

Anuj Mishra

AnujMishra24dsc24_MajorThesis_Report

 M.TECH MAJOR PROJECT II

Document Details

Submission ID

trn:oid:::27535:143703244

Submission Date

Jun 20, 2026, 2:25 PM GMT+5:30

Download Date

Jun 20, 2026, 2:30 PM GMT+5:30

File Name

AnujMishra24dsc24_MajorThesis_Report.pdf

File Size

3.1 MB

51 Pages

14,741 Words

82,860 Characters

Dr.

Digitally signed
by Dr. Rahul

Submission ID trn:oid:::27535:143703244

4% Overall Similarity Rahul

Date: 2026.06.20
19:07:55 +05'30'

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text
- Small Matches (less than 8 words)

Match Groups

-  **38 Not Cited or Quoted 3%**
Matches with neither in-text citation nor quotation marks
-  **12 Missing Quotations 1%**
Matches that are still very similar to source material
-  **0 Missing Citation 0%**
Matches that have quotation marks, but no in-text citation
-  **0 Cited and Quoted 0%**
Matches with in-text citation present, but no quotation marks

Top Sources

- 3%  Internet sources
- 2%  Publications
- 2%  Submitted works (Student Papers)

Match Groups

- 38 Not Cited or Quoted 3%**
Matches with neither in-text citation nor quotation marks
- 12 Missing Quotations 1%**
Matches that are still very similar to source material
- 0 Missing Citation 0%**
Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%**
Matches with in-text citation present, but no quotation marks

Top Sources

- 3% Internet sources
- 2% Publications
- 2% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

- 1** Internet
arxiv.org <1%
- 2** Publication
"Natural Language Processing and Chinese Computing", Springer Science and Bu... <1%
- 3** Publication
"Advances in Information Retrieval", Springer Science and Business Media LLC, 20... <1%
- 4** Internet
www.arxiv-vanity.com <1%
- 5** Internet
dspace.vutbr.cz <1%
- 6** Internet
www.next.gr <1%
- 7** Student papers
Cyprus University of Technology on 2025-11-12 <1%
- 8** Student papers
Heriot-Watt University on 2024-04-17 <1%
- 9** Publication
Jorge Andoni Valverde Tohalino. "Extractive document summarization using com... <1%
- 10** Student papers
CSU, San Jose State University on 2023-12-05 <1%

11	Publication	Smriti Singh, Anil Kumar Singh. "Evaluating Transformer-Based Models for Abstra...	<1%
12	Student papers	The Robert Gordon University on 2024-09-30	<1%
13	Internet	purehost.bath.ac.uk	<1%
14	Internet	toloka.ai	<1%
15	Student papers	Babes-Bolyai University on 2021-06-15	<1%
16	Internet	dspace.cvut.cz	<1%
17	Publication	Shashi Narayan, Claire Gardent. "Deep Learning Approaches to Text Production", ...	<1%
18	Student papers	University of Essex on 2023-08-25	<1%
19	Student papers	University of Westminster on 2024-09-09	<1%
20	Publication	Wang, Guanghua. "Synergizing Unsupervised and Supervised Learning in Multi-St...	<1%
21	Publication	Xingbo Fu, Feng Gao, Jiang Wu, Xinyu Wei, Fangwei Duan. "Spatiotemporal Attent...	<1%
22	Internet	workshop2017.iwslt.org	<1%
23	Internet	www.cyberfoundry.io	<1%
24	Publication	A. Vivek, V. Susheela Devi. "Chapter 26 SumBART - An Improved BART Model forA...	<1%

25	Student papers	Dublin City University on 2012-06-15	<1%
26	Publication	He, Jingyi. "Evaluation of Multi-Sentence Aggregation in Abstractive Text Summar..."	<1%
27	Student papers	Liverpool John Moores University on 2023-04-18	<1%
28	Publication	Mao, Yuning. "Guided Text Summarization With Limited Supervision", University ...	<1%
29	Student papers	University of Birmingham on 2025-09-01	<1%
30	Student papers	University of York on 2024-06-16	<1%
31	Internet	aclweb.org	<1%
32	Internet	scofizz.github.io	<1%
33	Internet	thinkmind.org	<1%
34	Internet	uu.diva-portal.org	<1%
35	Student papers	Ajman University on 2026-05-06	<1%
36	Student papers	German University of Technology in Oman on 2026-06-10	<1%
37	Publication	N. Radha, R. Swathika, K. R. Uthayan, Mitul Krishna B. "AI-Driven Summarization ...	<1%
38	Publication	Ridam Srivastava, Prabhav Singh, K.P.S. Rana, Vineet Kumar. "A topic modeled un...	<1%

39	Internet	isl.anthropomatik.kit.edu	<1%
40	Internet	link.springer.com	<1%
41	Internet	repository.library.carleton.ca	<1%
42	Internet	repository.mut.ac.ke:8080	<1%
43	Internet	theses.hal.science	<1%
44	Internet	www.dei.unipd.it	<1%
45	Internet	www.gorschek.com	<1%

Abstract

The rapid growth in the amount of digital textual information available on news websites, academic databases, social media sites, legal databases corporate information systems has put a significant cognitive burden on human readers who seek to glean useful information from the information available. Abstractive summarization, which entails the need for computers to construct coherent summaries using new sentences and not snippets of sentences taken directly from the documents, is one of the most difficult and crucial problems in the area of natural language processing. The emergence of large-scale pre-trained transformers has revolutionized abstractive summarization and T5, BART, PEGASUS and FLAN-T5 have set successively better benchmarks on standard evaluation datasets. However, abstractive summarization is still riddled with many unresolved problems, including the problem of hallucination where generative models generate information that is not present in the source document, the inability of ROUGE metrics to evaluate semantic accuracy, lack of statistical rigor in performance comparison assertions and heavy computational costs associated with summarization.

The thesis aims to propose a dual contribution study which will try to fill this gap. In two methodologically connected parts. The first step is a Systematic Literature Review (SLR) is carried out using methodology that was introduced by The report was compiled by Kitchenham and Charters, which is a collection of 42 scholarly papers published between 2020 and 2025, that documents how generative AI methods have evolved for text.2025, that chronicles the evolution of generative AI methods for text. summarization, highlights the state of the art of the model architecture and Identifies the important constraints and future outlook of the benchmark datasets used perspectives. Second, a comparison analysis of how second language English is implemented was conducted. We use 4 popular transformer models: T5-Base (220M), BART-Large (400M), The PEGASUS-Large (570M) and FLAN-T5-Large (780M) is performed with the following cases: experimental setup with two widely used news benchmarks, strictly unified. (CNN/DailyMail and XSum) and the out-of-domain dialogue dataset SAMSUm in the zero-shot learning setup. across three metrics: surface-level lexical quality estimated using ROUGE-1/2/L, semantic similarity estimated using BERTScore and factual consistency estimated using AlignScore.

