healthy life. All of these, we offer to You.

Certificate of Originality

I hereby declare that this submission titled "Text Summarization from Judicial Records using Deep Neural Machines" is my own work. To the best of my knowledge it contains no materials previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any degree or diploma at NUST SEECS or at any other educational institute, except where due acknowledgement has been made in the thesis. Any contribution made to the research by others, with whom I have worked at NUST SEECS or elsewhere, is explicitly acknowledged in the thesis. I also declare that the intellectual content of this thesis is the product of my own work, except for the assistance from others in the project's design and conception or in style, presentation and linguistics, which has been acknowledged. I also verified the originality of contents through plagiarism software.

Student Name: AYESHA SARWAR

Student Signature: Ayesha

Acknowledgment

I begin by thanking the Almighty Allah for His countless blessings and benevolence throughout my research work that led me to complete the research successfully.

I would like to express genuine gratitude to my research supervisor, Dr. Faisal Shafait for giving me the opportunity to do research under his opulent banner and providing me with invaluable guidance throughout the process. I am immensely grateful to all my GEC members namely, Dr. Seemab Latif, Dr. Rabia Irfan, and especially Dr. Adnan Ul-Hasan who have always guided me and extended their helping hand and support for all my queries and inquiries.

I am beyond grateful to my parents, for their endless prayers, sacrifices and motivation. I would like to extend a special round of thanks to my siblings, Amna Sarwar and Fahad Sarwar for their encouragement.

In the end, I pay my immense gratitude to my friends namely, Maha Rasheed, Maria Rasheed, Nida Ahmed, Hiba Faisal, Ayesha Ashfaq and Ayesha Sadia Hashmi for their support and encouragement throughout this no-doubt challenging journey. Finally, my gratitude goes to everyone who has been involved in this journey to complete the research work whether directly or indirectly.

Table of Contents

List of Abbreviations

TS	Text Summarization
ETS	Extractive Text Summarization
ATS	Abstractive Text Summarization
RNN	Recurrent Neural Network
LSTM	Long Short Term Memory
LCS	Longest Common Subsequence
BERT	Bidirectional Encoder Representations from Transformers
BART	Bidirectional and Auto-Regressive Transformer
LED	Longformer Encoder-Decoder
SCP	Supreme Court of Pakistan
IHCP	Islamabad High Court of Pakistan

List of Figures

List of Tables

Abstract

The judiciary is the branch of the government whose task is the administration of justice. The courts are generating a large amount of data as legal proceedings. The legal documents are in the form of cases and their judgments. A judgment is a long, and detailed document. To prepare for a case, a lawyer has to read through hundreds of legal documents to find out the relevant judgments. In Pakistan, the ratio of cases that are registered every year and the judgments made is very high mainly due to the time it takes to prepare for a trial. Providing lawyers and judges with the summary of the relevant judgments will not only help them to get an overview without reading the whole judgment but also save a lot of their precious time, and hence more judgments can be made every year. Artificial Intelligence (AI) is finding its application in all domains of our lives. The use of AI techniques can also be helpful in courtrooms. Text Summarization is one of the applications of Natural Language Processing (NLP) which can be used to provide a brief overview of the judgment to both the lawyers and the judges. Transformer-based models in NLP, now-a-days, are a benchmark in solving sequence-to-sequence modelling problems. Therefore, they can be utilized to help legal domain experts save their time for writing judgment summaries in the real world. However, text summarization in legal documents differs from the regular text. The summarization task is dependent on the type of summary that is required. Moreover, the legal documents consist of tens of pages and hence more number of words. Therefore, existing pre-trained models on regular text cannot be helpful. Among other transformer-based models, Longformer has been introduced recently to deal with the long input sequence lengths up to $16,384$ tokens [2]. Training a model with such a configuration demands high computation power. Fine-tuning a pre-trained legal Longformer Encoder-Decoder (LED) on a downstream task showed better accuracy scores on the dataset.

Chapter 1

Introduction and Motivation

In this information age, the amount of data produced every day is truly astounding. Finding relevant data is a tedious and time-consuming process. The accessibility and availability of huge volumes of text data on the internet is a major challenge. The problem of information overload has arisen as data accessibility has increased. The amount of unstructured data accessible on the Internet is increasing which in turn is creating the need to develop new techniques for conveying content in a concise manner. Huge research efforts have been made to aid in the automatic processing of such online texts. Because human capacity for information consumption is limited, it is vital to retrieve only valuable and meaningful information from the massive amounts of unstructured data available. Therefore, extracting useful information from potentially massive amounts of text necessitates text summarization [3].

1.1 Introduction

Document summarization involves condensing the contents into a succinct form that captures only the most important ideas of a document. Automatic text summarization is one of the important areas of Natural Language Processing (NLP) and is designed to compress and render large text documents, allowing end users to quickly understand and read information. It requires compressing a long written document into a few words or paragraphs that conveys the main point [4]. The growing availability of information has necessitated the development of systems that can automatically summarise one or more documents, as the digitalization is embraced.

Every day, tremendous amounts of unstructured text are generated by legal systems all over the world. In Pakistan alone, lawyers, judges, and case workers process and evaluate millions of cases every year. These files can be rather lengthy, with hundreds of pages of dense legal material. According to the most recent Law and Justice Commission of Pakistan (LJCP) figures, the number of cases in the Supreme Court which are still pending is $38,539$ out of which $293,947$ cases are in five high courts, in the four provinces' subordinate courts and the federal capital.

As the volume of legal information continues to grow, appropriate efforts are required in the areas of automated processing and access to relevant forensic information. Many stakeholders, including lawyers, judges, and other professionals, value having access to up-to-date and relevant legal information from a single large site repository. Approximately 1.8m cases are pending in various courts of Pakistan. Generations of litigants suffer as a result of a backlog of cases. A well structured summary of judgment can provide the same insight and understanding as reading the long judgment. The information will be reproduced in a general form and this will save considerable time. There is a great deal of effort being made to create a manual copy of the case summary.

Reviewing long court judgments is a time-consuming task which involves human intervention and cognitive effort. When a new case is filed, the lawyers review a large number of previous court judgments to support their case. Lawyers and judges turn to a legal editor to evaluate a particular case. The court has a team of professional staff to evaluate the case. Lawyers rely on those summaries which are human-generated to find an effective set of discussions to answer lawsuit questions and support their arguments. If the requirements are immediate, there will be unnecessary delays and dependencies. Legal professionals may be able to better manage their workload if the review process is automated or simplified. Thus, automatic text summarization is a useful tool in helping them.

In the legal area, automatic summarization has a wide range of uses, ranging from making it easier for lawyers to navigate through a massive body of legal papers to quickly retrieving relevant judgments. Manually drafting case headnotes, synopsis, and summaries currently consumes a significant amount of time and effort. It is difficult to prepare a legal case summary that incorporates all relevant facts and precedents in order to convince the court to rule in the plaintiff's favor. Attorneys spend considerable time preparing legal briefs. In order to address these concerns, automatic text summarization is necessary [5].

1.2 What is Text Summarization?

According to [6], a summary refers to "a text that is created from one or more texts, conveys crucial information in the original text(s), and is no longer than half of the original text(s) and usually substantially less." The emphasis is on the most significant points, while minor details and instances are left out. A summary is a disassembled and reconstructed version of its source text that only contains the content's essence. Using a technology to automatically extract the most relevant information from a text and condensing it into a readable summary is commonly known as automatic summarizing. Significant research has been focused on the many forms of summaries, as well as the methods for creating and evaluating them.

In [7], authors divided summarization operations into three categories based on three aspects as follows:

- Input factors include length of text, and single vs. multi documents.
- Purpose factors define the user and the objective of text summarization.
- Output factors include flowing text or headed text, and so on.

Summarization can be performed on a single document or a collection of documents, and is referred to as single-document summarization or multi-document summarization. In [8], authors reviewed a number of document summarization types including single and multi-document summarization. The authors also discussed the issues faced due to automatic summarization, as well as addressed their assessment. Following sections describe the types of summarization in detail.

1.2.1 Single Document Text Summarization

Whenever a single document is long enough for a user to read in its entirety, its content can be summarized in which the relevant information is preserved. The topic of single document summarization is, therefore, of considerable research interest. Traditional extractive summarization methods focus on extracting important information from a document by recognizing sentence level content. Various methods have been applied for this purpose for sentence selection from documents [9].

1.2.2 Multi Document Text Summarization

Multi-document summarization, as the name implies, generates a summary from many documents published on the same subject. The summary assists the user in quickly familiarising themselves with the material included in a broader group of papers. In general, the summaries generated in this manner are both brief and thorough. Due to topic variety among a large number of papers, summarizing the multi document is more complicated and harder as compared to single document summarization [10].

1.3 Approaches of Text Summarization

There are two main approaches of how to summarize the text in Natural Language Processing (NLP). The two broad categories of approaches to the Text Summarization (TS) are extraction and abstraction. We discussed in detail the key consideration, both the approaches utilize.

1.3.1 Extractive Text Summarization (ETS)

Extractive summarization is a type of summary in which the material is fully extracted and the sentences of summary being extracted are phrases or words taken from the original text [11].

1.3.2 Abstractive Text Summarization (ATS)

In contrast to extractive summarization, abstractive summaries construct new sentences, sometimes known as paraphrases, which build summaries using those words not found in the original text. Abstractive text summaries create summaries by utilising natural language generation algorithms to paraphrase the document's main content. Because creating abstractive summaries necessitates significant Natural Language Processing (NLP), they are more complicated and challenging than extractive summaries. Abstractive summarization methods are often more difficult than extractive methods and more challenging regarding resources and computational complexity.

1.4 What is Legal Text Summarization?

Text Summarization can also help the legal community(judges, lawyers and petitioners) with its applications. However, the legal text differs from the general text. For example, general documents of the news genre have little or no structure. The summarization for general text does not focus on the structure of the text, but rather on the content words. The hierarchy of the structure, on the other hand, is critical in legal texts. A legal document follows a predefined structure. A legal judgment contains a header containing the information such as Case No, petition name, appellant name, etc., introduction, background, analysis, jurisdiction, and then conclusion. The inclusion of the same term at different levels of the hierarchy has different implications. The source of the ruling will determine the significance of the terms in the ruling (whether it is from a District Court, High Courts, or Supreme Court). In general, we can ignore references/citations when summarising materials, however this may not be possible in the case of legal writings. Moreover, the general documents of the news genre contain fewer paragraphs, whereas the legal documents or judgments are based on tens of pages. Most of the sequence-to-sequence neural models accept a standard fixed range of 512 input words token length. Therefore, the techniques used to summarize legal documents differ from the ones which are used to summarize the general text. Hence, legal text summarising requires unique attention and, as a result, necessitates a separate research from general text summarization.

1.5 Practical Challenges of Legal Text Summarization

The challenges in summarizing the legal document includes the preparation of a dataset and the characteristics of the legal text. These distinctions play a significant role in the summary of legal texts.

1.5.1 Characteristics of legal text

Legal text differ from the general text based on the following differences [12]:

1. Size:

   Legal documents are larger in size than documents in other domains.

2. Structure:

   They follow hierarchical structure.

3. Vocabulary:

   Legal texts use domain specific terminologies.

4. Ambiguity:

   The same term, phrase, or statement can have multiple meanings.

5. Citations:

   They indicate the main issues of the case.

The characteristics of legal text poses different challenges for both the approaches of TS such as vocabulary, in case of abstractive summarization, the generated summary could use the synonym words, which can have different meanings and implications in the given context. However, if trained on a larger dataset, the model can learn to adopt to the domain-specific vocabulary. In case of extractive summarization, the fluency and flow of the generated summary is a major concern, since it selects the top-ranked sentences from the source document to generate a summary. Abstractive summarization is harder than the extractive text summarization since it requires real-word knowledge, and semantic and contextual analysis [13]. However, abstractive summarization is better than the extractive summarization in a way that it is an approximate representation of the original document with human-generated language [14].

1.5.2 Dataset Preparation

Legal information scientists have conducted a significant study on automatic legal text summarization, and suggested solutions are based on a variety of methodologies. The majority of these methods for document segmentation rely on using labelled data or raw text's characteristics extraction for summary generation. The majority of legal text summarising approaches are presented as supervised learning algorithms when enough labelled data is available. Whereas, the labelled data produced in the legal field is scarce and expensive. One of the other major issue is the availability of a prepared dataset in the legal domain, such as FIRE-2014 (Information access in the legal domain) dataset [15]. It contains summaries of 1500 judgments in TREC format from the Supreme court of India. These judgments are from the period 1950-1989. But, the data is not available publicly and is encrypted with the password. The Center for Machine Learning and Intelligent Systems at the University of California, Irvine, has however, prepared a corpus of 4,000 legal cases (UCI Machine Learning Repository: Legal Case Report Dataset) [16] in the Federal Court of Australia (FCA) and is also available publicly [17]. It is designed for use in research involving automatic summarization and citation analysis. Catchphrases, citation sentences, citation classes, and citation catchphrases are collected for each document. As for summarization tasks, the catchphrases are different from the comprehensive summary. Therefore, it requires to prepare a full-fledged dataset from scratch to train the model on.

The challenges in preparing the dataset of legal domain involves the following factors:

1. Pre-processing:

   The input data needs to be mapped to the appropriate model format. The sequence-to-sequence models separate individual sentences, whereas the judgments and the headnotes are prepared by the courts according to their requirements paragraph wise in the documents format. The common way to differentiate individual sentences is using a period, but it can be used at many places as well such as in abbreviations, so it cannot be considered. There is no one standard library available to convert the paragraphs into sentences because of the use of the different language context. Therefore, it also requires much manual effort to prepare a full-fledged neat dataset of a considerable size from scratch for the first time.

2. Size:

   Deep Neural Networks are data hungry models. A supervised neural network model requires a significantly large amount of data for training purposes. Preparing a dataset from scratch for training purposes require both effort and time. Transfer Learning (TL) is particularly very useful approach when we have a smaller dataset. Hence, a model that has been pre-trained can be fine-tuned to achieve improved results using the dataset. In this case, the size of the dataset should also be considerable.

3. Domain Knowledge:

   In [18], it was proposed to use domain knowledge to automatically generate the labelled training data for legal text segmentation. However, a big challenge in dataset preparation of legal text summarization is the domain knowledge, to prepare the gold summaries, since it requires the help of professional lawyers.

1.6 Problem Statement

Reviewing long court judgments is a time-consuming task which involves human intervention and cognitive effort. Approximately 1.8m cases are pending in various courts of Pakistan. Generations of litigants suffer amid a backlog of cases [19]. A well structured summary of judgment can provide the same insight and understanding as reading the long judgment. This will not only provide a general version of the original information but save a lot of time. In the era of machine learning, deep neural networks have been used to summarize the documents. Therefore, it can be utilized to help legal domain experts save their time for writing judgment summaries in the real world. However, the legal text differs from the general text on the basis of the size, structure, vocabulary, citations, and the type of summary required. Therefore, techniques used to summarize legal documents differ from the standard ones.

A solution is required that considers both the length of the input sequence of long documents with limited resources, as well as an approach to compensate for the differences in characteristics of the legal text.

1.7 Solution Statement

The aim is to design and develop a system to summarize long legal documents with the implementation of deep neural network models while considering the existing limitations and requirements of the system. Transformer-based models in Natural Language Processing (NLP), now-a-days, are a benchmark in solving sequence-to-sequence modelling problems. Therefore, it can be utilized to help legal domain experts save their time for writing judgement summaries in the real world.

Transfer Learning (TL) is particularly very useful approach when we have a smaller dataset. However, Out-of-Vocabulary (OOV) is a common problem that arises in transfer learning. Therefore, existing pre-trained models on regular text cannot be helpful. A pre-trained Longformer Encoder-Decoder (LED) on a legal domain dataset can be fine-tuned by down-streaming it requiring less resources as well. This also addresses the OOV issue.

1.8 Key Contributions

There are two main key contributions i.e. dataset preparation and transfer-learning based transformer models for TS.

[20] and [21] are the two open source repositories available for the legal text summarization. The summaries are prepared by the professionals for every judgment. However, both the judgments and the headnotes need to be pre-processed and converted into individual sentences from paragraphs for the model's input. In Pakistan, the summaries for the legal judgments are also prepared for every judgment. We choose the judgments of Pakistan for our dataset to serve as a baseline for the future researchers to work on.

One of the key considerations in legal text summarization is the size of the input document. Most of the sequence-to-sequence neural models accept a standard fixed range of 512 input words token length. Transformer based models have been introduced recently to deal with the long input sequence lengths. Moreover, the available judgments are not enough to train a model from scratch. Whereas, deep neural network models are data hungry. Therefore, fine-tuning an existing pre-trained model with long input sequence length on legal documents is the best consideration. We propose transfer-learning based transformer models for legal text summarization. The results obtained through evaluation of this approach on the prepared dataset have shown an improved and satisfactory performance. This is the first time that deep neural networks have been used to summarize the legal documents of Pakistan. This work provides a baseline for future research involving our dataset, making it our second contribution.

1.9 Upcoming Chapters

Later part of the thesis document is organized in the following chapters.

The chapter 2 serves as a window into the notable work that has been done on text summarization generally and in the legal domain over the period of last decades. This section sets a research direction in this dissertation.

The chapter 3 discusses our proposed approach of text summarization for legal text in detail. It breaks down our approach into different modules and provides an insight into their technical details.

The chapter 4 presents the experiments and their results. It also provides the analysis of the results in detail with good and bad examples.

The last chapter provides the conclusive remarks and sheds light upon the future direction for the research community.

Chapter 2

Literature Review

The amount of textual material created on the internet is growing at an exponential rate so it has become critical to employ approaches in order to extract material that is most relevant to the user's information requirements. Humans may now use AI and machine learning algorithms to simplify a variety of activities, thanks to the advancements in AI and machine learning algorithms. However, automatic text summarization is a very challenging task. There are many key considerations such as sentence redundancy, sentence scoring, sentence ordering, etc., when summarizing documents, thereby, making the task more complex. Different approaches are being used for automatic summarization of text. Deep learning methods are used for summarising general text which have not been used for legal text commonly. In order to better understand and get clear picture of existing work done in literature, we breakdown the literature review into classical and deep learning based approaches for both general text and legal documents.

2.1 Techniques in Text Summarization

The approaches in text summarization can be broadly classified into two main categories i.e., supervised and unsupervised approaches. Following section firstly describe existing survey studies for summarization followed by state-of-the-art approaches that are compiled in the light of existing surveys as well as by performing literature review.

2.1.1 Supervised Approaches

With supervised approaches, the extractive text summarization ranks the individual sentences in a document based on the similarity with the provided summary documents. Whereas, the abstractive text summarization works using the language generation models and compares it with the sentence structure in the gold summaries. There exists many approaches for both types of summarization from traditional to deep neural machines. The introduction of Recurrent Neural Networks (RNNs) first enabled both extractive and abstractive approaches to advance rapidly and consistently. After the invention of the Transformer [22], the rate of progress accelerated, and this architecture has come to dominate state-of-the-art techniques. The following studies discuss the work done in the text summarization using supervised approaches.

Keneshloo et al. [23] proposed a transfer reinforcement learning based approach for a good generalization performance on different summarization datasets by training the model on a common vocabulary. The model described addresses the issue of common vocabulary between two datasets. The work demonstrates the generalization of the proposed model on unseen datasets. The pointer-generator model is, hence, the basis of the proposed model for knowledge transfer with added reward and self-critic policy gradient approach. The encoder and decoder units in the proposed TransferRL framework are shared between the source and target datasets. The model generates a sentence by sampling based on the output distribution, and learns from its own output distribution. The achieved results are presented for four datasets in text summarization. The datasets include are Newsroom, CNN/Daily Mail, DUC2003, and DUC2004. The encoder in the pointer-generator model uses LSTMs (Long Short Term Memory), which means that the output from the encoder is used by the decoder as it takes the last state as the input. However, transformer based models outperform the LSTM for the neural machine translation tasks.

BERT is a fairly recent (2018), state-of-the-art model developed by Google AI for a number of various NLP tasks. BERT took one of the most popular attention models, Transformer, and applied it in a bidirectional method causing the model to have a deeper sense of both context and flow of the word embeddings [24]. Bert has been applied for a series of NLP tasks which includes Question Answering techniques [25], [26], [27], Chatbots for multiple languages which are able to offer predictive texts as well as incorporate QA techniques to answer questions from a pre-determined corpora [28], [29], [30], [31] as well as for classification based on its abstract language modeling abilities for universal language representations [32], [33], [34], [35]. The BERTSumm is a transformer based model with its two variants, one for extractive and one for abstractive text summarization. BERTSummExt works by assigning a label to each sentence in the document whether it needs to be included in the final summary or not. BERTSummAbs, on the other hand, adopts language generation models in order to create summaries that contain novel words and phrases not appeared in the source text, but contain the same meaning and context. For BERTSumm, the input document first needs to be converted into proper sentences. The start of sentence contains a [CLS] token and end of sentence contains a [SEP] token. All the sentences in a document are then converted into an embedding vector for the representation. After obtaining the sentence representation or embedding vectors, document-level features are extracted using Transformer. For each sentence, the expected final score Yˆ is calculated. The loss is calculated of the model which is the binary classification entropy regarding Yˆ against standard label Y for every sentence. Training BERTSumm requires hours of training. The summary is generated as a set of sentences for every document. The final output contains standard metric values of ROUGE-1, ROUGE-2 and ROUGE-L scores. The BERTSummAbs [36] is ranked 1 in ATS on the news dataset, namely CNN/Daily Mail dataset and has a value of 41.72 for ROUGE-1. (CNN/Daily Mail Leaderboard). Whilst BERTSumm is a transformer based model, which considers BERT based contextual embeddings for summary generation, it works with small datasets.

Shi et al. [37] reviewed a wide range of models for abstractive text summarization. The article presents the review of the network structure, training approach besides the algorithms for summary generation were all examined. Although various articles have looked into abstractive summarization models, only a few have done so in depth [38]. Furthermore, most earlier studies, such as [39], [40] covered the methodologies until 2018, despite the fact that they were released in 2019 and 2020.

Abstractive summarization became a reality in the era of deep learning. Instead than employing actual terms from the reference materials, an abstractive summarization approach constructs a text. PEGASUS [41] is one of the more recent efforts on abstractive summarization. The majority of abstractive models are built around Transformers [22], which have a quadratic memory need in relation to the amount of input tokens whereas, transformer-based models can only handle 512 sub-word tokens, which is a significant constraint for long text summarization. Fortunately, some models, such as the Longformer [2], can convert the quadratic memory requirement to a linear memory requirement.

2.1.2 Unsupervised Approaches

Unsupervised approaches process the acquired results using heuristics or algorithms to finalize the insights. Statistical approaches in the unsupervised text summarization belong to either one of the three categories i.e., term-frequency [42], [43], [44], latent semantic indexing [45], [46], [47], [48], and graphical methods [49], [50], [51], [52], [53]. In term frequency, a higher score is assigned to the sentences that resemble the frequency of document terms. In latent semantic analysis approach, the sentence that best represents the latent concept is chosen. In graphical methods, a sentence to sentence similarity matrix is produced from a sentence term matrix, and each sentence gets a score. Following studies show the work done in unsupervised text summarization using deep learning techniques.

This paper [54] proposed an unsupervised deep learning model known as the Restricted Boltzmann Machine(RBM). The proposed approach works in three phases. In the first phase, all the required feature were extracted. A total of 9 features were extracted. These include number of thematic words, sentence position, sentence length, sentence position relative to paragraph, number of proper nouns, number of numerals, number of named entities, term frequency, and sentence to centroid similarity. A sentence-feature matrix is generated using feature values calculated for each sentence in the document. In the second phase, all the required features were enhanced. A RBM having one hidden layer and 9 perceptrons for every feature is trained to obtain an enhanced feature matrix. The final matrix values are used to generate a score for every sentence. In the last phase, the summary was generated. On the basis of the score of each sentence, the final summary is generated. This paper evaluated the results by calculating precision and recall values. The proposed approach achieved an average precision and recall values of 0.7 and 0.63 respectively on single-document factual reports from different domains such as news, sports, health, etc.

Alami et al. [55] used the concept of ensemble learning and word embeddings for word representation in unsupervised deep neural network techniques. The work has shown better results for automatic text summarization with three ensemble techniques. The work focuses on the word representation technique i.e., ensemble technique with Word2Vec representation performs better compared to those based on BOW (Bag-of-Words) approach. The datasets used were English emails and Arabic newspapers.

S Xu et al. [56] proposed an unsupervised technique for extractive summarization using pre-trained sentence-level self-attention transformers. This paper has performed a comparison between PACSUM [49] and proposed approaches on CNN/DailyMail and New York Times datasets. At the end, better results have also been obtained by combining both models. All documents, however, are truncated to 512 sub-word tokens due to the positional embedding of RoBERTa [57]. The F1 measures of ROUGE-1 score, ROUGE-2 score, and ROUGE-L score are 41.26, 18.18, 37.48 respectively.

Padmakumar et al. [58] showed how sentence embedding clustering can be utilised to achieve both extractive and abstractive text summarization. The summary is created by selecting a representative from each cluster of sentences. The extraction technique selects the sentence from the text whose embedding is closest to the cluster's centroid in terms of Euclidean distance. The abstractive strategy involves training a Recurrent Neural Network (RNN) with long short-term memory (LSTM) to decode embeddings into sentences and selecting a representative of each cluster. They consider the fact that the phrases which form a cluster in the vector space are likely to have similar meanings, retaining one sample from each cluster is sufficient to generate a summary. The results have been demonstrated and compared on the two datasets i.e., Tipster [59] and Opinosis [60].

Only five abstractive summarization models [61], [62], [63], [64], [65] were examined by [40]. The datasets and training procedures, as well as the architecture of multiple abstractive summarization models, were the focus of this study. There was no discussion on the key characteristic of the created summary of the various procedures and evaluation measures.

Abstractive summarization is harder than the extractive text summarization since it requires real-word knowledge, and semantic and contextual analysis [13]. However, abstractive summarization is better than the extractive summarization in a way that it is an approximate representation of the original document with human-generated language [14].

2.2 Techniques in Legal Text Summarization

Various techniques for summarizing legal text documents have been implemented. In the literature, work has been done on english judgments of Canada [66], India [67] and Australia [1]. The techniques include thematic segmentation, gravitational search algorithm, and K-means clustering. A brief summary of the literature review for the legal text summarization is shown in the table 2.1. The work in the domain of legal text summarization has started earlier but the progress is not at a very significant pace. One reason could be the non-availability of a standard dataset publicly.

2.2.1 Heuristics based Approaches

Heuristic based approaches include the pre-defined heuristic functions related to the calculated information. The following papers used labelled datasets for legal text summarization.

Kanapala et al. [67] developed a new summarization algorithm based on the gravitational search algorithm (GSA) using five objective functions on the Indian judgments from FIRE-2014 dataset. Based on different features associated with the criteria of sentence selection, such as length of the sentence, position of the sentence, keywords frequency, sentence similarity, every sentence in the document is binary classified as whether to include in the final summary or not. A gravitational search algorithm is implemented to optimize the summary of the document. F-measures for ROUGE-1 and ROUGE-2 are 0.4316 and 0.1749 respectively. The work mainly focuses on the features associated with the sentence relevancy, and the solution is optimized using GSA which can sometimes, trap into local optimum.

LETSUM, a legal text summarizer system [66], generates a table style summary based on four themes i.e. introduction, context, juridical analysis and conclusion). The system was trained on judgments of Canadian federal courts. Based on the sections identified in the judgment, a summary is generated for every section individually, and then combined. It works by assigning a score to each sentence in the judgement based on pre-defined heuristic functions. Other than evaluating the summaries using the standard evaluation metrices, human based evaluations are also performed for validation purposes. The labelled dataset is prepared by the professional lawyers, and consists of the associated label for every section of the judgment.

2.2.2 Neural Network Approaches

Texts containing legal issues in multiple dimensions posed difficulties for automatic summarization because of different styles of writing and the way the issues are discussed. For summarizing legal texts, asymmetric weighted graphs are used [68], where nodes represent sentences. The summary is only included in sentences with high node values. Each sentence inside a connected component of a document is represented by a connected graph. The result is a cohesive flow that promotes diversity. Fuzzy Analytic Hierarchical Process (FAHP) weighting for features is presented in [69] as a novel technique for producing an effective and efficient legal judgment summary. These summaries are subsequently evaluated by experts and are found to be more accurate than summaries produced by traditional approaches. The use of discriminant analysis to summarize Arabic texts from multiple documents [70] is proposed. According to [71], the clustering of legal judgments, according to topics, can be achieved using a hierarchical Latent Dirichlet Allocation (hLDA) strategy. In order to calculate hLDA and derive the summary of each document using the topics that are similar, similarity measures are used. A simplified method is presented in [72], where the importance scores are computed by adding up the TF-IDF scores from individual sentences and normalizing by the length of each sentence. In the same way, the section headings are treated differently, entity names, dates, and segments are also treated differently. There is an annotation method that combines different granularities of textual units to identify significant text [73]. They exploit the structure of legal text and identify semantically similar text fragments. For segmentation and annotation, these methods entirely rely on labelled data.

The framework utilizing deep auto-encoders along with phrase embeddings for extractive text summarization of single document is presented in this paper [74]. The method creates a summary based on three criteria: content relevance of sentence, position significance and novelty of sentence. The proposed model used an auto-encoder network for sentence content relevance, cosine similarity for sentence novelty, and a function for relative position of the sentence in the document for determining the sentence position relevancy. A fusion scheme is being utilized for joining the three proposed sentence features for the selection strategy. The Tor Illegal Documents Summarization (TIDSumm) dataset includes two sets of golden summaries of 100 documents from a website on the Tor network (The Onion Router). On the TIDSumm dataset, the scores for ROGUE-1, ROGUE-2, ROGUE-L, and ROGUE-SU4 are 58.8, 48.9, 49.3, and 45.9, respectively. On the TIDSumm dataset, this method achieves higher ROUGE scores.

A hybrid unsupervised method based on extractive summarization of single document is proposed by [1]. The automatic summaries of legal cases are created using k-means clustering and tf-idf (term frequency-inverse document frequency) word vectorizers. However, considering only the tf-idf could not be very efficient when it comes to large vocabularies, and does not consider the semantic similarities between the words. Three different steps are performed i.e., first it involves preprocessing, then clustering of similar sentences, and analyzing each cluster for top-ranked sentences. The paper used the Australian legal cases as a dataset. The F-Measure for ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-W are 27.88, 5.83, 33.5, and 11.38 respectively.

2.2.3 Deep Neural Network Approaches

There are techniques based on deep learning that have been proposed for summarization of legal texts [75], [76]. Legal decisions are often long, complex, and structured in some way. Legal trials can almost always be divided into two parts: the claim part and the realization part. The authors tested many models of BILSTM-CRF and demonstrated many ablations of these models. Two data sets were used in this survey. RRI records of Japan National Pension Law (JPL-RRE) and RRI records of Japan Civil Law. The first record contains the fragmented part of the sentence. Subordinate categories are used to mark these parts (topic part, background part, and subsequent parts). Therefore, it can be used as a collection with a unique name to represent RE components. Three employees manually entered the English translation of the Japanese Civil Code included in the English translation of the Japanese Civil Code. The logical part of the registry is divided into three parts: essential part, execution part and abnormal part. They are used to describe exceptions in legal notices.

This paper [18] prepared a labelled dataset from the judgments of Supreme Court of India. They calculated the similarity of each individual sentences with those in summaries, and then assigned labels to sentences in the document. The deep learning models i.e. Feed Forward Neural Network (FFNN) and Long Short Term Memory (LSTM) models have been applied on the generated labelled dataset to obtain extractive document summary. The best ROUGE-L score obtained is 38.2%.

Modern NLP is driven by transform-based models for solving sequence-to-sequence modeling problems. In transformer based machine learning technique, all of the BERT applications have been ventured into while keeping them specialized to specific domains. One of these specific domains is the legal domain that deals with legal or court documents as its corpus. The application of BERT to legal corpora has been termed as LEGAL-BERT [77], [78] which has been used in a large number of languages when it comes to legal NLP. However, researching the use of BERT in health and comprehension specific domains has been vast but there seems to be very little amount of research when discussion is set in the legal specific domain. [79] attributed this lack of demonstration on legal corpora due to the challenge of obtaining big legal datasets due to their confidential nature. Therefore, a handful of applications of BERT have been explored in legal environments including but not limited to ranking [80], classification [81], similarity scoring and information retrieval [82].

For producing legal document summaries, deep learning methods have rarely been applied, whereas, a high success rate has been achieved in recent years using deep learning for text summarization generally. The models for text summarization cover a range of architectures, from simple multilayer networks [45] to complex neural network architectures [2], [83], [84]. Since, one of the major differences in the characteristics of legal text with the general text is its length. The transformer based models such as BERT [35], BART [84] cannot be helpful if the length of the legal documents exceeds $1,024$ tokens, because of their limitation on the length of the input word embeddings. Longformer [2] has been recently introduced to deal with long input sequence lengths. However, their applications have not been explored in the field of legal text summarization as much. A brief summary of the literature review for the legal text summarization is shown in the table 2.1.

In Pakistan, there has been no work done in the field of legal text summarization. Z Nasar et al. [85] published a survey on different approaches of text summarization. This paper has discussed text summarization in general and then discussed the tools available for legal text summarization. We intend to provide a baseline with our proposed work in the field of legal text summarization.