The empirical results reveal that PEGASUS-Large obtains the highest ROUGE-1 score of 47.21 on the highly compressed XSum dataset due to its task-oriented pre-training objective of Gap Sentence Generation, while BART-Large and PEGASUS-Large obtain nearly identical ROUGE-1 scores on CNN/DailyMail (44.16 vs. 44.17) without statistically significant differences. Importantly, an empirically validated and novel result in this work pertains to the factual consistency aspect: FLAN-T5-Large obtains the highest AlignScore on both datasets (CNN/DailyMail: 0.512; XSum: 0.461) even though it does

not obtain the highest ROUGE scores, clearly demonstrating empirically that instruction tuning is more beneficial for factual consistency than lexical overlap. In addition, FLAN-T5-Large exhibits substantially better cross-domain generalizability on zero-shot evaluation on SAMSum in terms of ROUGE-1 (31.56 vs. 24.37) than BART-Large, thus making instruction tuning the preferable technique for practitioners who need domain-agnostic summarization models. T5-Base, which takes only 0.27 seconds per sample to make an inference and requires 1.9 GB of GPU memory, presents the best cost-performance trade-off.

Chapter 1

Introduction

The amount of textual information that is both created and consumed by people all around the world has never been higher than it is today in the digital age. The sum total of information produced by news agencies, publishers, courts, hospitals and enterprise systems amounts to billions of documents per year, leading to a situation where the sheer volume of information available outstrips the ability of human readers to process it efficiently. In such a scenario, automatic text summarization – the computerized task of producing a summary of a document or set of documents that retains the essential information contained within it – has moved from being an interesting topic for academic research to one with immediate practical value.

Text summarization algorithms can be categorized into two types based on the paradigm used: extractive and abstractive. Extractive text summarization identifies the most relevant sentences and phrases from the source document and forms a summary out of them. Although this method is computationally feasible and is by nature fact-based, extractive summaries often lack coherence, contain redundant information and cannot summarize information that is scattered throughout the text rather than being confined to a particular sentence. In contrast to extractive summarization, abstractive summarization requires that the algorithm generate new sentences which may or may not occur in the source document — a feature that mimics human behavior and enables concise and coherent summaries [1]. However, abstractive summarization requires greater natural language processing capabilities, making it a more difficult task to implement.

The emergence of architectures such as T5 [2], BART [3], A number of researchers have recently undertaken the task of creating larger language models, such as PEGASUS [4] and the GPT series [5] took the state-of-the-art summarization ability to new levels. previously impossible, coherent and fluently written and sometimes persuasive texts. A fairly accurate rendition of the source material. All of these models have the advantage of being user-friendly. Labeled as having been pretrained on extensive amounts of unlabeled text. substantial computational expense and obtaining wide-ranging linguistic On summarization tasks, smaller amounts of knowledge can be used to fine-tune the model, which may subsequently be applied to other, larger amounts of knowledge. amounts of

labeled data. The different pre-training tasks used by each of such as span masking for T5, denoising for BART, gap sentence generation for PEGASUS and instruction tuning for FLAN-T5 highlight the depth of experimentation conducted during this period.

1.1 Motivation and Problem Statement

But, all the developments that have occurred through Despite its many successes in transformer-based generative architectures, abstractive summarization remains challenging. There are significant issues that need to be addressed to enable deployment and reliability of current solutions. One of these phenomena is of special significance: as 'hallucination' or factual inconsistency, which is the ability to: Content based on generative models: producing content that is not supported or contradicted by the generative models. source document. The results of Maynez et al. indicate that. According to [6], between 25 to 30The current state of the art models produce such summaries that are: These inconsistencies, if they impact accuracy and information, can have significant consequences. The domain in question should have the ability of being verified. This is particularly true in the medical context, where introducing a drug dosage or test result that does not appear in the actual record might put the patient at risk.

The second critical issue lies in the insufficiency of current methodologies for evaluation. ROUGE [7] is the prevalent automatic metric used for evaluating summarization performance; ROUGE calculates the overlap of n-grams between the automatically generated summary and one or multiple reference summaries written by humans. Although ROUGE shows decent correlation with human judgments under experimental conditions, it is inherently a shallow measure based on surface similarity. In other words, ROUGE cannot recognize semantic equivalence, does not measure the accuracy of facts, moreover, tends to reward exact phrase copying from the source rather than paraphrasing. As a result of such dependence on ROUGE, the quality of generated summaries, especially the fidelity of facts, is essentially undetectable.

The third issue is related to the lack of statistics in comparative benchmarking research. Most comparative analysis studies comparing the performance of various models for text summarization do not provide confidence intervals and statistical tests; instead, they publish performance metrics estimated based on one seed point only. Hence, it is impossible to tell whether the performance difference observed between the models being compared represents real differences in capabilities or merely random fluctuations due to sampling.

Fourth, the computational availability of state-of-the-art summarization models is also difficult. With 780 million parameters, the FLAN-T5-Large model needs 7.1 GB of GPU memory during inference, whereas the PEGASUS-Large model needs 6.8 GB with 570 million parameters. This exceeds the capability of the hardware available in many applications in clinical, education and government settings in low-resource environments.

1.2 Research Objectives

This research, as presented in this thesis, has been prompted by the urgent need to find solutions to all these four issues, through a principled and empirical analysis. The specific goals set for the research include the following. The first goal is to carry out a systematic review of the literature in the domain of generative AI for text summarization from 2020 to 2025, based on the SLR approach proposed by Kitchenham and Charters [8], with a view to mapping the state of the art and identifying the main research gaps in the area. The second goal is to perform a comparative evaluation, using a consistent framework, of four state-of-the-art transformers, namely T5-Base, BART-Large, PEGASUS-Large and FLAN-T5-Large, that are trained and evaluated in similar settings on two benchmark datasets (CNN/DailyMail and XSum) zero-shot evaluated on the SAMSum dialogue dataset. Thirdly, the aim is to go beyond standard metrics like ROUGE by introducing BERTScore as a measure of semantic similarity and AlignScore as a measure of factual consistency, thus obtaining a comprehensive profile of model performance which includes properties that ROUGE fails to detect. Fourthly, each of the main findings will be complemented by 95% bootstrap confidence intervals in order to allow statistically meaningful comparison of the models. Finally, the aim is to analyze computational efficiency, measured as inference time and GPU memory consumption.