Study	Dataset	Approach	Features	P	R	F	R1	R2	RL
MY Kim et al. (2012) [68]	House of Lords	Graph-based summarization	tf-idf, centroid similarity	31.7	30.7	31.2	-	-	-
RK Venkatesh et al. (2013) [71]	Indian legal judgments	hLDA	Similarity measures	62.3	60.7	61.4	-	-	-
Kanapala et al. (2019) [67]	FIRE-2014	Gravitational Search Algorithm (GSA)	tf-idf, Heuristic functions	-	-	-	43.16	17.49	-
D Anand et al. (2019) [18]	Indian judgments (1947-1993)	LSTM + Glove	Word and sentence embeddings	-	-	-	37.6	21.7	33.5
Farzindar et al. (2004) [66]	CanLII	Thematic Segmentation	tf-idf, Heuristic functions	-	-	-	57.50	31.38	45.18
S Polsley et al. (2016) [72]	Legal cases from Federal Court of Australia (FCA)	Feature Selection	tf-idf, domain knowledge	-	-	-	19.4	11.4	6.1
A Joshi et al. (2019) [74]	TIDSumm	Deep Neural Networks	Sentence novelty, content, and position relevance	-	-	-	58.8	48.9	49.3
V Pandya (2019) [1]	AustL-II	k-means clustering	tf-idf	-	-	-	27.88	5.83	33.5
V Tran et al. (2018) [86]	Legal cases from Federal Court of Australia (FCA)	CNN	Scoring and Selection	-	-	-	22.95	-	-
L Zhong et al. (2019) [87]	US Board of Veterans' Appeals	CNN	Type classification and masked decisions	-	-	-	23.3	8.2	-
I Chalkidis et al. (2014) [88]	Legal cases from Federal Court of Australia (FCA)	Knowledge Base	Rule-based feature selection	-	-	-	66.3	-	-

Table 2.1: Literature summary for Legal Text Summarization.

Chapter 3

Design and Methodology

This chapter describes the architecture of the transformer based models in the field of text summarization and their key differences. Based on the comparison analysis, we proposed the flowchart of our methodology, and the experimental setup used.

3.1 Architecture Analysis

Transformer based models include the models such as BERT, BART, and Longformer. The following sections describe their architecture in detail, and the models utilizing their architecture for summarization tasks.

3.1.1 BERT Architecture

Language understanding tasks usually require the use of pre-trained language models. In recent years, several models have been used to solve generation problems [89], [90]. When language models were first introduced, they employed both Recurrent Neural Networks (RNN) and Convolutional Neural Networks (CNN) for NLP tasks. Then, in 2017, Google introduced self-attention based Transformers for the very first time. Even though both RNNs(Recurrent Neural Networks) and CNNs (Convolutional Neural Networks) are competent, Transformers are still superior, since unlike RNNs and CNNs, it does not require a fixed sequence of data to be processed. Transformers are capable of processing data in any order, so they allow large-scale datasets to be trained that were previously not possible. In the past, language models could only read text sequentially, either from left to right, or vice versa but not simultaneously. It has recently become evident that trained language models [91], [92], [93], [24], [94] are a promising technology for improving virtually all aspects of natural language processing. These models extend the idea of embedded words to large-scale corpora of contextual representations. Therefore, pre-trained models like BERT were developed because their training was based on a large set of language data. BERT is an acronym for Bidirectional Encoding Representations Based on Transformers, in which each output element is linked to every input element as well as the weight based on the degree of connection between them. It is trained using a masked language modeling and a next-sentence prediction task on 3,300M−word dataset [24]. Figure 3.1 shows the general architecture of the BERTSumm text summarization model employing BERT based embeddings. A couple of tokens are added to the input text as a pre-processing step. [CLS] token is placed at the start of the text; this token will be used to aggregate information from the entire sequence(for example for classification task). Each sentence is marked with a token [SEP] as an indication of the end of a sentence. Tokens are then used to represent the modified text as $X = [w_1, w_2, , w_n]$. There are three types of embeddings associated with each token w_i: *token embeddings* denote the meaning of the token, *segmentation embeddings* indicate the semantic relationship between two sentences(for instance, during sentence-pair classification), and *position embeddings* indicate token positions within the text. Each of these embeddings is combined to produce a single input vector x_i and subsequently fed into a bidirectional Transformer.

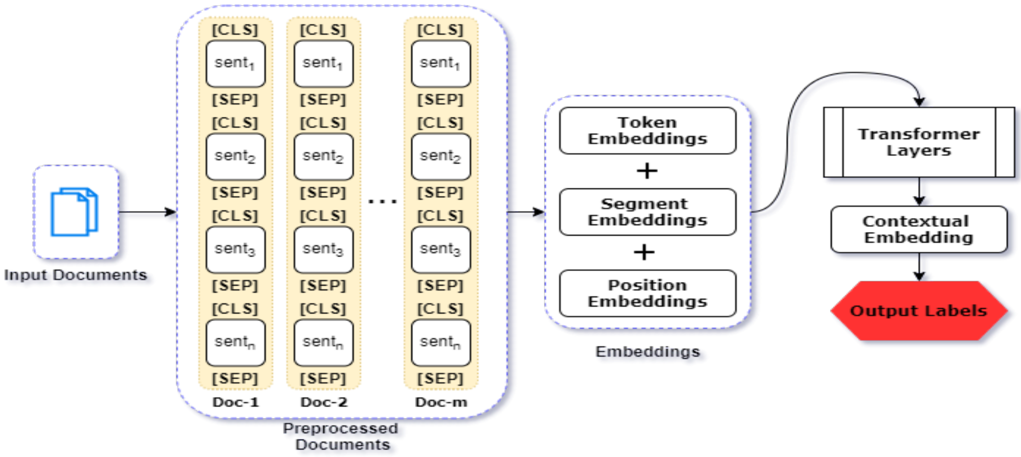


Figure 3.1: Architecture of the BERTSumm model.

3.1.2 BART Architecture

BART utilizes a sequence-to-sequence model for denoising auto-encoding that is applicable to a broad range of tasks [84]. BART follows the standard sequence-to-sequence Transformer architecture from [22] except that GeLUs activation functions [95] is implemented in place of ReLUs as an activation function and initialize parameters accordingly.

In this model, the text is encoded bidirectionally using bidirector encoder, and a decoder is applied that uses left-to-right autoregression. BART is viewed as an extension of BERT. There are several key differences with the BERT architecture compared to the Transformer sequence-to-sequence model: 1) the final hidden layer of the encoder is also cross-attended by each layer of the decoder; 2) BART doesn't include a feed-forward network before word prediction. BART has about 10% more parameters than BERT of the same size. Pretraining is divided into two stages: (1) corrupting the text with an arbitrary noise function, and (2) constructing the original text using a sequence-to-sequence model. This neural machine translation algorithm uses a Transformer-based architecture, though it may seem simple, but in fact, the BERT-based scheme generalizes several other recent pre-training schemes, including GPT and BERT. The architecture can be viewed as a "combination" of the BERT and GPT frameworks with an encoder-decoder model. In BERT, the goal is to predict missing samples using Masked Language Modelling and Next Sentence Prediction (NSP) using the Bidirectional Transformer. The same is true of GPT, where the model is autoregressive, and the goal is to predict the token's next position.

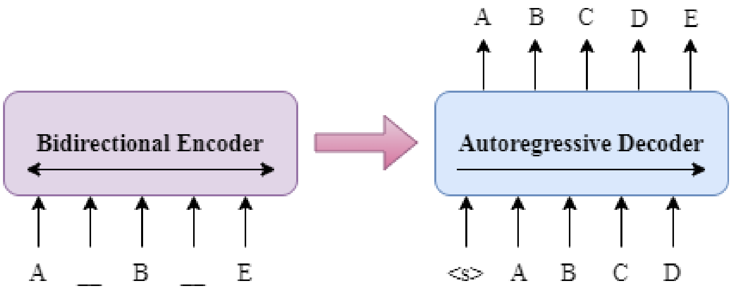


Figure 3.2: Bi-directional model employing encoder and auto-regressive decoder.

The example presented in the figure 3.2 shows the original document A B C D E. Before encoding, the spans [C D] is masked and an extra mask is attached to B, and corrupted document A _ B _ E is left and provided as an input to encoder. Decoding means reconstructing the original document, based on the encoder's output and previous uncorrupted tokens. The auto-regressive decoding of BART allows it to be fine-tuned for generating sequences, such as summarizing. The summarization process copies information from input yet controls it, which is similar to the denoising pre-training process. In this case, encoder inputs will be the input sequence while the decoder produces autoregressive outputs.

Among BART's benefits is its ability to be tuned to generate texts but it also works well for comprehension tasks. Along with providing a similar training experience to RoBERTa [96], GLUED [97] and SQUAD [98] are also designed to provide state-of-the-art results for a variety of abstractive dialogue tasks, question answering, and summarization tasks.Furthermore, BART allows for new approaches to fine-tuning. In [84], authors present a new method of machine translation that combines a BART model with several additional transformer layers. Through the propagation of BART, these layers are trained to convert foreign languages to noised English,and a result,it can be used as a pre-trained language model. The results show an increase in performance over the baseline of a strong back translation MT approach.

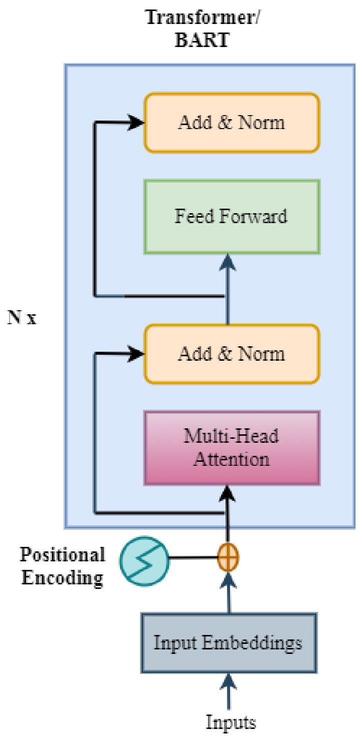


Figure 3.3: Architecture of the BART model.

A replicated ablation analysis that has been recently proposed examines the impact of a number of factors, including data and optimization parameters that have been shown to have a significant effect on training performance. According to analysis, BART demonstrates consistently strong performance across the entire list of tasks. The study compares two summarization datasets, CNN/DailyMail [65] and XSum [99], that have different characteristics to the state-of-the-art in summarization. CNN/DailyMail summaries resemble the original articles. In this case, abstractive models perform well, and even the baseline of the first three source sentences is competitive. However, BART outperforms conventional methods. XSum, on the other hand, is abstract and extractive models do not perform well. Qualitatively, samples are of high quality.

3.1.3 Longformer Architecture

Transformers have produced cutting-edge outcomes in a variety of natural language tasks both in generative language as well as discriminative language modelling. This achievement can be attributed in part to the network's self-attention module, which allows it to gather contextual data throughout the whole sequence. Although being powerful, it is not feasible for long sequences because when length increases, the memory and computational requirements also increase quadratically. To address this issue, a modified transformer design called Longformer is proposed whose self-attention grows linearly with length sequence that can be used to process long documents. This property makes it useful for tasks of natural language, for example, classification of long documents, co-reference resolution, question answering, etc.

Existing methods divide the large context and make them smaller sequences that fit inside the 512 token limit of pre-trained models like BERT. The loss of crucial information might arise from such partitioning. The Longformer, on the other hand, constructs the contextual representations of the complete context in which several layers of attention are used. The Longformer [2] replaces the self-attention layers with the sliding-window attention and offers an alternative to the quadratic memory problems of the Transformer. It results in linear complexity with respect to input length by reversing the dense matrix multiplication and replacing it with a sparse matrix multiplication. The primary purpose of local attention is to create contextual representations. Due to its design as a Transformer Encoder (TE), the Longformer cannot be used as-is to perform sequence-to-sequence tasks. The idea of replacing sliding window self-attention with dense self-attention layers also applies to other Transformer architectures, including the TED. Longformer Encoder Decoder (LED) is a variant of Longformer that is used for sequence-to-sequence tasks. Longformer's attention mechanism combines windowed local-context self-attention with task-motivated global attention that encodes the task's inductive bias. Both types of attention are necessary. LED works by substituting sliding-window self-attention for the BART [84] self-attention layers.

The O(n^2) memory requirement of vanilla self-attention can cause memory bottlenecks for long sequences. A solution such as the Longformer Encoder Decoder, LED, is offering sliding-window self-attention in combination with BART. The position embedding matrix of bart-base was simply replicated 16 times to be able to process 16K tokens [2]. LED can summarize documents that have a total length of $16,384$ tokens, four times longer than BART's maximum $1,024$ tokens. As a result, 2.5x longer sequences can fit on standard hardware. Two steps are required to convert BART to LED. First, BART's encoder's input length needs to be expanded for the model to process longer inputs. The weights are copied and then the matrices are concatenated to increase the positional embedding matrix. As BART allows a maximum input length of $1,022$ tokens, a resizeable positional embedding matrix could be created by copying and concatenating the positional embedding matrix. In the next step, a sliding-window model is designed to replace BART's layers of self-attention. By implementing the LED, the self-attention layers are only replaced in BART's encoder. The weights from the respective weight matrices from BART for each layer are then inserted into the query, key, and value matrices. Figure 3.4 shows the architecture of the LED model.

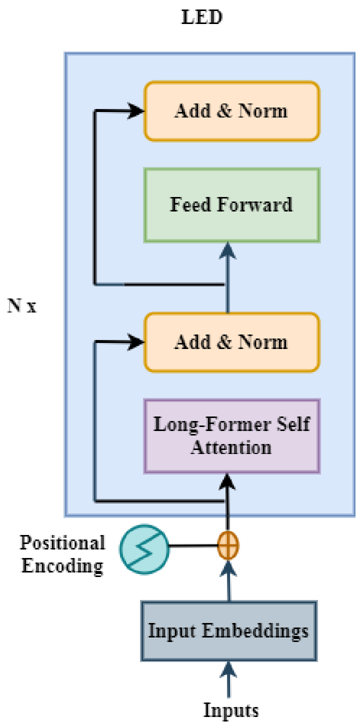


Figure 3.4: Architecture of the LED model.

3.2 BERT vs BART vs Longformer

This section explains the difference between the above explained primary models for text summarization and describes their design and implementation processes.

3.2.1 Architecture Comparison

A schematic comparison between BART, and Longformer is presented in Figure 3.5.

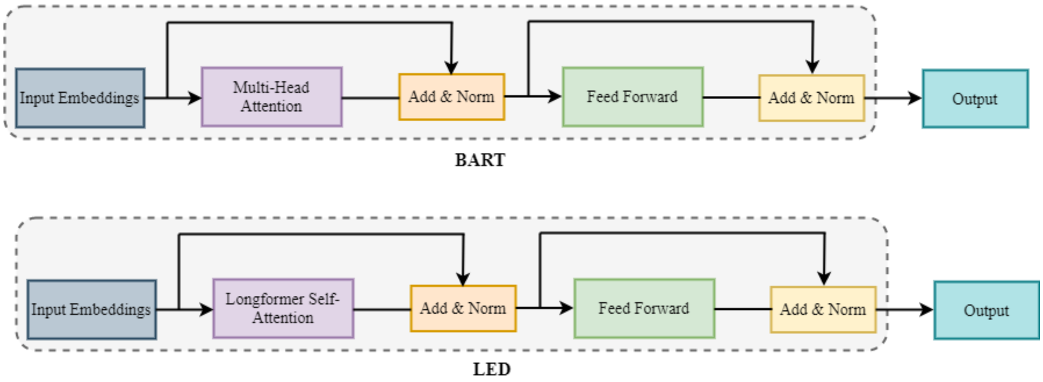


Figure 3.5: Schematic comparison of BART and Longformer architectures.

In the BART (Bidirectional and Auto-Regressive Transformers) architecture [84], the bidirectional encoder and auto-regressive decoder are combined. Grasping this architecture, pre-training can involve a greater range of noise-reducing transformations, including changes to the length of the input sequence. After experimenting with a variety of pre-training objectives, the authors find that shuffled order and the use of a single mask token to mask random sub-sequences of tokens results in the most useful pre-training. Through varying the length of sequences during training, a model can learn to deal with longer-range dependencies and the overall length of output. BART's pre-training regimen, therefore, becomes better suited for text summarization, resulting in breakthrough performance.

BART consists of an encoder and a decoder. It is trained on the same dataset as BERT, but is able to perform multiple tasks: token masking, token detection, text infilling, and sentence permutation. The authors [84] claim that BART is better than BERT for text generation because it has a decoder and is trained to do these tasks. Furthermore, the authors also released fine-tuned version of BART for use in other applications. In Seq2Seq architectures, summarization can be fine-tuned directly, without any new random initialization. Pre-training is also an effective method for predicting downstream actions. Both settings require the input document to be copied and modified. Seq2Seq-based models perform much better than the old, less-fancy ones in the CNN/Daily Mail abstractive summarization problem, and BART performed better.

A Transformer Encoder (TE) is designed as a Longformer by [2]. In their solution, the O(n^2) attention mechanism is replaced with sparse attention, which is linearly proportional to input length. It enables sparseness by employing "attention patterns" which indicate how positions serve other positions in the order. The Longformer uses sliding-window attention instead of self-attention layers, which offsets the quadratic complexity of the Transformer. This causes linear complexity as input length is increased, due to the replacement of dense matrix multiplications with sparse matrix multiplications. A TE is not suitable for seq2seq tasks, meaning the Longformer cannot be used as-is. A Longformer is constructed to include sliding window attention and replacing the self-attention layers. As a result of this idea, the sliding-window approach to self-attention can also be applied to other Transformer architectures such as the TED. As such, the LED builds upon BART substituting sliding-window self-attention for the self-attention layers.

3.2.2 Models Performance by Document Length

We noted that the self-attention memory consumes resources quadratically with input length. So, in this section, we will discuss the shortcomings of the above mentioned transformer models based on the input sequence length.

BERT has been very successful in both fine-tuning on specific tasks after pre-training and using the word embeddings in contrast to word2vec. The word embeddings can be used as features for other model, since a word will be represented contextually in contrast to word2vec. But, one of the limitations of BERT is its lack of ability to handle long input text sequences. BERT can only handle input sequence length up-to 512 tokens, and cuts off the rest of the input.