1.3 Summary of Contributions

The thesis contributes to the domain of automated text summarization in four main ways. First, it conducts a systematic review of 42 papers on generative AI techniques used for text summarization, published between 2020 and 2025, which covers all aspects related to the state of research on the architecture of the model, benchmarks, evaluation and limitations in the field. Second, it conducts a comparative study based on an implementation of transformer models under controlled experimental settings, using multiple metrics of evaluation, including factual consistency. Third, it offers unique empirical findings showing that instruction tuning (FLAN-T5) can lead to consistent improvement in factual faithfulness beyond what ROUGE metrics measure, which is only possible with the use of AlignScore as part of the evaluation procedure. Fourth, it shows that, despite being commonly perceived as a statistically significant gap, the difference in performance between BART-Large and PEGASUS-Large on the CNN/DailyMail dataset is not significant based on multiple-seed analysis and reporting of confidence intervals.

Chapter 2

Technical Background

The current chapter provides the theoretical and technological framework on which the research conducted for this thesis is based. It starts by discussing the core issue of text summarization, then moves on to discuss the historical development of the computational methods used to solve the problem, including statistical approaches, recurrent neural networks and transformers. The chapter ends with an in-depth discussion of the metrics used to evaluate the performance of the transformer-based models discussed in this thesis, along with their theoretical underpinnings and limitations.

2.1 Text Summarization: Definition and Paradigms

The formal definition of text summarization can be expressed as follows: It is the process of creating a compressed representation of one or more source documents wherein the output summary retains the most significant information from the source but at a significantly reduced length compared to the source [9].

The key characteristic of extractive summarization is that it involves the selection and combination of sentences or phrases from the source document, thereby guaranteeing that all tokens of the summary are also tokens of the source. The computational feasibility of extractive summarization is guaranteed by its very nature, since it makes no additions to the source content. On the other hand, extractive summaries tend to suffer from low coherence, as well as from redundancy [1].

Abstractive summarization necessitates the creation of new language, which can entail the use of vocabulary, syntax and combinations of information that were not contained in the original text. It is a more natural way to summarize texts since it allows for generating true summaries instead of mere shortenings, but it also involves the danger of being inaccurate because models that generate new sentences can also generate sentences that include information contradictory to the source text [6].

2.2 Evolution of Neural Approaches

The evolution of technologies in text summarization has passed through various stages. The first approaches to computational text summarization used heuristic score functions based on statistical characteristics of sentences, which included term frequency, their position in the text (initial sentences are usually more important), resemblance to the title and the first paragraph of the document and graph centrality approaches such as TextRank [10]. Such models were efficient, explainable achieved excellent results for summarizing news articles; however, they were not able to model semantic relations between sentences and to produce text.

Seq2Seq models with attention mechanisms [11] are the first truly enabling technology for end-to-end learnable abstractive summarization. A Seq2Seq model consists of an encoder, which encodes the input document into a fixed-dimensional context vector and a decoder, which generates output text using this context vector. The attention mechanism, introduced independently by [3], allowed to overcome the information bottleneck due to the use of the fixed-dimensional context vector by computing a weighted sum of the hidden states of the encoder at each decoding step, with a preference to the most informative parts of the input document. Pointer-Generator Network [12] further improved this architecture by allowing the decoder to copy the words from the input document, in addition to generating the tokens from the fixed vocabulary, which substantially increased factuality and ability to handle rare and out-of-vocabulary words.

The Transformer [13], which replaced recurrent computation by self-attention, allowing each position in the input sequence to attend all other positions, became a real breakthrough in this area. The ability of the Transformer to model long-range dependencies without being affected by the bottleneck of sequential computation that comes from recurrent architecture, as well as its ability to parallelize training allowed to train extremely deep models on large text corpora.

2.3 The Transformer Architecture

Unlike earlier sequence models, the Transformer architecture [13] dispenses entirely with recurrent or convolutional components. Its design is built around two stacks—an encoder and a decoder—each composed of repeated, identical layers. Within the encoder, each layer contains two components: a multi-head self-attention mechanism followed by a feed-forward network applied position-wise. The decoder layer is slightly more involved, containing three components: a masked multi-head self-attention block (which prevents any position from accessing information about subsequent positions), a cross-attention block that draws on the encoder's output and a position-wise feed-forward network. Residual connections and layer normalization are used in each sub-layer as well. The key idea in this design is self-attention: the model constructs each position's representation by com-

binning information from every other position, where the amount of information that is combined depends on how similar the query vector of the position is to the key vectors of the other positions. Formally,

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \quad (2.1)$$

where Q is a matrix representing the query, K and V are matrices representing the keys and values, respectively and d_k specifies the dimensionality of the keys.

The model does this calculation not just once, but in parallel with h attention heads, each working on a smaller subspace. This enables various heads to register various types of relationships in the sequence simultaneously, rather than requiring one attention pass to get it all at once.

A problem with self-attention alone is that it does not directly have a sense of word order—it essentially compares every word in the sequence with each other word in the sequence as an unordered set of vectors. To circumvent this, the information about the position is directly encoded in the input embeddings. The original Transformer used sinusoidal patterns for this, but there are alternatives that have been explored later: position encoding learned at the time of training, relative distance between tokens and rotary positional encodings.

2.4 Pre-Trained Language Models for Summarization

2.4.1 T5: Text-to-Text Transfer Transformer

T5 [2] is the architecture developed by Raffel et al., and it can be regarded as a paradigm shift in terms of the relationship between pre-training and fine-tuning. The fundamental idea of T5 lies in viewing every NLP task as a text-to-text task where the input and the output can both be encoded with text form, therefore making it possible to have a unified model, loss function and training procedure for all kinds of tasks including summarization, translation, classification, and question answering. Specifically, when it comes to summarization, the document will be prefixed with “summarize:” while the reference summary will be the output.

T5 uses the classic encoder-decoder framework of Transformer with relative position encoding. It has been pre-trained on Colossal Clean Crawled Corpus (C4), which is a cleaned web corpus of Common Crawl, through the span masking approach which is motivated by masked language modeling approach: some random spans of texts are masked with special tokens and then the model has to reproduce the masked span at the decoder. T5 has comprehensive knowledge of English grammar, semantics and the real world due to the scale of the pre-training objective and the huge amount of data involved. There are five different sizes of T5 ranging from Small (60M) to XXL (11B); we use the T5-Base (220M).