BART is performing well enough in a variety of tasks such as sequence classification and text generation tasks, but it can process sequences up to $1,022$ tokens, because its positional embeddings are $1,024$ width. By copying and concatenating the alignment matrix, BART can easily adapt to longer sequences. However, the costs associated with this then becomes impractical. The input document is truncated by taking the first $1,022$ tokens, so that BART may be applied to longer sequences.

Longformer has been proposed as a solution to this problem recently. It aims to address the limitation of the quadratic time complexity of the self-attention. A model based on BART is ostensibly used but instead of the usual “dense” self-attention layer, it uses a “sparse”, reduced-parameters self-attention layer to improve the computational complexity. In this way, these models are made more practical for longer documents by reducing spatial and time complexity from quadratic to linear. The replacement of the self-attention mechansism in any transformer based models by the sparse attention mechanism of the Longformer improved the performance in many NLP tasks.

In the Longformer, the maximum input token size is $16,384$ while BART allows for $1,024$. As a result, the Longformer can read documents without truncation, while BART algorithm cannot read more tokens without truncating them. Longformer outperforms BART for long documents and BART for short documents. As a result, standard Transformers such as BART truncate long documents around 1,000 tokens (approximately) when used for document summarization. BART’s cost increases when using long sequences. Using the Longformer will reduce the size of the truncated portions of the document, and therefore performs better. We will be utilizing Longformer Encoder Decoder (LED), a variant of Longformer, for supporting long document generative sequence-to-sequence tasks.

3.3 Proposed Methodology

As discussed above, Longformer Encoder Decoder (LED) base model can handle up-to 16K tokens. This model is best suited for question answering, comprehension and long summarization tasks. Furthermore, considering the issue of common vocabulary into account, a pre-trained model on a different domain dataset will not work. Our proposed methodology illustrates how can we handled the OOV words for the legal text summarization and performed a downstream task for input sequence length.

3.3.1 Common Vocabulary

A Longformer Encoder Decoder (led-base-16384) model [100] has been trained for the abstractive summarization of long documents on the legal domain. The sec-litigation-releases dataset [101], that contains around 2700 litigation releases and complaints from year 1995 to 2021, was used to train the legal-led-base-16384 model. These SEC releases detail the federal court civil lawsuits by the SEC (U.S. Securities and Exchange Commission). We will utilize the pre-trained legal-led-base-16384 model to fine-tune it on our dataset and avoid the Out-of-Vocabulary (OOV) words.

3.3.2 Fine-tune for Downstream Task

The input dataset has a median token length of $1,933$ with the 98%-ile token length being $6,101$. The output data has a median token length of 374 with the 90%-ile token length being 385. The legal Longformer Encoder Decoder (legal-led) base model with 16K tokens is fine-tuned on a downstream task for our prepared legal dataset with $8,192$ input tokens and 512 output tokens according to our data statistics. The datasamples are thus tokenized up-to the respective maximum lengths of $8,192$ and 512. We have performed fine-tuning on the pre-trained model up-to an input length of 8k tokens for summarization on the judgments from Supreme Court (SC) and Islamabad High Court(IHC). However, better performance can be achieved with fine-tuning the led-large-16384, on higher GPU. Thus, the maximum input length is set to $8,192$, and the maximum output length to 512 to ensure that the model can handle nearly all input tokens and generate enough output tokens.

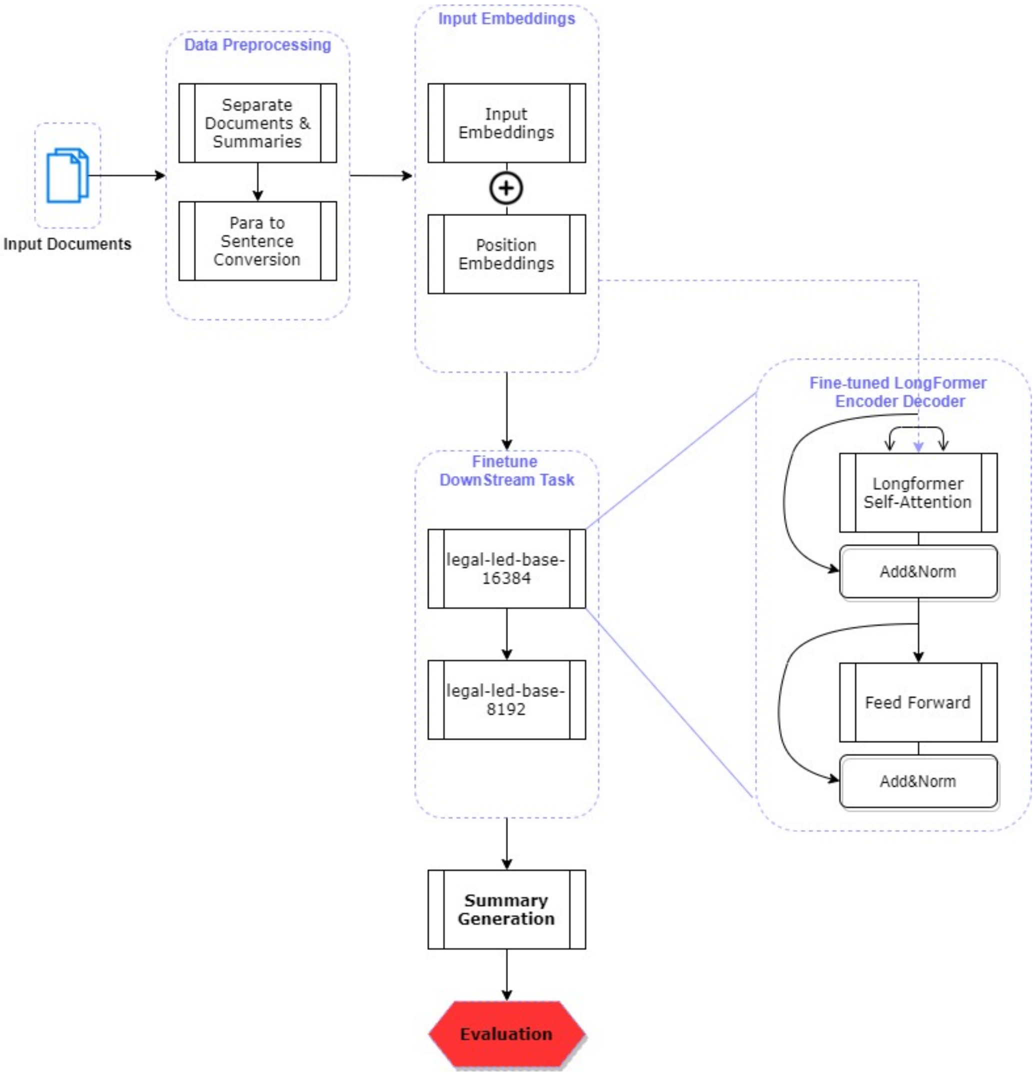


Figure 3.6: Architecture of the Proposed Methodology.

3.3.3 Model Architecture

Figure 3.6 shows the flowchart of our proposed methodology. This section describes in detail how we have adopted the deep learning model architecture for LED, and how we are integrating the evaluation feedback into our system.

The basic building blocks of Longformer are Transformers [22]. The transformer is an encoder-decoder architecture that utilizes the attention mechanism instead of Recurrent Neural Networks (RNN) for contextual training. The transformers hence, require a long-range memory that grows quadratically with input sequence length. Unlike other self-attention mechanisms, Longformer changes the self-attention method from full attention matrix to sliding window attention plus global attention, which increases linearly with the input sequence length thus improving memory efficiency. LED is a variant of Longformer used for summarization, and question answering tasks. The attention mask is utilized in the similar manner as in the original LED base model, and thus not calculated for the padded tokens.

The other training parameters are adjusted to train on a single GPU accordingly, such as batch-size of 2 with gradient-accumulation-steps to be 4, and beam-search to be 2. Beam search in NLP such as generation tasks help in generate the most likely sequences of words across the vocabulary of output words given their probability. Larger beam value results in the improved performance of the model, but at the cost of the speed at the decoding step. Therefore, an appropriate value should be selected accordingly. However, further improvements can be made in terms of time optimization depending upon the GPU RAM accordingly.

The model is evaluated on ROUGE, the standard metric used in automatic evaluation of machine translation. Both the system generated summary and the human generated summary are passed to the ROUGE library for comparison purposes. The ROUGE summarization evaluation package [102] includes four different ROUGE measures: ROUGE-N, ROUGE-L, ROUGE-W, and ROUGE-S for the evaluation. The same metric is utilized for evaluation during the training phase to improve the model performance.

We discussed in detail the process of our dataset preparation according to the model's input format, the experimental setup used, and a detailed analysis of the results of the model obtained on our dataset according to the standard evaluation metrices.

Chapter 4

Experiments, Results & Analysis

This chapter describes the details of the process for dataset acquisition, preparation, pre-processing, and details of the different experiments performed and their results. There is a detailed comparative analysis based on the produced results with the standard evaluation metrices followed by future guidance to pursue research on this work.

4.1 Dataset Preparation

Even though a great deal has been accomplished in terms of acquiring datasets, methods, and techniques, there are few papers that comprehensively summarize this field's current state of research. The judgments available in the legal domain for the task of text summarization are either not available pre-processed or publicly.

The dataset preparation involves mainly following three steps to be able to pass as an input to a model. These steps have been explained in detail, how they are followed to prepare our dataset.

4.1.1 Data Acquisition

Australia's law information service (AustL-II) is available free of charge online [20] in the PDF format. In its broadest sense, the Australian Legal Information Institute's mission is to improve access to justice through better information access. There are several primary collections maintained by AustL-II, including legislative and judicial decisions ("case law"). Sometimes these documents require a certain level of legal training or familiarity with the topic. The database maintained by AustL-II contains summaries for judgments in the PDF format. These judgments are from the period 2001-2008, and needs self pre-processing. [1] has utilized the same resource for performing their experiments.

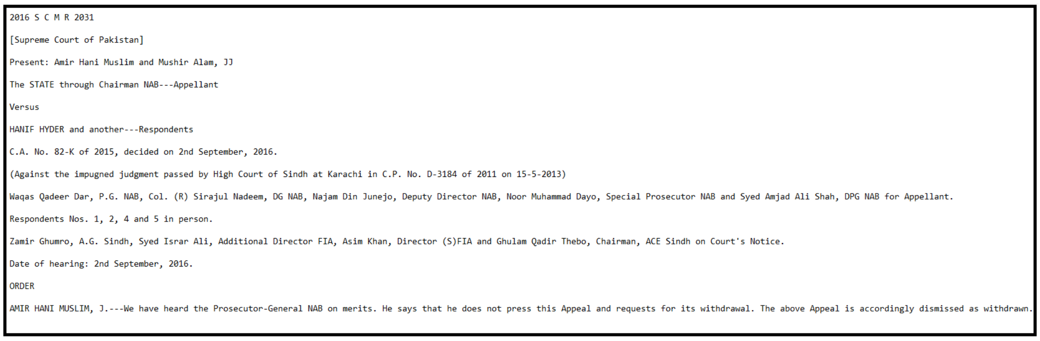
Supreme Court of Pakistan (SCP) maintains a corpus of both the judgments and the headnotes prepared by the professional lawyers in the text format. The judgments are available from the year 1991-Present. The mission of SCP is to empower their judges, and lawyers to provide them with the better information access to help them utilize their time better. However, the headnotes are prepared manually by the professional lawyers, and hence require a lot of effort and time. This work can be a step towards providing them with a baseline to automate the process of writing headnotes. As legal documents are of different kinds including civil, criminal judgments, revenue, written petition, etc. Subclasses of legal judgments can be found within a certain type of judgement. Criminal judgments, for example, are divided into subtypes including criminal revision, criminal appeal, criminal miscellaneous, and so on. The judgments used for this research are gathered from the Supreme Court of Pakistan (SCP) and Islamabad High Court of Pakistan (IHCP) for all available years.

All the judgments and their corresponding headnotes are downloaded manually as one judgment could be included in various journals. The documents were downloaded manually to avoid duplicates, and to ensure acquiring the desired file.

4.1.2 Data Pre-Processing

SCP maintains the headnotes of every passed judgment both within the judgment, and separately under several applicable legislative laws. The headnotes from the judgments need to be removed to separate the judgment text.

The next step is pre-processing. Since it is a sequence problem, the input is full single sentence. The pre-processing includes sentence segmentation. All the documents including judgments and summaries in the dataset are first converted into paragraphs. Then paragraphs are further segmented into sentences using a web service. Depending upon the nature of the document, the sentence separators vary a lot. The University of Malta provides an online tool [103] to convert the documents both into paragraphs and sentences which is also available as a web service. Considering the dot is not only used for sentence separation but in an abbreviation as well, manual verification is performed as a last step. Figure 4.1 shows the structure of the header in a judgment.



2016 S C M R 2031

[Supreme Court of Pakistan]

Present: Amir Hani Muslim and Mushir Alam, JJ

The STATE through Chairman NAB---Appellant

Versus

HANIF HYDER and another---Respondents

C.A. No. 82-K of 2015, decided on 2nd September, 2016.

(Against the impugned judgment passed by High Court of Sindh at Karachi in C.P. No. D-3184 of 2011 on 15-5-2013)

Waqas Qadeer Dar, P.G. NAB, Col. (R) Sirajul Nadeem, DG NAB, Najam Din Junejo, Deputy Director NAB, Noor Muhammad Dayo, Special Prosecutor NAB and Syed Amjad Ali Shah, DPG NAB for Appellant.

Respondents Nos. 1, 2, 4 and 5 in person.

Zamir Ghumro, A.G. Sindh, Syed Israr Ali, Additional Director FIA, Asim Khan, Director (S)FIA and Ghulam Qadir Thebo, Chairman, ACE Sindh on Court's Notice.

Date of hearing: 2nd September, 2016.

ORDER

AMIR HANI MUSLIM, J.---We have heard the Prosecutor-General NAB on merits. He says that he does not press this Appeal and requests for its withdrawal. The above Appeal is accordingly dismissed as withdrawn.

Figure 4.1: Civil Appeal Judgment from Supreme Court of Pakistan.

Once all the documents are passed through the first step of sentence conversion, regular expressions are used for domain specific sentence segmentation. The regular expressions are designed according to the document requirements. As documents are of different types and it is not necessary that a sentence could only end at dot, and the document may contain abbreviations as well. Therefore, it also involved manual effort to cross-check for any kind of errors.

4.1.3 Data Distribution

The pre-processed dataset comprises a total of 429 judgments. The dataset contains 94 judgments from the Supreme court and 335 from the Islamabad High Court of Pakistan (IHCP). The judgments from SCP and IHCP are then combined together. All the judgments contained words fewer than 8000.

The following table shows the distribution of the documents from both courts into training, validation, and testing splits individually.

Table 4.1: Dataset distribution for Train-Test split.

Court Name	Train	Valid	Test
Supreme Court	76	9	9
Islamabad High Court	269	33	33

The judgments were splitted into training, validation, and testing considering the 80-20% distribution for both courts separately to avoid overfitting.

4.2 Experimental Setup

This section explains the experimental setup employed with its characterization, and the selected fine-tuned hyper-parameters that minimized the pre-defined model loss function. We also describe the process of mapping the input data to the appropriate model format.

The input dataset has a median token length of $1,933$ with the 98%-ile token length being $6,101$. The output data has a median token length of 374 with the 90%-ile token length being 385. Therefore, we defined an input length of 8192 and an output length of 512 to make sure that the model can handle most inputs and can generate enough outputs. The minimum output length is set to 100, and maximum to 512 to make sure that the output length is within the specified range. Tokenizing data samples is carried out up to their respective maximum lengths of 8192 and 512. Tokens are generated according to a model specification, and we force the model to generate no more than 512.

To prevent out-of-memory errors, we trained on batch size of 2. To save memory, we used beam search with only two beams. Beam search in NLP such as generation tasks help in generate the most likely sequences of words across the vocabulary of output words given their probability. Larger beam value results in the improved performance of the model, but at the cost of the speed at the decoding step. Therefore, an appropriate value should be selected accordingly.

A number of other parameters have been set in order to improve the summary generation. According to the GPU RAM specifications, we converted gradient accumulation to a batch size of 8, by setting gradient accumulation steps to 4. Since the batch size is 2, and the gradient accumulation steps are 4, so the gradient accumulation batch size becomes 8. Besides the usual attention mask, LED can make use of the global attention mask to define which input tokens are being handled globally and which are being handled locally. In summarizing, we follow the recommendations of the paper [2] and only apply global attention to the very first token.

As part of the training process, the model should be evaluated on the most common summarization metric, ROUGE, in order for the model to improve during training as well. In addition to the gold labels, the ROUGE metrics also expects the generated output, called predictions since the rouge score is calculated based on the decoding of the tokens.

4.3 Performance Evaluation

As a crucial part of evaluating the system's performance, different performance measures are used. ROUGE scores are used to measure the accuracy of the sequence length problems. ROUGE is the de facto standard automatic evaluation metric for text summarization. ROUGE stands for Recall-Oriented Understudy for Gisting Evaluation. It compares the system generated summary to a set of gold standard summaries [104]. There are different variants of the rouge scores to evaluate the quality of the generated summaries and they are discussed below in detail.

ROUGE-N Score

ROUNE-N determines whether the simulated text matches the reference text in terms of n-grams.N-grams are simply groups of tokens. A uni-gram (or one-gram) consists of one word. Two consecutive words make up a bi-gram (2-gram).

Example

Original: “the quick brown fox jumps over the lazy dog”

1. Uni-grams:

   [’the’, ’quick’,’brown’, fox’, ’jumps’, ’over’, ’the’, ’lazy’, ’dog’]

2. Bi-grams:

   [’the quick’, ’quick brown’, ’brown fox’, ’fox jumps’, ’jumps over’, ’over the’, ’the lazy’, ’lazy dog’]

3. Tri-grams:

   [’the quick brown’, ’quick brown fox’, ’brown fox jumps’,’fox jumps over’, ’jumps over the’, ’over the lazy’, ’the lazy dog’]

ROUGE-N refers to the n-gram that we use.We would measure the match-rates between our model output and the reference based on ROUGE-1.A bi-gram would be used by ROUGE-2 while a tri-gram would be used by ROUGE-3.

Equation for ROUGE-N

$$= \frac{\sum_{S \in \{ReferenceSummaries\} \in S} \sum_{gram_n} Count_{match}(gram_n)}{\sum_{S \in \{ReferenceSummaries\} \in S} \sum_{gram_n} Count(gram_n)} \tag{4.1}$$

where,

The n-gram has a length of n. In a generated summary of n-grams, a count match (gram n) shows how many times an n-gram appears in the generated summary as well as gold/human generated summaries.

ROUGE-L Score

The ROUGE-L measure measures overlap based on the longest common subsequences (LCS) in the summaries. Using this method, two summaries X of length m, and Y of length n are compared, where X is a gold standard summary and Y is a generated summary.

Equation for ROUGE-L

$$R_{lcs} = \frac{LCS(X,Y)}{m} \tag{4.2}$$

$$P_{lcs} = \frac{LCS(X,Y)}{n} \tag{4.3}$$

$$F_{lcs} = \frac{(1+\beta^2)R_{lcs}P_{lcs}}{R_{lcs}+\beta^2 P_{lcs}} \tag{4.4}$$

So here ROUGE-L would be 1 if X=Y. In the absence of a common sub sequence, ROUGE-L will be equal to 0.

ROUGE-W Score

Despite taking only sub-sequences into consideration, ROUGE-L does not take into account whether those sub-sequences are consecutive. Using a weighted scheme will improve the LCS by favoring consecutive sub-sequences over those that are not consecutive. A shorter consecutive sub-sequence can be stored by simply storing its length. The sub-sequences that are consecutive and common are scored more than those that are not.

ROUGE-S Score

Using Skip-bi-grams, this statistic shows co-occurrences. Bi-grams with arbitrary gaps between them are known as Skip-bi-grams.

Equation for ROUGE-S

$$R_{skip2} = \frac{SKIP2(X,Y)}{C(m,2)} \tag{4.5}$$

$$P_{skip2} = \frac{SKIP2(X,Y)}{C(n,2)} \tag{4.6}$$

$$F_{skip2} = \frac{(1+\beta^2)R_{skip2}P_{skip2}}{R_{skip2}+\beta^2 P_{skip2}} \tag{4.7}$$

4.4 Results and Analysis

This section explains the results of our performed experiments, followed by the effect of change of some of the hyper-parameters. ROUGE-1, ROUGE-2, and ROUGE-L scores are considered and calculated for the system generated summaries. The comparison of these scores on our SCP dataset for the base pre-trained model on CNN/DM dataset and the model fine-tuned on the 2700 legal sec-litigation-releases is given in the table below. Both the base and the legal models are fine-tuned on a dowstream task with 8K tokens. The following results are achieved on the SCP dataset.

Table 4.2: Results comparison of base model and fine-tuned model on the SCP dataset.

Model	led-base-8192	legal-led-base-8192
rouge1	48.95	**53.11**
rouge1-recall	43.87	**48.25**
rouge1-precision	64.48	64.22
rouge2	28.72	**32.12**
rouge2-recall	25.27	**28.96**
rouge2-precision	40.13	39.67
rougeL	31.22	**34.09**
rougeL-recall	27.59	**30.79**
rougeL-precision	43.17	41.91

In Transfer Learning (TL), the problem occurs with the use of uncommon vocabulary. The technique used in transfer learning to avoid the unnecessary Out-of-Vocabulary (OOV) words is to use a common vocabulary between two datasets [23]. Although word2vec [105] and FastText [106] are trained using, for example, Wikipedia or other online corpora, the vocabulary that is used in these systems is finite. When training, words that aren't frequently used are often omitted. It is possible, therefore, that legal words specific to competition law aren't supported in the dictionary. With pre-trained word embeddings, the OOV words are usually replaced with the UNK token. There are also a number of words denoted (UNKnown word token) and all of them share the same vector. A corpus that is domain-specific is highly inefficient, as domain-specific words often have significant meaning. Considering that UNK tokens can replace most (meaning-carrying) words consequently, the model will be unable to learn much.

Since, the size of our prepared dataset is small, training the model from scratch will cause the issue of over-fitting. Moreover, fine-tuning a model with a dataset from different domain will lead to the issue of Out-of-Vocabulary (OOV) words. Therefore, the approach of transfer learning with a model pre-trained on a dataset with a similar domain will overcome this issue. The graph in the figure 4.2 shows the accuracy comparison of ROUGE-1, ROUGE-2 and ROUGE-L of our trained models.

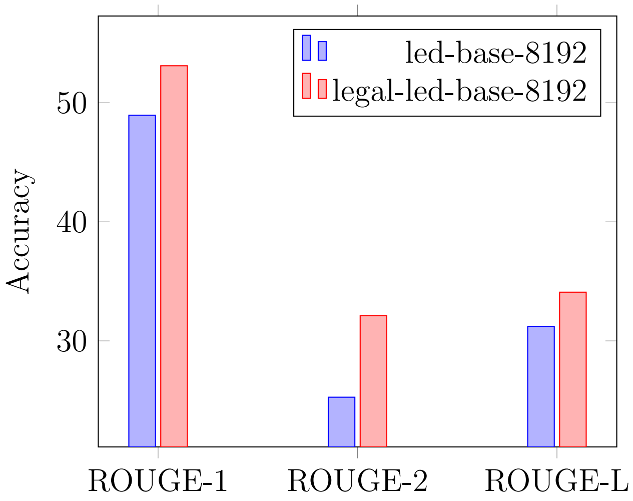
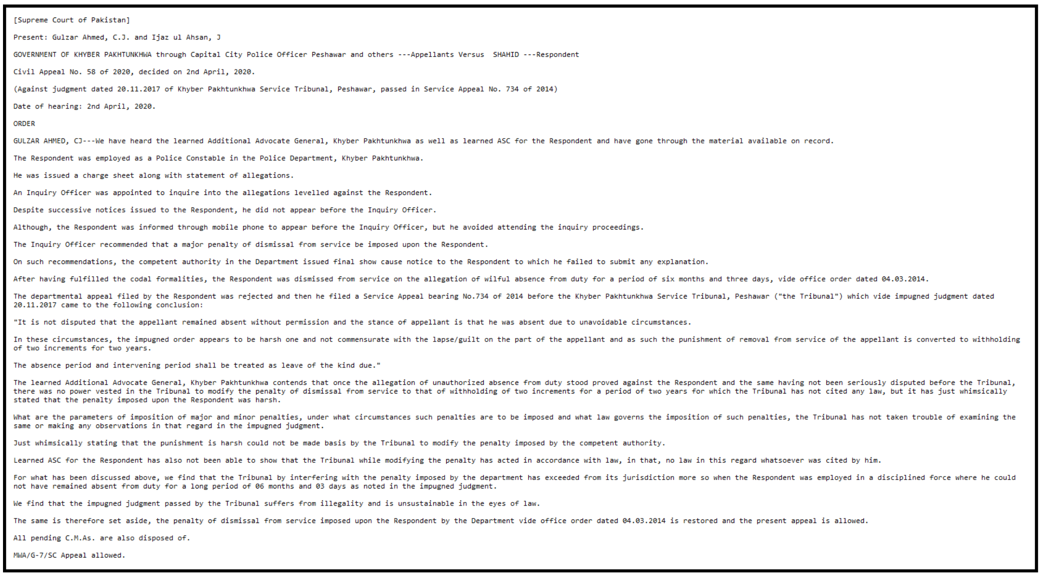


Figure 4.2: Comparison of ROUGE-1, ROUGE-2, and ROUGE-L for led-base-8192 and legal-led-base-8192 on SCP dataset.

The comparison in the graph above shows that the accuracy has improved significantly. The F-measure of all the rouge scores have improved. This shows that the model performance can be improved with transfer learning if a model is fine-tuned on a dataset with similar vocabulary or domain. ROUGE measures recall which means that how many words or n-grams from the reference summaries appeared in the system generated summaries. The increase in the ROUGE scores is an indication that there are many words from the human reference summaries in the system results. With the abstractive summarization, there cannot be a higher overlap between the words and phrases of a human-written summary and the machine-written summary, but we can get a sense of the overlap.

Figure 4.3 shows the snippet of one of the Civil Appeal judgments from Supreme Court of Pakistan. The following figures 4.4 and 4.5 show the human written and the system generated headnote of the same judgment.



[Supreme Court of Pakistan]

Present: Gulzar Ahmed, C.J. and Ijaz ul Ahsan, J

GOVERNMENT OF KHYBER PAKHTUNKHWA through Capital City Police Officer Peshawar and others ---Appellants Versus SHAHID ---Respondent

Civil Appeal No. 58 of 2020, decided on 2nd April, 2020.

(Against judgment dated 20.11.2017 of Khyber Pakhtunkhwa Service Tribunal, Peshawar, passed in Service Appeal No. 734 of 2014)

Date of hearing: 2nd April, 2020.

ORDER

GULZAR AHMED, CJ---We have heard the learned Additional Advocate General, Khyber Pakhtunkhwa as well as learned ASC for the Respondent and have gone through the material available on record.

The Respondent was employed as a Police Constable in the Police Department, Khyber Pakhtunkhwa.

He was issued a charge sheet along with statement of allegations.

An Inquiry Officer was appointed to inquire into the allegations levelled against the Respondent.

Despite successive notices issued to the Respondent, he did not appear before the Inquiry Officer.

Although, the Respondent was informed through mobile phone to appear before the Inquiry Officer, but he avoided attending the inquiry proceedings.

The Inquiry Officer recommended that a major penalty of dismissal from service be imposed upon the Respondent.

On such recommendations, the competent authority in the Department issued final show cause notice to the Respondent to which he failed to submit any explanation.

After having fulfilled the codal formalities, the Respondent was dismissed from service on the allegation of wilful absence from duty for a period of six months and three days, vide office order dated 04.03.2014.

The departmental appeal filed by the Respondent was rejected and then he filed a Service Appeal bearing No.734 of 2014 before the Khyber Pakhtunkhwa Service Tribunal, Peshawar ("the Tribunal") which vide impugned judgment dated 20.11.2017 came to the following conclusion:

"It is not disputed that the appellant remained absent without permission and the stance of appellant is that he was absent due to unavoidable circumstances.

In these circumstances, the impugned order appears to be harsh one and not commensurate with the lapse/guilt on the part of the appellant and as such the punishment of removal from service of the appellant is converted to withholding of two increments for two years.

The absence period and intervening period shall be treated as leave of the kind due."

The learned Additional Advocate General, Khyber Pakhtunkhwa contends that once the allegation of unauthorized absence from duty stood proved against the Respondent and the same having not been seriously disputed before the Tribunal, there was no power vested in the Tribunal to modify the penalty of dismissal from service to that of withholding of two increments for a period of two years for which the Tribunal has not cited any law, but it has just whimsically stated that the penalty imposed upon the Respondent was harsh.

What are the parameters of imposition of major and minor penalties, under what circumstances such penalties are to be imposed and what law governs the imposition of such penalties, the Tribunal has not taken trouble of examining the same or making any observations in that regard in the impugned judgment.

Just whimsically stating that the punishment is harsh could not be made basis by the Tribunal to modify the penalty imposed by the competent authority.

Learned ASC for the Respondent has also not been able to show that the Tribunal while modifying the penalty has acted in accordance with law, in that, no law in this regard whatsoever was cited by him.

For what has been discussed above, we find that the Tribunal by interfering with the penalty imposed by the department has exceeded from its jurisdiction more so when the Respondent was employed in a disciplined force where he could not have remained absent from duty for a long period of 06 months and 03 days as noted in the impugned judgment.

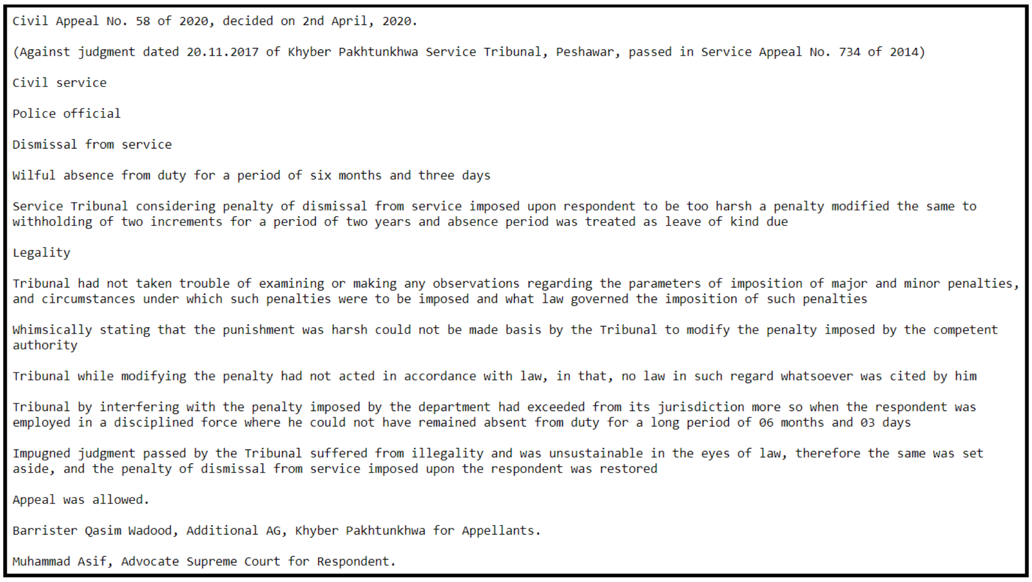
We find that the impugned judgment passed by the Tribunal suffers from illegality and is unsustainable in the eyes of law.

The same is therefore set aside, the penalty of dismissal from service imposed upon the Respondent by the Department vide office order dated 04.03.2014 is restored and the present appeal is allowed.

All pending C.M.As. are also disposed of.

MWA/G-7/SC Appeal allowed.

Figure 4.3: Example Civil Appeal Judgment from Supreme Court of Pakistan.



Civil Appeal No. 58 of 2020, decided on 2nd April, 2020.

(Against judgment dated 20.11.2017 of Khyber Pakhtunkhwa Service Tribunal, Peshawar, passed in Service Appeal No. 734 of 2014)

Civil service

Police official

Dismissal from service

Wilful absence from duty for a period of six months and three days

Service Tribunal considering penalty of dismissal from service imposed upon respondent to be too harsh a penalty modified the same to withholding of two increments for a period of two years and absence period was treated as leave of kind due

Legality

Tribunal had not taken trouble of examining or making any observations regarding the parameters of imposition of major and minor penalties, and circumstances under which such penalties were to be imposed and what law governed the imposition of such penalties

Whimsically stating that the punishment was harsh could not be made basis by the Tribunal to modify the penalty imposed by the competent authority

Tribunal while modifying the penalty had not acted in accordance with law, in that, no law in such regard whatsoever was cited by him

Tribunal by interfering with the penalty imposed by the department had exceeded from its jurisdiction more so when the respondent was employed in a disciplined force where he could not have remained absent from duty for a long period of 06 months and 03 days

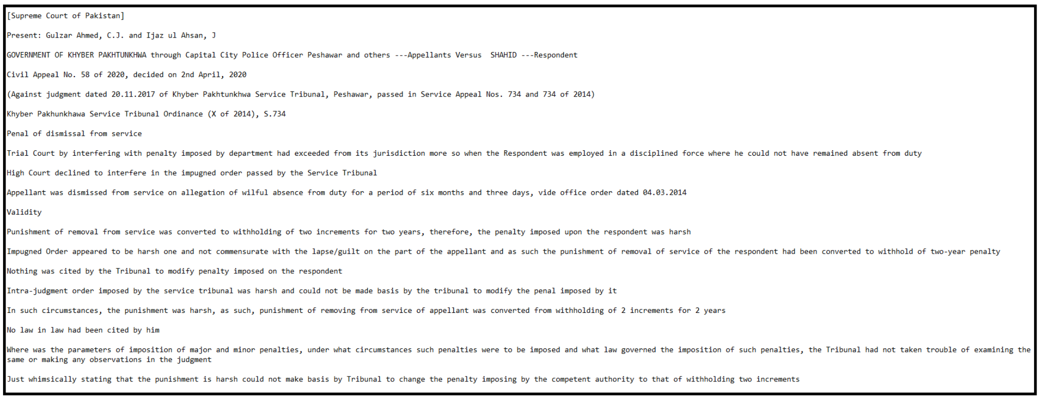
Impugned judgment passed by the Tribunal suffered from illegality and was unsustainable in the eyes of law, therefore the same was set aside, and the penalty of dismissal from service imposed upon the respondent was restored

Appeal was allowed.