2.4.2 BART: Bidirectional and Auto-Regressive Transformers

The model of BART [3], developed by Lewis et al., is the denoising autoencoder for the sequence-to-sequence task that includes bidirectional encoder like BERT and auto-regressive decoder like GPT. The basis of the pre-training algorithm of BART is the usage of different text corruption functions that allow the model to reconstruct the originally uncorrupted text. The text corruption functions utilized in the pre-training of BART include token masking, token deletion, text infilling (substitution of the continuous part of text with the single mask token), sentence permutation and document rotation. The variety of the text corruption methods is aimed at the creation of the system where the encoder creates the bidirectional context representation for the corrupted text while the decoder reconstructs the original text.

The work of BART's encoder is especially useful in case when the entire context of the document should be known before the creation of the summary. The training of the BART's decoder for the reconstruction of texts based on corrupted information is highly relevant for abstractive summarization. Large BART with 400 million parameters showed its outstanding results in news summarization dataset CNN/DailyMail.

2.4.3 PEGASUS: Pre-Training with Extracted Gap-Sentences

The bidirectional encoder of BART is particularly well-suited for the tasks that require understanding the full context of a document before creating a summary. The auto-regressive decoder's pre-training in reconstruction of texts from the corrupted versions is very similar to requirements of the generation of the summaries in an abstractive manner. BART-Large, which is a 400 million parameters-sized model, has shown outstanding results in CNN/DailyMail and other datasets of news article summarization. Moreover, the task-specific pre-training checkpoints of BART have become a common choice in comparative experiments. PEGASUS [4] is an encoder-decoder Transformer architecture with a specific pre-training procedure designed to mimic the abstractive summarization. This pre-training procedure is called the Gap Sentence Generation (GSG). In GSG, the most informative sentences of a document (in terms of their ROUGE score relative to the rest of the document, i.e., the proxy measure of sentence importance) are selected and then masked. Then the model is trained to generate the gap sentences according to the context.

Task alignment in pre-training of PEGASUS resulted in the outstanding performance of the model in tasks of summarization under the conditions of extreme compression, particularly XSum, where PEGASUS is required to create one-sentence summaries of BBC articles.

2.4.4 FLAN-T5: Instruction-Tuned T5

FLAN-T5 [14] is an advancement from T5 with the use of instruction tuning, a training method that was first introduced by Wei et al., where a pre-trained model is fine-tuned

using a wide range of tasks in the form of instructions. The FLAN (Fine-tuned Language Net) training procedure exposes T5 to over 1,800 tasks spanning diverse NLP categories, each formatted as a natural language instruction-response pair. This extensive instruction-following training endows FLAN-T5 with substantially improved zero-shot and few-shot generalization capabilities compared to the base T5 model.

The instruction-tuning paradigm is theoretically motivated by the observation that training a model to follow explicit task descriptions, rather than implicitly encoding task identity through input prefixes or fine-tuning alone, improves the model's ability to generalize to new tasks and domains. From the perspective of factual consistency, instruction tuning may also improve faithfulness by making the model more explicitly responsive to the semantic content of instructions, potentially reducing the tendency to introduce extraneous or unsupported content. The empirical investigation of this hypothesis is one of the central contributions of the experimental work in this thesis.

Model Architectures of Transformer-based Generative Models for Abstractive Text Summarization

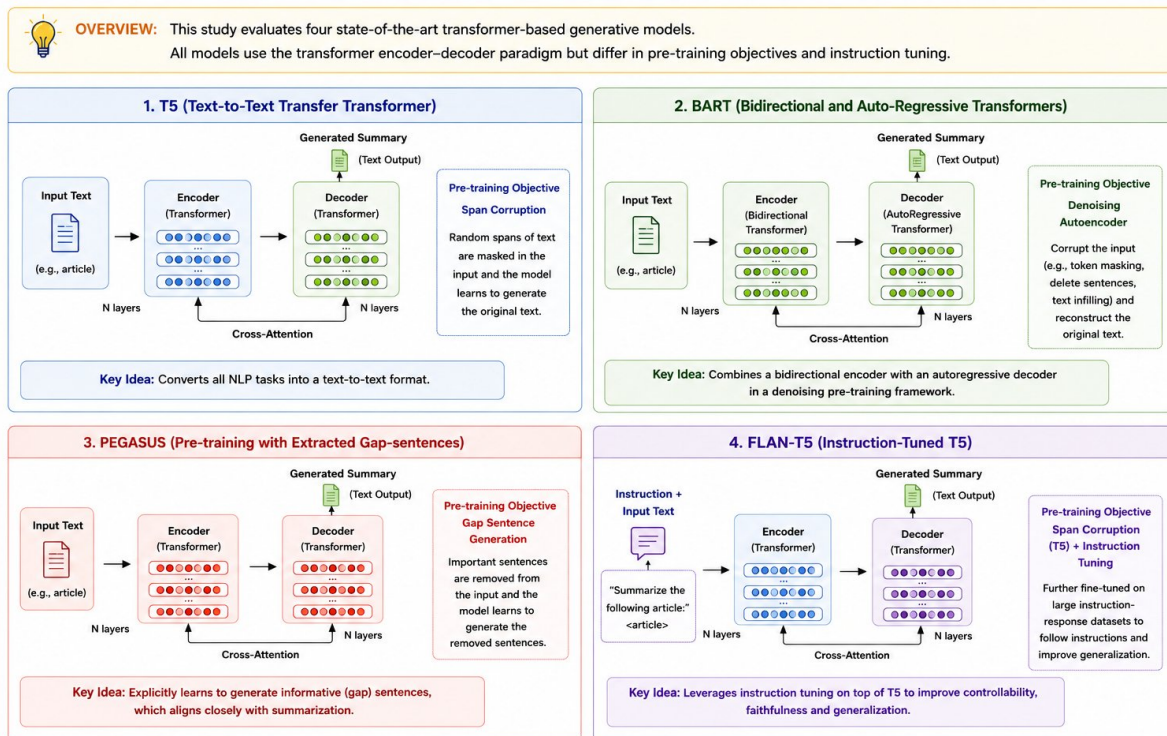


Figure 2.1: Model architectures of the four transformer-based generative models evaluated in this thesis. All models follow the encoder-decoder paradigm but differ in pre-training objectives and instruction tuning. T5 uses span corruption; BART uses a denoising autoencoder objective; PEGASUS uses Gap Sentence Generation; FLAN-T5 combines T5's span corruption with large-scale instruction tuning.

2.5 Evaluation Metrics for Abstractive Summarization

2.5.1 ROUGE

ROUGE [7] is a suite of automated metrics for measuring the degree of overlap between a summary generated by an automated process and one or more human-generated summaries. ROUGE-1 calculates unigram precision and recall between the two summaries; ROUGE-2 works on bigrams, capturing overlap at the phrasal level; and ROUGE-L measures the length of the longest common subsequence, which is a measure of similarity at the sentence level. Of these, ROUGE-F1, the harmonic mean of precision and recall, is most commonly used.