Barrister Qasim Wadood, Additional AG, Khyber Pakhtunkhwa for Appellants.

Muhammad Asif, Advocate Supreme Court for Respondent.

Figure 4.4: Reference Headnote of the Civil Appeal Judgment from Supreme Court of Pakistan.



[Supreme Court of Pakistan]

Present: Gulzar Ahmed, C.J. and Ijaz ul Ahsan, J

GOVERNMENT OF KHYBER PAKHTUNKHWA through Capital City Police Officer Peshawar and others ---Appellants Versus SHAHID ---Respondent

Civil Appeal No. 58 of 2020, decided on 2nd April, 2020

(Against judgment dated 20.11.2017 of Khyber Pakhtunkhwa Service Tribunal, Peshawar, passed in Service Appeal Nos. 734 and 734 of 2014)

Khyber Pakhunkhawa Service Tribunal Ordinance (X of 2014), S.734

Penal of dismissal from service

Trial Court by interfering with penalty imposed by department had exceeded from its jurisdiction more so when the Respondent was employed in a disciplined force where he could not have remained absent from duty

High Court declined to interfere in the impugned order passed by the Service Tribunal

Appellant was dismissed from service on allegation of wilful absence from duty for a period of six months and three days, vide office order dated 04.03.2014

Validity

Punishment of removal from service was converted to withholding of two increments for two years, therefore, the penalty imposed upon the respondent was harsh

Impugned Order appeared to be harsh one and not commensurate with the lapse/guilt on the part of the appellant and as such the punishment of removal of service of the respondent had been converted to withhold of two-year penalty

Nothing was cited by the Tribunal to modify penalty imposed on the respondent

Intra-judgment order imposed by the service tribunal was harsh and could not be made basis by the tribunal to modify the penal imposed by it

In such circumstances, the punishment was harsh, as such, punishment of removing from service of appellant was converted from withholding of 2 increments for 2 years

No law in law had been cited by him

Where was the parameters of imposition of major and minor penalties, under what circumstances such penalties were to be imposed and what law governed the imposition of such penalties, the Tribunal had not taken trouble of examining the same or making any observations in the judgment

Just whimsically stating that the punishment is harsh could not make basis by Tribunal to change the penalty imposing by the competent authority to that of withholding two increments

Figure 4.5: System generated Headnote of Civil Appeal Judgment from Supreme Court of Pakistan.

Evaluating summaries is subjective in nature. The relevance and utility of each sentence in the summary varies depending on who is analysing them. Whereas evaluating such a large number of summaries manually is not feasible.

Therefore, in terms of quantitative analysis, different variants of the ROUGE scores provide a good insight of the quality of the candidate summaries being produced. ROUGE-1 and ROUGE-2 determine the informativeness of the generated summary. The learned positional embeddings with the attention mechanism helps the model learn the positional context and hence selecting the appropriate sentences from every part of the document. For the qualitative analysis, the ROUGE-L score is determinant for the fluency of a candidate summary [96]. Since, ROUGE-L takes longest common sub-sequence of the candidate summary from the gold summary into account which is considered as a fluency metric. We can determine the quality of the summary generated by the legal-led-base-model-16384 for the SCP dataset from the comparison of figures 4.4 and 4.5. The summary produced is representative of the required format, and fluent in its language.

The comparison of the ROUGE scores on the AustL-II judgments for the model fine-tuned on the legal sec-litigation-releases and the proposed model in [1] are given in the table below.

Table 4.3: Results comparison of our fine-tuned model and proposed model in [1] for AustL-II judgments.

Model	proposed-methodology-in-[1]	legal-led-base-8192
rouge1	27.88	**37.97**
rouge1-recall	28.16	**28.61**
rouge1-precision	27.62	**73.75**
rouge2	5.83	**20.04**
rouge2-recall	5.88	**14.86**
rouge2-precision	5.77	**41.33**
rougeL	**33.5**	23.49
rougeL-recall	**33.78**	17.48
rougeL-precision	33.24	**48.59**

The comparison in the above table shows that the accuracy has improved significantly for the judgments from AustL-II. The proposed methodology in [1] uses unsupervised approach. They utilize the approach of the k-means clustering algorithm for gathering the similar sentences under one cluster. Since, the extractive summarization is considered as a classification problem, all the sentences are ranked scores using tf-idf and the sentences having a higher score are selected for generating the summary with required number of sentences.

The reason in the the difference of the ROUGE-N scores is because of the reason that the methodology proposed in [1] are the sentences from the judgments, whereas the headnotes are generated in an abstractive way by the professional lawyers. The headnotes are not the original sentences from the judgment document. The reason in the the difference of the ROUGE-L scores is because of the reason that the longest common sub-sequence with the abstractive summarization is shorter than the extractive summarization. But, the increase in ROUGE-2 shows that the legal-led-base-model is using the same words in the summary generation task.

The following graph shows the accuracy comparison of ROUGE-1, ROUGE-2 and ROUGE-L of our proposed trained model and the model in [1] on the AustL-II judgments.

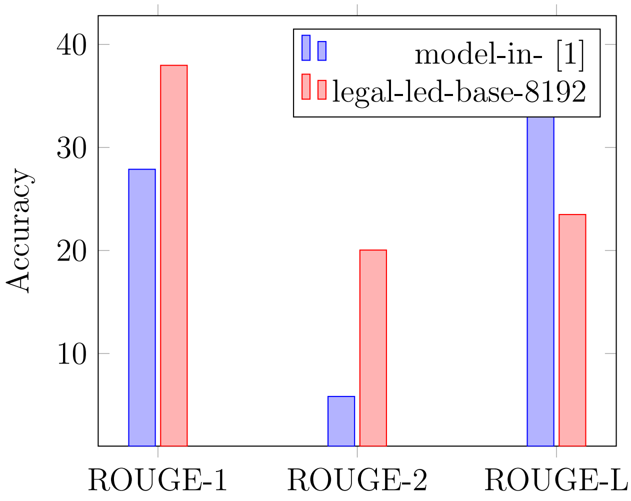
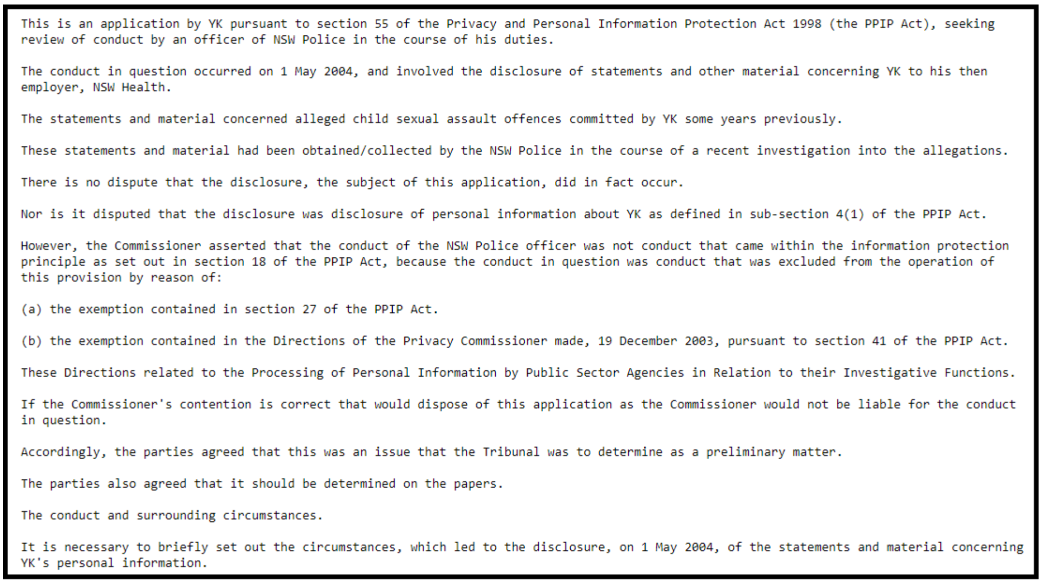


Figure 4.6: Comparison of ROUGE-1, ROUGE-2, and ROUGE-L for led-base-8192 and legal-led-base-8192 on AustL-II judgments.

Figure 4.7, 4.8 and 4.9 shows the snippet of one of the judgment from AustL-II and the associated human written and system generated headnote.



This is an application by YK pursuant to section 55 of the Privacy and Personal Information Protection Act 1998 (the PPIP Act), seeking review of conduct by an officer of NSW Police in the course of his duties.

The conduct in question occurred on 1 May 2004, and involved the disclosure of statements and other material concerning YK to his then employer, NSW Health.

The statements and material concerned alleged child sexual assault offences committed by YK some years previously.

These statements and material had been obtained/collected by the NSW Police in the course of a recent investigation into the allegations.

There is no dispute that the disclosure, the subject of this application, did in fact occur.

Nor is it disputed that the disclosure was disclosure of personal information about YK as defined in sub-section 4(1) of the PPIP Act.

However, the Commissioner asserted that the conduct of the NSW Police officer was not conduct that came within the information protection principle as set out in section 18 of the PPIP Act, because the conduct in question was conduct that was excluded from the operation of this provision by reason of:

(a) the exemption contained in section 27 of the PPIP Act.

(b) the exemption contained in the Directions of the Privacy Commissioner made, 19 December 2003, pursuant to section 41 of the PPIP Act.

These Directions related to the Processing of Personal Information by Public Sector Agencies in Relation to their Investigative Functions.

If the Commissioner's contention is correct that would dispose of this application as the Commissioner would not be liable for the conduct in question.

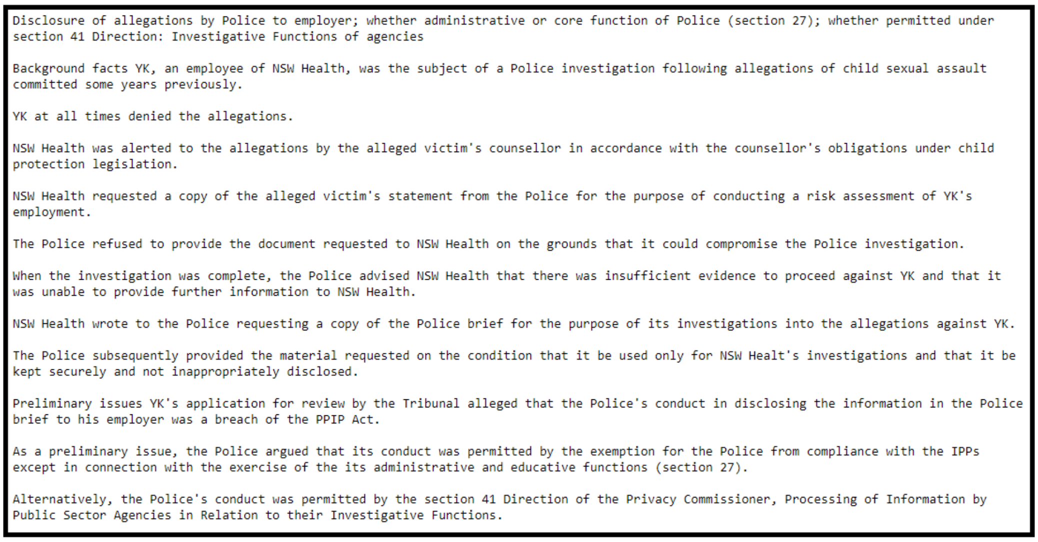
Accordingly, the parties agreed that this was an issue that the Tribunal was to determine as a preliminary matter.

The parties also agreed that it should be determined on the papers.

The conduct and surrounding circumstances.

It is necessary to briefly set out the circumstances, which led to the disclosure, on 1 May 2004, of the statements and material concerning YK's personal information.

Figure 4.7: Example Judgment from AustL-II.



Disclosure of allegations by Police to employer; whether administrative or core function of Police (section 27); whether permitted under section 41 Direction: Investigative Functions of agencies

Background facts YK, an employee of NSW Health, was the subject of a Police investigation following allegations of child sexual assault committed some years previously.

YK at all times denied the allegations.

NSW Health was alerted to the allegations by the alleged victim's counsellor in accordance with the counsellor's obligations under child protection legislation.

NSW Health requested a copy of the alleged victim's statement from the Police for the purpose of conducting a risk assessment of YK's employment.

The Police refused to provide the document requested to NSW Health on the grounds that it could compromise the Police investigation.

When the investigation was complete, the Police advised NSW Health that there was insufficient evidence to proceed against YK and that it was unable to provide further information to NSW Health.

NSW Health wrote to the Police requesting a copy of the Police brief for the purpose of its investigations into the allegations against YK.

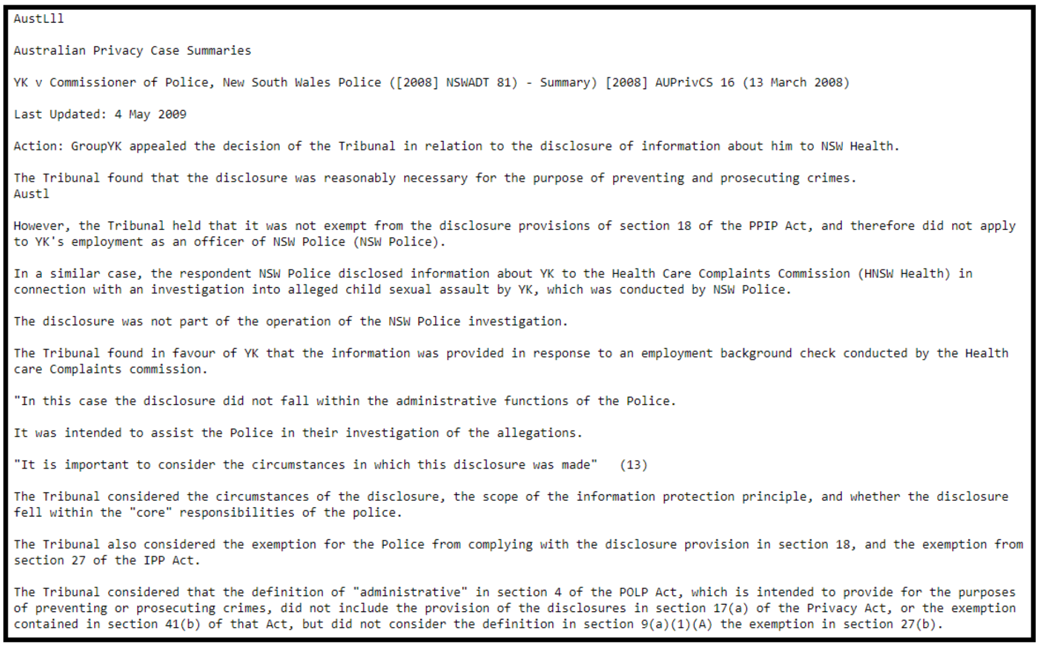
The Police subsequently provided the material requested on the condition that it be used only for NSW Healt's investigations and that it be kept securely and not inappropriately disclosed.

Preliminary issues YK's application for review by the Tribunal alleged that the Police's conduct in disclosing the information in the Police brief to his employer was a breach of the PPIP Act.

As a preliminary issue, the Police argued that its conduct was permitted by the exemption for the Police from compliance with the IPPs except in connection with the exercise of the its administrative and educative functions (section 27).

Alternatively, the Police's conduct was permitted by the section 41 Direction of the Privacy Commissioner, Processing of Information by Public Sector Agencies in Relation to their Investigative Functions.

Figure 4.8: Reference Headnote of the Judgment from AustL-II.



AustLll

Australian Privacy Case Summaries

YK v Commissioner of Police, New South Wales Police ([2008] NSWADT 81) - Summary) [2008] AUPrivCS 16 (13 March 2008)

Last Updated: 4 May 2009

Action: GroupYK appealed the decision of the Tribunal in relation to the disclosure of information about him to NSW Health.

The Tribunal found that the disclosure was reasonably necessary for the purpose of preventing and prosecuting crimes.
Austl

However, the Tribunal held that it was not exempt from the disclosure provisions of section 18 of the PPIP Act, and therefore did not apply to YK's employment as an officer of NSW Police (NSW Police).

In a similar case, the respondent NSW Police disclosed information about YK to the Health Care Complaints Commission (HNSW Health) in connection with an investigation into alleged child sexual assault by YK, which was conducted by NSW Police.

The disclosure was not part of the operation of the NSW Police investigation.

The Tribunal found in favour of YK that the information was provided in response to an employment background check conducted by the Health care Complaints commission.

"In this case the disclosure did not fall within the administrative functions of the Police.

It was intended to assist the Police in their investigation of the allegations.

"It is important to consider the circumstances in which this disclosure was made"   (13)

The Tribunal considered the circumstances of the disclosure, the scope of the information protection principle, and whether the disclosure fell within the "core" responsibilities of the police.

The Tribunal also considered the exemption for the Police from complying with the disclosure provision in section 18, and the exemption from section 27 of the IPP Act.