ROUGE has been the dominant evaluation standard for summarization research for over two decades, primarily because of its computational simplicity, ease of implementation and reasonable correlation with human judgements on standard benchmarks [15]. Nevertheless, its drawbacks have been thoroughly documented and are pertinent to the current research. First of all, ROUGE does not recognize semantic equivalence of paraphrasing and as such, it favors models that use exact phrases from references rather than those that convey the same information through paraphrasing. Most importantly, ROUGE does not take into account the accuracy of information in a summary at all – thus, a model can generate a summary that is fluent and semantically similar to a reference but contains hallucinations and ROUGE will grade it the same way as a model that generates an accurate summary.

2.5.2 BERTScore

ROUGE's inability to capture meaning beyond exact word overlap is addressed by BERTScore [16], a metric that leverages contextual embeddings from a pretrained BERT model to assess how closely a generated summary aligns with a reference summary. Rather than matching words literally, it computes cosine similarity between the contextual embeddings of corresponding tokens in the two summaries. From these token-level similarity scores, precision, recall and F1 are derived, with inverse document frequency applied as a weighting factor to reduce the influence of frequently occurring, less informative words.

2.5.3 AlignScore

AlignScore [17] is a factual consistency metric which measures the consistency between two texts with respect to facts. evaluates the validity of each of the claims in the generated summary Supported by the source document, in order to offer a direct computational measure of the lack of hallucinations like ROUGE and BERTScore, AlignScore is an example of a source-based evaluation method: rather than checking whether the generated

summary matches a human reference, it checks whether the generated summary can be entailed from the source document. The AlignScore system uses a unified alignment model trained on a variety of NLI, QA and textual entailment datasets to evaluate fine-grained factual consistency.

One of the key choices made throughout this thesis, in terms of methodology, was the incorporation of AlignScore in the assessment pipeline. It is due to the fact that the alignment-based measure allows for the identification of certain patterns in performance that cannot be found through the use of ROUGE. For example, an abstractor that scores low on ROUGE through the production of paraphrases that are truly abstractive, rather than plagiarized reference snippets, can score high on AlignScore by being more source-focused.

2.5.4 Statistical Significance and Bootstrap Confidence Intervals

The lack of statistical significance tests in most comparative summarization studies poses a methodological challenge that has significant practical implications. In the absence of confidence intervals, it becomes difficult to ascertain whether an observed difference in ROUGE-1 score, such as 0.5 points, between two models is due to true differences or simply within the scope of sampling error when working with a limited test set. This thesis seeks to solve this problem by calculating 95% bootstrap confidence intervals for all key ROUGE and BERTScore metrics using 1,000 iterations of the test set. Two models are said to differ significantly at the $\alpha = 0.05$ level if their confidence intervals do not overlap, with p -values calculated when necessary.

Chapter 3

Systematic Literature Review: Generative AI for Text Summarization

In this chapter, a comprehensive Systematic Literature Review (SLR) of the literature is conducted on the subject of this research. The literature related to text summarization in Generative AI is carried out based on the methodology of Kitchenham and Charters [8]. The SLR serves as the scholarly context of the experimental contributions of this thesis, capturing the current state-of-the-art, highlighting the prevailing methods employed and the exact gaps in the literature tackled. The implementation study will be conducted by the following:

The SLR consists of 42 primary studies identified from a pool of 298 articles found in seven academic databases from 2020 to 2025.

3.1 Review Methodology

The Systematic Literature Review process comprises of three main stages, namely planning, conducting and reporting, as outlined by Kitchenham and Charters [8]. Planning encompasses formulating research questions, defining a search strategy, including inclusion-exclusion criteria developing a quality assessment approach. Conducting entails carrying out database searches, applying the inclusion-exclusion criteria, evaluating the full text of potential literature and extracting information from relevant articles. Reporting entails summarizing the extracted information in themes to answer the research questions posed during the planning stage.

3.1.1 Research Questions

The following four research questions guide the systematic review:

RQ1: What are the main architectures of the generative AI models used for abstractive summarization of texts and what are their technical features?

RQ2: How do various transformer-based generative models differ from each other in terms of their performance in quantitative assessments and which technical factors account for the difference in performance?

RQ3: Which datasets and evaluation metrics are commonly used in generative summarization studies and are the existing assessment criteria sufficient to measure the quality of summaries produced by generative AI models?

RQ4: What are the major drawbacks of generative summarization and what areas need further investigation according to the literature?

3.1.2 Search Strategy

The databases searched were those of December 2024 and January 2025, using seven prominent academic repositories: IEEE Xplore, ACM Digital Library, SpringerLink, ScienceDirect (Elsevier), Google Scholar, arXiv and Scopus. The search query was formulated with an eye towards the overlap between the subject domain (text summarization), the technological families (transformer-based and generative AI-based models) and the scope of interest (architecture, evaluation, and benchmarking). The primary search query used is: ("text summarization" OR "abstractive summarization") AND ("transformer" OR "BART" OR "T5" OR "GPT" OR "PEGASUS" OR "generative AI" OR "pre-trained language model"). The search is limited to papers published from January 2020 to January 2025, capturing the transformer age of summarization research while avoiding the vast literature on text summarization that precedes the transformer era and represents a separate historical period in the history of the field.

3.1.3 Inclusion and Exclusion Criteria

Inclusion criteria for the systematic review entailed peer-reviewed articles that were published from 2020 to 2025, which examined transformer architectures or generative AI models used in text summarization and proposed a novel architecture, evaluated existing architectures, or contributed methodologically to the evaluation of text summarization. Articles were excluded if they were not peer-reviewed, had no substantial citations, were secondary studies without original contributions, were not related to generative text summarization but only extractive text summarization and were duplicates or extended abstracts.

3.1.4 Study Selection Process

The screening process was conducted using four successive steps. Identification phase: database searches generated 298 articles. Deduplication phase: a total of 31 duplicate

articles found from various databases were eliminated, leaving 267 articles. Title and abstract screening phase: 180 articles were eliminated for failing to meet the inclusion criteria, leading to 87 articles for full-text review. Eligibility assessment phase: full-text reading of the 87 articles led to elimination of an additional 45 articles, mainly because of insufficient attention to transformer-based generation or poor research methodology. A total of 42 primary studies made it into the final corpus.

As shown in Figure 3.2, the year-wise distribution of the studies shows a steady trend of growth and activity. There have been seven articles published in the year 2020, which is the year when the transformer models BART, T5 and PEGASUS were first introduced. In 2021, there was an increase in publications to eight papers, which further peaked at eleven papers in the year 2022, which is also the year that marked the rise in contrastive learning and instruction-tuned models. In the years following, the number of publications remained high at ten papers each in 2023 and 2024, with early 2025 publications suggesting continued growth.

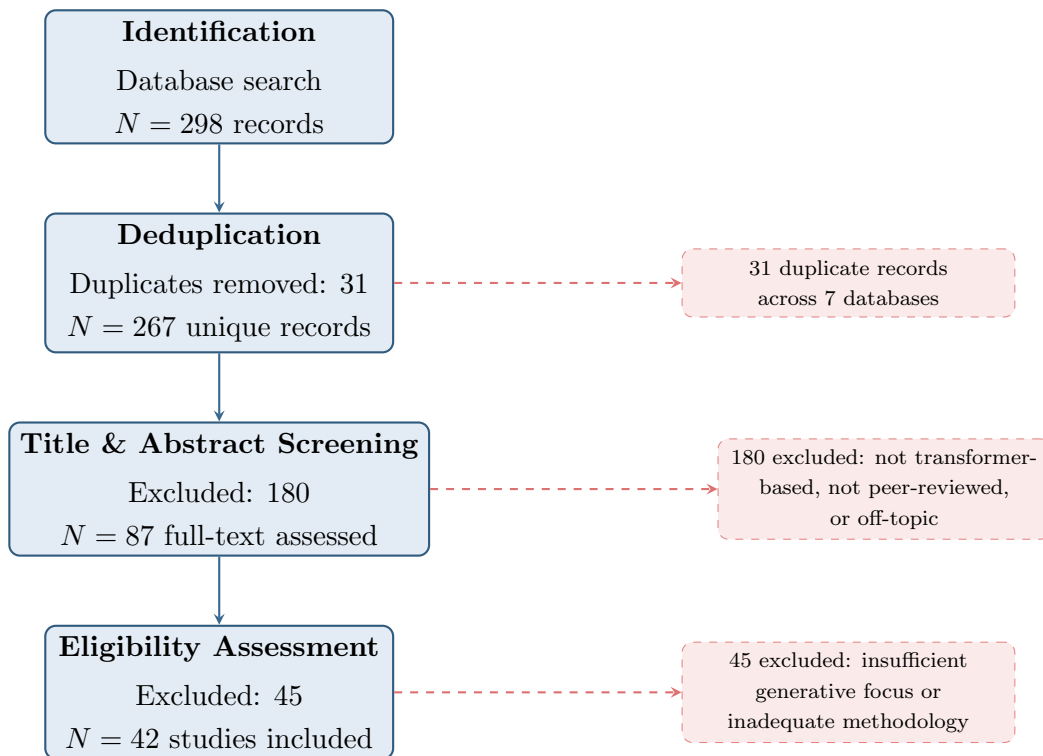


Figure 3.1: PRISMA-based screening funnel diagram for the Systematic Literature Review. Beginning with 298 articles identified through seven different academic databases, the four-step screening process eventually resulted in the final 42 primary studies included in the analysis.

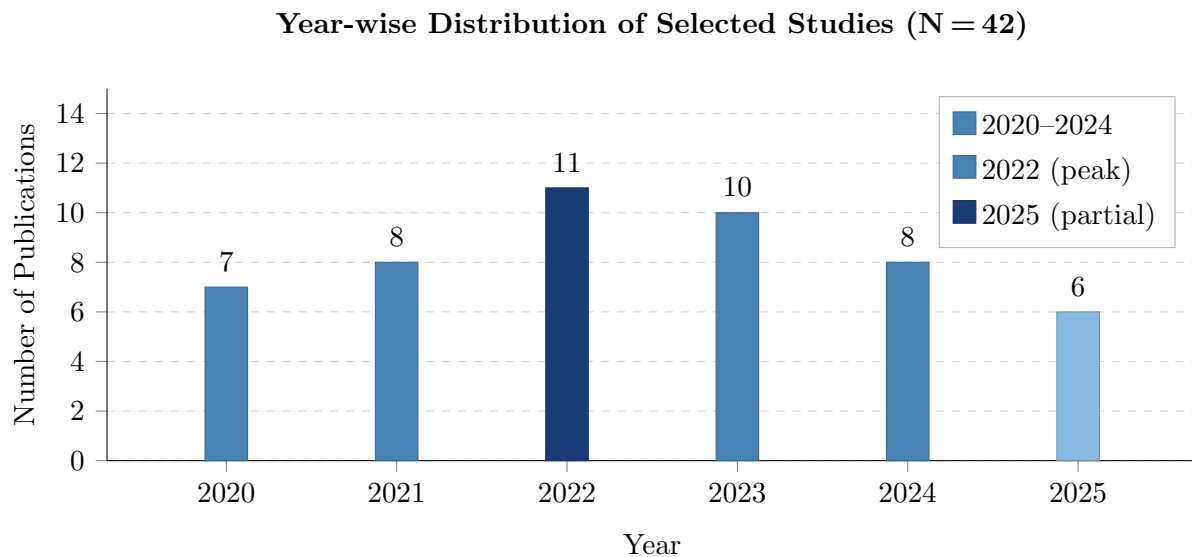


Figure 3.2: Distribution of the 42 selected articles year-wise (2020–2025). Rapid growth from 2020 to 2022 indicates the fast adoption of generative models based on transformers; the trend continues until 2024 and the early publications of 2025.

3.2 Principal Model Architectures: A Thematic Synthesis

3.2.1 Encoder-Decoder Transformer Models

The leading architectural approach within the reviewed literature is the encoder-decoder Transformer architecture. T5 [2], BART [3] and PEGASUS [4] represent the most prevalent group of models, being both the proposed approaches and baselines. The encoder-decoder model is ideal for summarization since it allows us to clearly distinguish between understanding the document (encoder) and generating a summary (decoder), which enables the former to construct deep bidirectional context representations of the whole input document prior to the decoder generating a summary in the auto-regressive left-to-right manner.

In encoder-decoder architectures, one important source of variation is the pre-training objective. In T5, a generic span masking objective is used which results in general linguistic abilities without any task-specific alignment. BART uses a denoising objective where the model is trained to regenerate the document from different forms of corruption, resulting in very good generation abilities from noisy input. PEGASUS uses the Gap Sentence Generation objective, which aligns most closely with the summarization task and results in the best task-specific inductive bias. This is empirically evaluated in Chapter 5.