The Tribunal considered that the definition of "administrative" in section 4 of the POLP Act, which is intended to provide for the purposes of preventing or prosecuting crimes, did not include the provision of the disclosures in section 17(a) of the Privacy Act, or the exemption contained in section 41(b) of that Act, but did not consider the definition in section 9(a)(1)(A) the exemption in section 27(b).

Figure 4.9: System generated Headnote of the Judgment from AustL-II.

4.4.1 Experiments and Results

Different performance metrices are used to evaluate different machine learning algorithms. Accuracy and loss are the two most well-known metrices in the field of machine learning. We have performed experiments to analyze the behaviour of our model through loss and accuracy graphs after each iteration of optimization.

Analayzing the loss graph of a model can provide us an insight to how well a model is performing. Figure 4.10 shows the value for the cost function in terms of the number of epochs for the legal-led-base-8192 model on the SCP dataset.

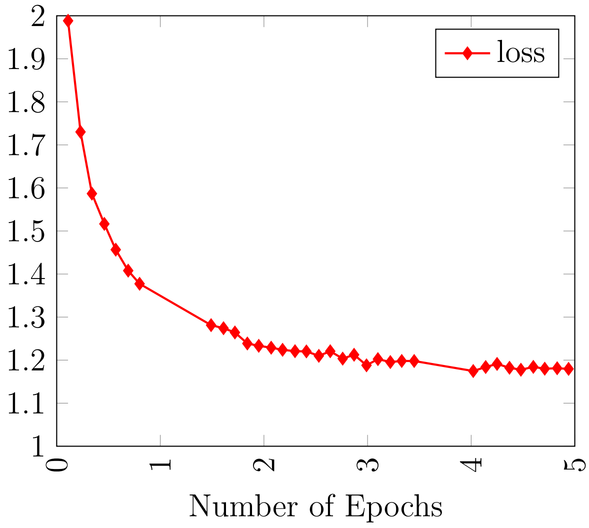


Figure 4.10: Loss vs. Number of Epochs for legal-led-base-8192.

The loss is decreased with the increase in the number of epochs. It is an implication of how well a model is performing after each iteration of optimization. As the number of epochs increases, the curve goes from underfitting to optimal to overfitting curve. The number of epochs are not significant but the optimal loss value between the underfitting and the overfitting curve. We intend to find the optimal number of epochs for our training. The training should continue as long as the error keeps dropping. The optimal number of epochs would be when the drop in loss has become constant.

Accuracy and loss measure different things. Accuracy determines the performance of a model whereas the loss is indicative of how poorly or well a model behaves after each iteration of optimization. Whereas, they appear to be inversely proportional to each other. As the loss value decreases, the accuracy increases. Figure 4.11 shows the value of the accuracy with respect to the number of epochs for the legal-led-base-8192 model on the SCP dataset.

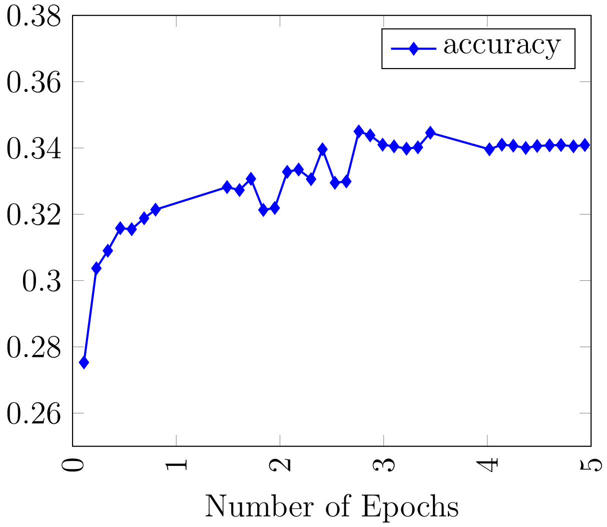


Figure 4.11: Accuracy vs. Number of Epochs for legal-led-base-8192.

Accuracy is an implication of a model's performance after each iteration of optimization, which means that the total count of predictions where the predicted value is equal to the true value. In this case, it is the count of longest common sub-sequence between the gold summary and the candidate summary. The accuracy is increased with the increase in the number of epochs. We intend to find the optimal number of epochs for our training. The training should continue as long as the accuracy keeps increasing. The optimal number of epochs would be when the increase in accuracy has become constant. The accuracy is increased with the number of epochs and it was stable and constant in the last iterations, making it a suitable fine-tuned hyper-parameter value. This indicates 5 to be an optimal number for epochs in this case.

4.4.2 Discussion and Analysis

In this section, we discuss in detail the model performance from different aspects, where it is better and how can it be improved. The results have improved significantly with the model fine-tuned on the legal domain in comparison to the base model. Since, the size of our prepared dataset is small, training the model from scratch will cause the issue of over-fitting. Moreover, fine-tuning a model with a dataset from different domain will lead to the issue of Out-of-Vocabulary (OOV) words. Therefore, the approach of transfer learning with a model pre-trained on a dataset with a similar domain overcame this issue. With the model fine-tuned on the legal sec-litigation-releases [101], the issue of Out-of-Vocabulary (OOV) has been addressed. A model trained using such a vocabulary will perform well on these two datasets. The base model suffers from poor generalization to other unseen datasets, and thus didn't show improved results on the legal domain dataset. Based on the results presented in the table 4.2 and 4.3, we can say that if the model is fine-tuned further on a larger dataset with similar document structure, accuracy can be improved further because this will address the issue of Out-of-Vocabulary (OOV) words. However, it requires an effort to prepare a very large dataset.

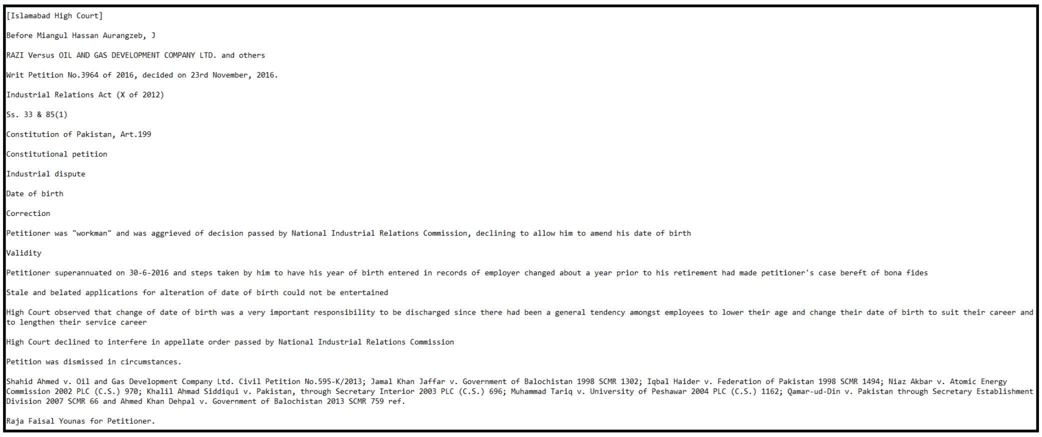
With regards to the input sequence length, the Longformer Encoder-Decoder (LED) model uses Longformer which can handle up to $16,384$ input tokens. If the input document exceeds this input sequence length, longformer will not be able to handle them. However, there could be many aspects that can be considered in which the input sequence length could be handled, and the model performance could be improved. Whilst we have not specified any criteria to select sentences for the generated summary from separated different portions of the document depending upon its classification such as introduction, context, juridical, background, conclusion, etc., for it will make sure to select sentences from all parts of the document and decrease the input document length. The legal text documents and judgments follow a specific structure. Text with the same subject form a thematic segmentation. But, for such kind of thematic segmentation, we require a help from legal experts to provide such a baseline. Selecting sentences for the final summary from pre-defined paragraphs will produce more meaningful results. In this case, the generated summary will be representative of all parts of the judgment.

We have also analyzed the results of the fine-tuned model on low ROUGE scores. Table 4.4 shows the lowest five ROUGE scores (1,2 and L) for the fine-tuned model on the SCP and IHCP dataset.

Table 4.4: Results of lowest rouge f-scores for the fine-tuned model on the SCP and IHCP dataset.

rouge-1	rouge-2	rouge-L
2.36	1.17	2.25
2.77	1.61	2.29
2.87	3.47	2.87
5.63	3.87	4.99
7.57	4.50	6.49

With the abstractive summarization, ROUGE is only used as an indicator of how much the machine-written summary overlaps with the human written summary, because it does not consider the semantic meaning. Therefore, low ROUGE scores, sometimes, do not give us a good indication because of the use of other similar terms in the machine generated summary. We used ROUGE longest common subsequence (ROUGE-L) for text summarization because it has the most overlap. Figures 4.12 and 4.13 depict a scenario in which the machine-generated summary is factually correct, but the ROUGE score (2.87 percent) did not provide us with a good signal because it indicated that the summary is mediocre.



[Islamabad High Court]

Before Miangul Hassan Aurangzeb, J

RAZI Versus OIL AND GAS DEVELOPMENT COMPANY LTD. and others

Writ Petition No.3964 of 2016, decided on 23rd November, 2016.

Industrial Relations Act (X of 2012)

Ss. 33 & 85(1)

Constitution of Pakistan, Art.199

Constitutional petition

Industrial dispute

Date of birth

Correction

Petitioner was "workman" and was aggrieved of decision passed by National Industrial Relations Commission, declining to allow him to amend his date of birth

Validity

Petitioner superannuated on 30-6-2016 and steps taken by him to have his year of birth entered in records of employer changed about a year prior to his retirement had made petitioner's case bereft of bona fides

Stale and belated applications for alteration of date of birth could not be entertained

High Court observed that change of date of birth was a very important responsibility to be discharged since there had been a general tendency amongst employees to lower their age and change their date of birth to suit their career and to lengthen their service career

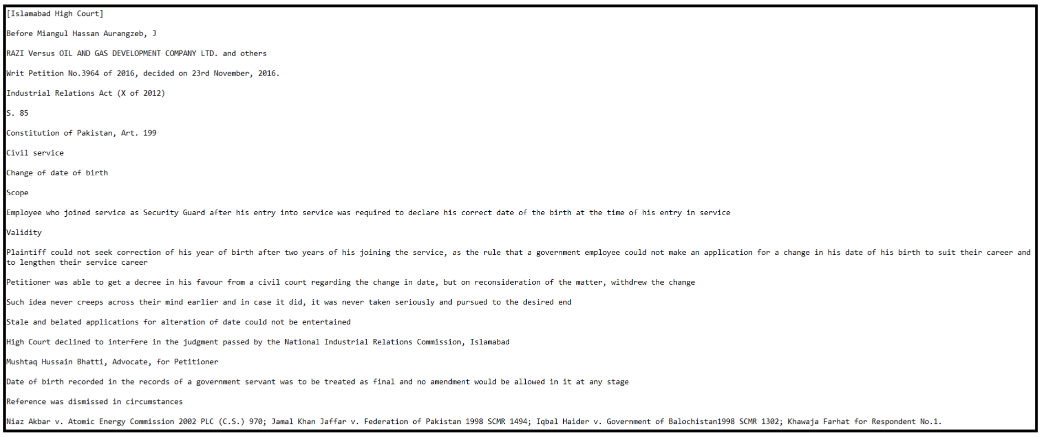
High Court declined to interfere in appellate order passed by National Industrial Relations Commission

Petition was dismissed in circumstances.

Shahid Ahmed v. Oil and Gas Development Company Ltd. Civil Petition No.595-K/2013; Jamal Khan Jaffar v. Government of Balochistan 1998 SCMR 1302; Iqbal Haider v. Federation of Pakistan 1998 SCMR 1494; Niaz Akbar v. Atomic Energy Commission 2002 PLC (C.S.) 970; Khalil Ahmad Siddiqui v. Pakistan, through Secretary Interior 2003 PLC (C.S.) 696; Muhammad Tariq v. University of Peshawar 2004 PLC (C.S.) 1162; Qamar-ud-Din v. Pakistan through Secretary Establishment Division 2007 SCMR 66 and Ahmed Khan Dehpal v. Government of Balochistan 2013 SCMR 759 ref.

Raja Faisal Younas for Petitioner.

Figure 4.12: Reference Headnote of the Judgment from IHCP.



[Islamabad High Court]

Before Miangul Hassan Aurangzeb, J

RAZI Versus OIL AND GAS DEVELOPMENT COMPANY LTD. and others

Writ Petition No.3964 of 2016, decided on 23rd November, 2016.

Industrial Relations Act (X of 2012)

S. 85

Constitution of Pakistan, Art. 199

Civil service

Change of date of birth

Scope

Employee who joined service as Security Guard after his entry into service was required to declare his correct date of the birth at the time of his entry in service

Validity

Plaintiff could not seek correction of his year of birth after two years of his joining the service, as the rule that a government employee could not make an application for a change in his date of his birth to suit their career and to lengthen their service career

Petitioner was able to get a decree in his favour from a civil court regarding the change in date, but on reconsideration of the matter, withdrew the change

Such idea never creeps across their mind earlier and in case it did, it was never taken seriously and pursued to the desired end

Stale and belated applications for alteration of date could not be entertained

High Court declined to interfere in the judgment passed by the National Industrial Relations Commission, Islamabad

Mushtaq Hussain Bhatti, Advocate, for Petitioner

Date of birth recorded in the records of a government servant was to be treated as final and no amendment would be allowed in it at any stage

Reference was dismissed in circumstances

Niaz Akbar v. Atomic Energy Commission 2002 PLC (C.S.) 970; Jamal Khan Jaffar v. Federation of Pakistan 1998 SCMR 1494; Iqbal Haider v. Government of Balochistan1998 SCMR 1302; Khawaja Farhat for Respondent No.1.

Figure 4.13: System generated Headnote of the Judgment from IHCP.

ROUGE is considered as an intrinsic evaluation, whereas extrinsic evaluation is also as much necessary. Intrinsic evaluations measure the performance against a defined standard while extrinsic evaluations involves the human judgment. If the summary is well-written, and covers all the important facts of the source judgment, and required information, the user will be able to answer all the related questions. In this case, a set of related questions need to be prepared. In another scenario, if the legal expert is satisfied with the produced summary, we can consider it as a true one. There could be many possibilities to integrate such kind of improvements but involving humans is always an expensive task.

Chapter 5

Conclusion and Future Work

5.1 Conclusions

In this research, we have employed transfer learning based transformer model for legal text summarization. The results are collected on the judgments from the Supreme Court of Pakistan (SCP) and Islamabad High Court of Pakistan (IHCP). As a metric for our evaluation, we have used the ROUGE metric i.e., the de facto standard evaluation metric for text summarization. The results have been verified on the judgments from AustL-II. The results obtained on the prepared datasets using the approach of transfer learning are quite satisfactory.

5.2 Future Work

There is still research in overcoming the token limit to summarize very long documents. The future efforts can concentrate on overcoming the word token limit, since it seems to be a limitation of our system. Also, preparing a large dataset is expected to be reflected for the improved results, since the deep neural network models are data hungry. In addition to that, different variants of the trained model can be explored in further research to exploit the full potential of this approach. Various features can be extracted from the judgments based on the characteristics of the legal documents to design a better text suumarization model. This needs support from the professional lawyers to prepare a dataset.

Bibliography

[1] V. Pandya, "Automatic text summarization of legal cases: A hybrid approach," *arXiv preprint arXiv:1908.09119*, 2019.

[2] I. Beltagy, M. E. Peters, and A. Cohan, "Longformer: The long-document transformer," *arXiv:2004.05150*, 2020.

[3] W. S. El-Kassas, C. R. Salama, A. A. Rafea, and H. K. Mohamed, "Automatic text summarization: A comprehensive survey," *Expert Systems with Applications*, p. 113679, 2020.

[4] S. Adhikari *et al.*, "Nlp based machine learning approaches for text summarization," in *2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 535–538, IEEE, 2020.

[5] O.-M. Sulea, M. Zampieri, S. Malmasi, M. Vela, L. P. Dinu, and J. Van Genabith, "Exploring the use of text classification in the legal domain," *arXiv preprint arXiv:1710.09306*, 2017.

[6] D. R. Radev, E. Hovy, and K. McKeown, "Introduction to the special issue on summarization," *Computational linguistics*, vol. 28, no. 4, pp. 399–408, 2002.

[7] K. S. Jones *et al.*, "Automatic summarizing: factors and directions," *Advances in automatic text summarization*, pp. 1–12, 1999.

[8] D. Das and A. Martins, "A survey on automatic text summarization. literature survey for the course language and statistics ii," tech. rep., Technical report, Carnegie Mellon University, 2007.

[9] X. Mao, S. Huang, L. Shen, R. Li, and H. Yang, "Single document summarization using the information from documents with the same topic," *Knowledge-Based Systems*, p. 107265, 2021.

[10] C. Ma, W. E. Zhang, M. Guo, H. Wang, and Q. Z. Sheng, "Multi-document summarization via deep learning techniques: A survey," *arXiv preprint arXiv:2011.04843*, 2020.

[11] A. Sinha, A. Yadav, and A. Gahlot, "Extractive text summarization using neural networks," *arXiv preprint arXiv:1802.10137*, 2018.

[12] A. Kanapala, S. Pal, and R. Pamula, "Text summarization from legal documents: a survey," *Artificial Intelligence Review*, vol. 51, no. 3, pp. 371–402, 2019.