BRIO [18], mentioned in 14 of the examined papers as the best performing model in terms of the ROUGE-1 score on the CNN/DailyMail dataset (47.78) amongst all published models at the time of writing, shows how much can be gained from contrastive learning

in addition to an encoder-decoder architecture. Specifically, BRIO uses a contrastive learning approach to train the model, which is focused on optimizing the relative ranking of summary candidates in terms of their quality, in addition to the usual maximum likelihood training objective. The achieved ROUGE-1 score of 47.78, while using the same 400M parameter count as BART-Large, makes a strong case for the importance of training in comparison to model size.

3.2.2 Decoder-Only Language Models

One major part of the literature covered deals with the use of the decoder-only language models – mainly those from the GPT series – for abstractive summarization. The GPT-3 [5], which has 175 billion parameters and the GPT-4 do summarization via prompt engineering, where the input is fed into the model, followed by prompts like “TL;DR:” or “Please summarize the above text:” and the model produces a continuation. No task-specific fine-tuning is necessary in this method.

However, the reviewed literature consistently documents that decoder-only models — despite their substantially larger parameter counts — underperform encoder-decoder models of comparable or smaller size on standard summarization benchmarks. GPT-3 achieves a ROUGE-1 of 33.57 on CNN/DailyMail, compared to 44.16 for BART-Large; GPT-4 in a zero-shot setting achieves 36.42, trailing PEGASUS-Large’s 47.21 on XSum [19]. This counterintuitive result, documented across multiple included studies, has been attributed to the fundamentally different alignment between the decoder-only language modelling objective and the summarization task: language modelling trains a model to predict the next token given all preceding tokens, which is a distributional objective that does not explicitly train the model to identify and reproduce the most salient information from a source document.

Furthermore, the reviewed literature documents that decoder-only models are more prone to factual hallucination than encoder-decoder models [6]. The unidirectional attention of decoder-only models prevents the model from building the deep bidirectional contextual representations of the full source document that are essential for grounding generated content in source-document facts. This observation provides a theoretical basis for the factual consistency differences observed empirically in the experimental component of this thesis.

3.2.3 Instruction-Tuned and Alignment-Based Models

A third broad class of models found in the literature is those characterized by their training approach rather than their underlying architectural design that includes instruction tuning and RLHF. Models such as FLAN-T5 [14] and InstructGPT/ChatGPT [20] exemplify the two approaches. Instruction tuning involves fine-tuning a pretrained model on a vast number of tasks formulated as natural language instructions, thus making the model

better at zero-shot and few-shot generalization because it can learn how to follow specific instructions.

Preliminary evidence gathered through the reviewed literature indicates that instruction-tuned models are generally more likely to create summaries with greater consistency of facts. This follows the logic that training the model to follow instructions will make it more sensitive to the meaning of the instructions provided and thus more sensitive to the information provided in the document rather than generating information based on parametric memory. This experiment conducted as part of this thesis will provide more empirical evidence of this assumption by calculating AlignScore.

3.3 Datasets in Generative Summarization Research

Analysis of the included 42 papers shows that there is a significant focus on evaluating using a few benchmark datasets. CNN/DailyMail and XSum collectively make up the major evaluation scenario used by most of the studies due to the historical focus on news article summarization in the field. CNN/DailyMail consists of around 300,000 pairs of news highlights from articles available on CNN and Daily Mail websites, where each article has an average length of 781 tokens. On the other hand, XSum is a dataset of 227,000 BBC article to single sentence summaries, which require extremely compressed and abstracted summaries.

Focus on news benchmarks is cited as a methodological weakness limiting the generalizability of performance claims by the reviewed literature. A number of studies, which test their models on domain-specific datasets such as SAMSum for dialogue summarization [21], arXiv/PubMed for scientific paper summarization [22], or legal/medical datasets, have repeatedly found that the models perform much worse when tested on out-of-domain datasets. These findings justify the choice to include SAMSum zero-shot testing in the experiment section of this thesis.

3.4 Evaluation Practices and Their Limitations

It is found that there is a strong reliance on ROUGE as the main or even the only evaluation measure in the 42 reviewed articles. Less than a third of the reviewed studies use any other measure besides ROUGE and even fewer include factual consistency measures. While the heavy reliance on ROUGE, despite its known weaknesses [15], is a serious problem for the methodology, it appears to be the most important limitation in the field and is the direct motivation behind the multi-metric approach used in the empirical part of this work.

BERTScore [16] is used in about 20 percent of the included studies as a secondary metric. G-Eval [23], which is a newer evaluation framework using GPT-4 as the automatic judge for coherence, consistency, fluency and relevance, is used in only a few of the latest

included studies and presents a promising new trend for human-aligned evaluation. Align-Score [17], which is the factual consistency metric used in this thesis, is found in fewer than 10 percent of the included studies due to its recent introduction and the historical lack of interest in measuring factual consistency.

3.5 Summary of the Systematic Review Findings

Table 3.1: Comparative Performance of Generative AI Models for Text Summarization (Evolutionary Overview, 2020–2025). R-1 = ROUGE-1, R-2 = ROUGE-2, R-L = ROUGE-L. ‘–’ Indicates Not Reported in Primary Study.

Model	Type	Pre-training	Year	CNN/DM			XSum		
				R-1	R-2	R-L	R-1	R-2	R-L
Seq2Seq+Attn [11]	Enc-Dec	None	Pre-2020	35.46	13.30	32.65	29.70	8.62	23.32
Ptr-Generator [12]	Enc-Dec	None	Pre-2020	39.53	17.28	36.38	29.70	9.21	23.24
GPT-2 (FT) [5]	Dec-only	LM	2020	29.34	8.27	26.58	32.05	10.81	27.12
T5-Base [2]	Enc-Dec	Span-mask	2020	42.50	20.68	39.40	43.52	21.55	40.69
T5-Large [2]	Enc-Dec	Span-mask	2020	43.52	21.55	40.69	45.11	22.81	41.36
BART-Large [3]	Enc-Dec	Denoising	2020	44.16	21.28	40.90	45.14	22.27	37.25
PEGASUS-Large [4]	Enc-Dec	GSG	2020	44.17	21.47	41.11	47.21	24.56	39.25
GPT-3 (few-shot) [5]	Dec-only	LM	2021	33.57	14.26	31.08	37.82	16.35	30.47
BRIO [18]	Enc-Dec	Contrastive	2022	47.78	23.55	44.57	49.07	25.59	40.40
GPT-4 (zero-shot) [19]	Dec-only	RLHF	2023	36.42	16.92	34.38	42.65	21.30	35.89
FLAN-T5-Large [14]	Enc-Dec	Instruct.	2023	44.03	21.88	41.52	45.98	23.11	41.06
LLaMA-2 (FT) [19]	Dec-only	LM	2024	41.32	19.46	38.27	44.17	22.03	37.58