[13] D. Suleiman and A. Awajan, "Deep learning based abstractive text summarization: Approaches, datasets, evaluation measures, and challenges," *Mathematical Problems in Engineering*, vol. 2020, 2020.

[14] C. Sunitha, A. Jaya, and A. Ganesh, "A study on abstractive summarization techniques in indian languages," *Procedia Computer Science*, vol. 87, pp. 25–31, 2016.

[15] "Information access in the legal domain." `https://www.isical.ac.in/~fire/2014/legal.html`. [Online; accessed 13-Sept-2021].

[16] "Federal Court of Australia." `https://www.fedcourt.gov.au/`. [Online; accessed 13-Sept-2021].

[17] "UCI Machine Learning Repository: Legal Case Report Dataset." `https://archive.ics.uci.edu/ml/datasets/Legal+Case+Reports`. [Online; accessed 13-Sept-2021].

[18] D. Anand and R. Wagh, "Effective deep learning approaches for summarization of legal texts," *Journal of King Saud University-Computer and Information Sciences*, 2019.

[19] "Over 1.8 million cases pending in Pakistanís courts." `https://www.dawn.com/news/1384319`. [Online; accessed 13-Sept-2021].

[20] "Cases & Legislation." `https://www.austlii.edu.au/database-cases.html`. [Online; accessed 13-Sept-2021].

[21] "Canadian Legal Information Institute." `https://www.canlii.org/en/`. [Online; accessed 13-Sept-2021].

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, pp. 5998–6008, 2017.

[23] Y. Keneshloo, N. Ramakrishnan, and C. K. Reddy, "Deep transfer reinforcement learning for text summarization," in *Proceedings of the 2019 SIAM International Conference on Data Mining*, pp. 675–683, SIAM, 2019.

[24] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[25] Y. He, Z. Zhu, Y. Zhang, Q. Chen, and J. Caverlee, "Infusing disease knowledge into bert for health question answering, medical inference and disease name recognition," *arXiv preprint arXiv:2010.03746*, 2020.

[26] Y. Zhang, G. Xu, Y. Wang, D. Lin, F. Li, C. Wu, J. Zhang, and T. Huang, "A question answering-based framework for one-step event argument extraction," *IEEE Access*, vol. 8, pp. 65420–65431, 2020.

[27] W. Yang, Y. Xie, A. Lin, X. Li, L. Tan, K. Xiong, M. Li, and J. Lin, "End-to-end open-domain question answering with bertserini," *arXiv preprint arXiv:1902.01718*, 2019.

[28] E. Amer, A. Hazem, O. Farouk, A. Louca, Y. Mohamed, and M. Ashraf, "A proposed chatbot framework for covid-19," in *2021 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)*, pp. 263–268, IEEE, 2021.

[29] S. Yoo and O. Jeong, "An intelligent chatbot utilizing bert model and knowledge graph," *Journal of Society for e-Business Studies*, vol. 24, no. 3, 2020.

[30] T. Nguyen and M. Shcherbakov, "Enhancing rasa nlu model for vietnamese chatbot," *International Journal of Open Information Technologies*, vol. 9, no. 1, pp. 31–36, 2021.

[31] K. Yawata, T. Suzuki, K. Kiryu, and K. Mohri, "Performance evaluation of japanese bert model for intent classification using a chatbot," in *The 35th Annual Conference of the Japanese Society for Artificial Intelligence, 2021*, pp. 2N4IS2c05–2N4IS2c05, Japanese Society for Artificial Intelligence, 2021.

[32] C. Sun, X. Qiu, Y. Xu, and X. Huang, "How to fine-tune bert for text classification," pp. 194–206, Springer, 2019.

[33] A. Adhikari, A. Ram, R. Tang, and J. Lin, "Docbert: Bert for document classification," *arXiv preprint arXiv:1904.08398*, 2019.

[34] Z. Gao, A. Feng, X. Song, and X. Wu, "Target-dependent sentiment classification with bert," *IEEE Access*, vol. 7, pp. 154290–154299, 2019.

[35] Q. Chen, Z. Zhuo, and W. Wang, "Bert for joint intent classification and slot filling," *arXiv preprint arXiv:1902.10909*, 2019.

[36] Y. Liu and M. Lapata, "Text summarization with pretrained encoders," *arXiv preprint arXiv:1908.08345*, 2019.

[37] T. Shi, Y. Keneshloo, N. Ramakrishnan, and C. K. Reddy, "Neural abstractive text summarization with sequence-to-sequence models," *ACM Transactions on Data Science*, vol. 2, no. 1, pp. 1–37, 2021.

[38] A. Joshi, E. Fidalgo, E. Alegre, and U. de Leíon, "Deep learning based text summarization: approaches, databases and evaluation measures," in *Proceedings of the International Conference of Applications of Intelligent Systems*, 2018.

[39] Y. Dong, "A survey on neural network-based summarization methods," *arXiv preprint arXiv:1804.04589*, 2018.

[40] A. Mahajani, V. Pandya, I. Maria, and D. Sharma, "A comprehensive survey on extractive and abstractive techniques for text summarization," *Ambient Communications and Computer Systems*, pp. 339–351, 2019.

[41] J. Zhang, Y. Zhao, M. Saleh, and P. Liu, "Pegasus: Pre-training with extracted gap-sentences for abstractive summarization," in *International Conference on Machine Learning*, pp. 11328–11339, PMLR, 2020.

[42] R. A. García-Herníandez and Y. Ledeneva, "Word sequence models for single text summarization," in *2009 Second International Conferences on Advances in Computer-Human Interactions*, pp. 44–48, IEEE, 2009.

[43] K. Sarkar, "An approach to summarizing bengali news documents," in *proceedings of the International Conference on Advances in Computing, Communications and Informatics*, pp. 857–862, 2012.

[44] A. Mackey and I. Cuevas, "Automatic text summarization within big data frameworks," *Journal of Computing Sciences in Colleges*, vol. 33, no. 5, pp. 26–32, 2018.

[45] K. Merchant and Y. Pande, "Nlp based latent semantic analysis for legal text summarization," in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 1803–1807, IEEE, 2018.

[46] Y. Gong and X. Liu, "Generic text summarization using relevance measure and latent semantic analysis," in *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 19–25, 2001.

[47] M. G. Ozsoy, F. N. Alpaslan, and I. Cicekli, "Text summarization using latent semantic analysis," *Journal of Information Science*, vol. 37, no. 4, pp. 405–417, 2011.

[48] J. Steinberger, K. Jezek, *et al.*, "Using latent semantic analysis in text summarization and summary evaluation," *Proc. ISIM*, vol. 4, pp. 93–100, 2004.

[49] H. Zheng and M. Lapata, "Sentence centrality revisited for unsupervised summarization," *arXiv preprint arXiv:1906.03508*, 2019.

[50] M. Litvak and M. Last, "Graph-based keyword extraction for single-document summarization," in *Coling 2008: Proceedings of the workshop Multi-source Multilingual Information Extraction and Summarization*, pp. 17–24, 2008.

[51] F. D. Malliaros and K. Skianis, "Graph-based term weighting for text categorization," in *Proceedings of the 2015 IEEE/ACM International*

Conference on Advances in Social Networks Analysis and Mining 2015, pp. 1473–1479, 2015.

[52] R. Mihalcea, "Graph-based ranking algorithms for sentence extraction, applied to text summarization," in *Proceedings of the ACL interactive poster and demonstration sessions*, pp. 170–173, 2004.

[53] A. F. Sevilla, A. Ferníandez-Isabel, and A. Díaz, "Enriched semantic graphs for extractive text summarization," in *Conference of the Spanish Association for Artificial Intelligence*, pp. 217–226, Springer, 2016.

[54] S. Verma and V. Nidhi, "Extractive summarization using deep learning," *arXiv preprint arXiv:1708.04439*, 2017.

[55] N. Alami, M. Meknassi, and N. En-nahnahi, "Enhancing unsupervised neural networks based text summarization with word embedding and ensemble learning," *Expert systems with applications*, vol. 123, pp. 195–211, 2019.

[56] S. Xu, X. Zhang, Y. Wu, F. Wei, and M. Zhou, "Unsupervised extractive summarization by pre-training hierarchical transformers," *arXiv preprint arXiv:2010.08242*, 2020.

[57] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[58] A. Padmakumar and A. Saran, "Unsupervised text summarization using sentence embeddings," *Technical Report, University of Texas at Austin*, pp. 1–9, 2016.

[59] I. Mani, D. House, G. Klein, L. Hirschman, T. Firmin, and B. M. Sundheim, "The tipster summac text summarization evaluation," in *Ninth Conference of the European Chapter of the Association for Computational Linguistics*, pp. 77–85, 1999.

[60] K. Ganesan, C. Zhai, and J. Han, "Opinosis: A graph based approach to abstractive summarization of highly redundant opinions," 2010.

[61] J. Steinberger and K. Ježek, "Text summarization and singular value decomposition," in *International Conference on Advances in Information Systems*, pp. 245–254, Springer, 2004.

[62] J. Li, M.-T. Luong, and D. Jurafsky, "A hierarchical neural autoencoder for paragraphs and documents," *arXiv preprint arXiv:1506.01057*, 2015.

[63] S. Dohare, H. Karnick, and V. Gupta, "Text summarization using abstract meaning representation," *arXiv preprint arXiv:1706.01678*, 2017.

[64] J. Cheng and M. Lapata, "Neural summarization by extracting sentences and words," *arXiv preprint arXiv:1603.07252*, 2016.

[65] R. Nallapati, B. Zhou, C. Gulcehre, B. Xiang, *et al.*, "Abstractive text summarization using sequence-to-sequence rnns and beyond," *arXiv preprint arXiv:1602.06023*, 2016.

[66] A. Farzindar, "Atefeh farzindar and guy lapalme,'letsum, an automatic legal text summarizing system in t. gordon (ed.), legal knowledge and information systems. jurix 2004: The seventeenth annual conference. amsterdam: Ios press, 2004, pp. 11-18.," in *Legal Knowledge and Information Systems: JURIX 2004, the Seventeenth Annual Conference*, vol. 120, p. 11, IOS Press, 2004.

[67] A. Kanapala, S. Jannu, and R. Pamula, "Summarization of legal judgments using gravitational search algorithm," *Neural Computing and Applications*, vol. 31, no. 12, pp. 8631–8639, 2019.

[68] M.-Y. Kim, Y. Xu, and R. Goebel, "Summarization of legal texts with high cohesion and automatic compression rate," in *JSAI International Symposium on Artificial Intelligence*, pp. 190–204, Springer, 2012.

[69] N. Bansal, A. Sharma, and R. Singh, "Fuzzy ahp approach for legal judgement summarization," *Journal of Management Analytics*, vol. 6, no. 3, pp. 323–340, 2019.

[70] H. Oufaida, O. Nouali, and P. Blache, "Minimum redundancy and maximum relevance for single and multi-document arabic text summarization," *Journal of King Saud University-Computer and Information Sciences*, vol. 26, no. 4, pp. 450–461, 2014.

[71] R. K. Venkatesh, "Legal documents clustering and summarization using hierarchical latent dirichlet allocation," *IAES International Journal of Artificial Intelligence*, vol. 2, no. 1, 2013.

[72] S. Polsley, P. Jhunjhunwala, and R. Huang, "Casesummarizer: a system for automated summarization of legal texts," in *Proceedings of COLING 2016, the 26th international conference on Computational Linguistics: System Demonstrations*, pp. 258–262, 2016.

[73] H. Yamada, S. Teufel, and T. Tokunaga, "Designing an annotation scheme for summarizing japanese judgment documents," in *2017 9th International Conference on Knowledge and Systems Engineering (KSE)*, pp. 275–280, IEEE, 2017.

[74] A. Joshi, E. Fidalgo, E. Alegre, and L. Ferníandez-Robles, "Summcoder: An unsupervised framework for extractive text summarization based on deep auto-encoders," *Expert Systems with Applications*, vol. 129, pp. 200–215, 2019.

[75] S. N. Truong, N. Le Minh, K. Satoh, T. Satoshi, and A. Shimazu, "Single and multiple layer bi-lstmcrf for recognizing requisite and effectuation parts in legal texts," in *Proc. of the 2nd Workshop on Automated Semantic Analysis of Information in Legal Texts*, 2017.

[76] T.-S. Nguyen, L.-M. Nguyen, S. Tojo, K. Satoh, and A. Shimazu, "Recurrent neural network-based models for recognizing requisite and effectuation parts in legal texts," *Artificial Intelligence and Law*, vol. 26, no. 2, pp. 169–199, 2018.

[77] I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, and I. Androutsopoulos, "Legal-bert: The muppets straight out of law school," *arXiv preprint arXiv:2010.02559*, 2020.

[78] I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, and I. Androutsopoulos, "Legal-bert:"preparing the muppets for court"," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pp. 2898–2904, 2020.

[79] E. Elwany, D. Moore, and G. Oberoi, "Bert goes to law school: Quantifying the competitive advantage of access to large legal corpora in contract understanding," *arXiv preprint arXiv:1911.00473*, 2019.

[80] L. Sanchez, J. He, J. Manotumruksa, D. Albakour, M. Martinez, and A. Lipani, "Easing legal news monitoring with learning to rank and bert," *Advances in Information Retrieval*, vol. 12036, p. 336, 2020.

[81] P. Bambroo and A. Awasthi, "Legaldb: Long distilbert for legal document classification," in *2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*, pp. 1–4, IEEE, 2021.

[82] H. Westermann, J. Savelka, and K. Benyekhlef, "Paragraph similarity scoring and fine-tuned bert for legal information retrieval and entailment," in *JSAI International Symposium on Artificial Intelligence*, pp. 269–285, Springer, 2020.

[83] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing," *ieee Computational intelligenCe magazine*, vol. 13, no. 3, pp. 55–75, 2018.

[84] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," *arXiv preprint arXiv:1910.13461*, 2019.

[85] Z. Nasar, S. W. Jaffry, and M. K. Malik, "Textual keyword extraction and summarization: State-of-the-art," *Information Processing & Management*, vol. 56, no. 6, p. 102088, 2019.

[86] V. Tran, M. L. Nguyen, and K. Satoh, "Automatic catchphrase extraction from legal case documents via scoring using deep neural networks," *arXiv preprint arXiv:1809.05219*, 2018.

[87] L. Zhong, Z. Zhong, Z. Zhao, S. Wang, K. D. Ashley, and M. Grabmair, "Automatic summarization of legal decisions using iterative masking of predictive sentences," in *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law*, pp. 163–172, 2019.

[88] F. Galgani, P. Compton, and A. Hoffmann, "Hauss: Incrementally building a summarizer combining multiple techniques," *International journal of human-computer studies*, vol. 72, no. 7, pp. 584–605, 2014.

[89] S. Edunov, A. Baevski, and M. Auli, “Pre-trained language model representations for language generation,” *arXiv preprint arXiv:1903.09722*, 2019.

[90] S. Rothe, S. Narayan, and A. Severyn, “Leveraging pre-trained checkpoints for sequence generation tasks,” *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 264–280, 2020.

[91] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, “Deep contextualized word representations,” *arXiv preprint arXiv:1802.05365*, 2018.

[92] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, “Improving language understanding by generative pre-training,” 2018.

[93] Y. Dong, Y. Shen, E. Crawford, H. van Hoof, and J. C. K. Cheung, “Banditsum: Extractive summarization as a contextual bandit,” *arXiv preprint arXiv:1809.09672*, 2018.

[94] X. Zhang, F. Wei, and M. Zhou, “Hibert: Document level pre-training of hierarchical bidirectional transformers for document summarization,” *arXiv preprint arXiv:1905.06566*, 2019.

[95] D. Hendrycks and K. Gimpel, “Gaussian error linear units (gelus),” *arXiv preprint arXiv:1606.08415*, 2016.

[96] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “Roberta: A robustly optimized bert pretraining approach,” *arXiv preprint arXiv:1907.11692*, 2019.

[97] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, “Glue: A multi-task benchmark and analysis platform for natural language understanding,” *arXiv preprint arXiv:1804.07461*, 2018.

[98] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, “Squad: 100,000+ questions for machine comprehension of text,” *arXiv preprint arXiv:1606.05250*, 2016.

[99] S. Narayan, S. B. Cohen, and M. Lapata, “Don’t give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization,” *arXiv preprint arXiv:1808.08745*, 2018.

[100] "LED for legal summarization of documents." https://huggingface.co/nsi319/legal-led-base-16384. [Online; accessed 9-Oct-2021].

[101] "U.S. Securities and Exchange Commission." https://www.sec.gov/litigation/litreleases.htm. [Online; accessed 9-Oct-2021].

[102] "pyrouge." https://github.com/andersjo/pyrouge. [Online; accessed 9-Oct-2021].

[103] "Sentence Splitter - Maltese Language Software Services." http://metanet4u.research.um.edu.mt/SentenceSplitter.jsp. [Online; accessed 9-Oct-2021].

[104] K. Ganesan, "Rouge 2.0: Updated and improved measures for evaluation of summarization tasks," *arXiv preprint arXiv:1803.01937*, 2018.

[105] K. W. Church, "Word2vec," *Natural Language Engineering*, vol. 23, no. 1, pp. 155–162, 2017.

[106] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.