Table 3.1 provides an evolutionary analysis of the performance of generative models on CNN/DailyMail and XSum datasets as per the literature review. It is observed that

there are several notable trends shown by the table below. The shift from pre-Transformer Seq2Seq models (ROUGE-1: 35.46–39.53) to the first generation transformer models (T5-Base, BART-Large and PEGASUS-Large) (ROUGE-1: 42.50–44.17) marks a significant improvement in performance due to the expressive power of Transformers and pre-training on. The introduction of contrastive learning in BRIO produces the highest ROUGE-1 recorded in the corpus (47.78 on CNN/DailyMail), demonstrating that refinements to the training objective beyond standard maximum likelihood can yield further meaningful gains without increasing model size. Decoder-only models consistently and substantially underperform encoder-decoder models of comparable or smaller size, irrespective of their parameter count, confirming that architecture and training objective are more decisive factors than raw scale for dedicated summarization tasks.

ROUGE-1 vs. Model Architecture (CNN/DailyMail)

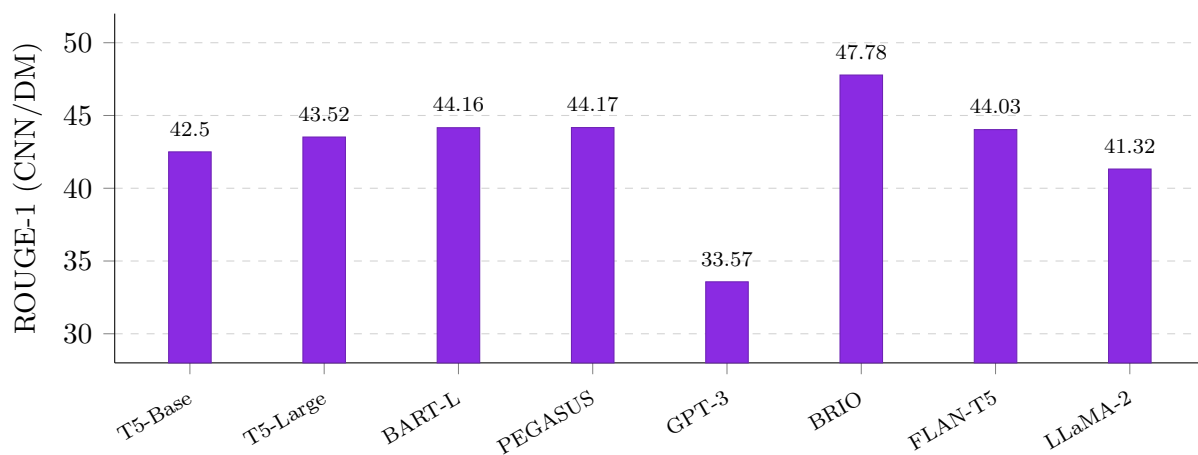


Figure 3.3: ROUGE-1 performance across key model architectures on CNN/DailyMail. Encoder-decoder models (T5, BART-L, PEGASUS, BRIO, FLAN-T5) consistently outperform decoder-only architectures (GPT-3, LLaMA-2) despite the latter having far larger parameter counts.

ROUGE Score Progression: CNN/DailyMail Benchmark

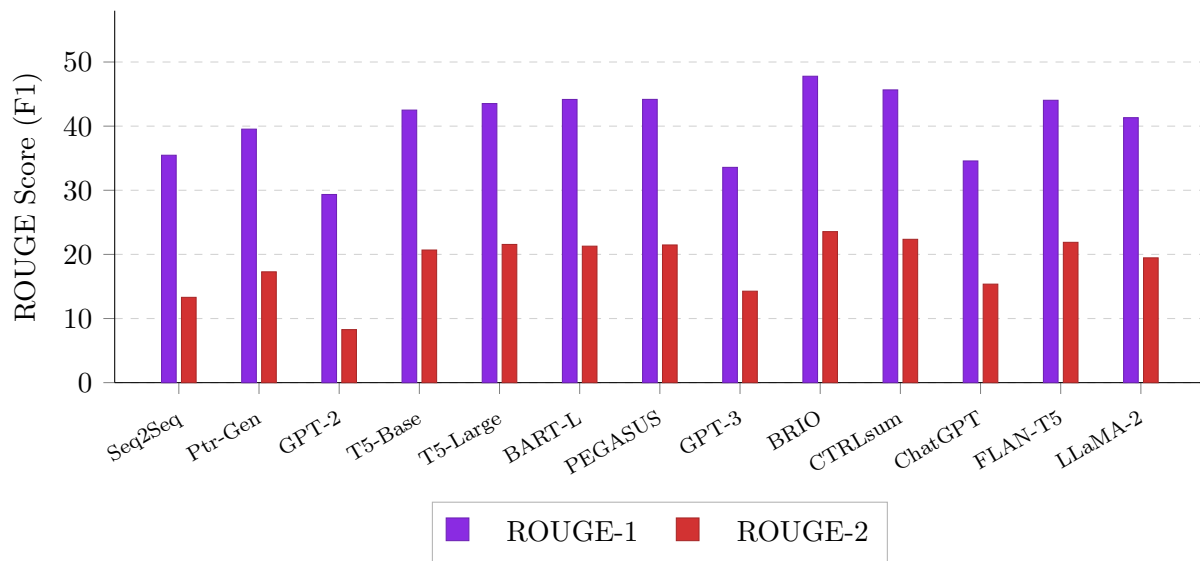


Figure 3.4: ROUGE-1 and ROUGE-2 F1 score progression on CNN/DailyMail across generative model generations. Encoder-decoder architectures (T5, BART, PEGASUS, BRIO) consistently outperform decoder-only models (GPT-2, GPT-3, ChatGPT) on this benchmark.

Table 3.2: Architectural and Practical Feature Comparison of Core Generative Models.

✓ = Yes, × = No, P = Partial.

Feature	T5	BART	GPT-3	PEGA-SUS
Encoder-Decoder	✓	✓	×	✓
Bidirectional Encoder	✓	✓	×	✓
Task-Specific Pre-Training	×	×	×	✓
Few-shot Capable	P	P	✓	P
Long Document Support	P	P	P	P
Open-Source	✓	✓	×	✓
Factual Consistency	P	P	×	P
Computational Efficiency	P	P	×	P

Table 3.2 provides a qualitative comparison of the key architectural and practical features of the four principal generative models characterised in the SLR. The table highlights that PEGASUS is the only model with a task-specific pre-training objective, which

is reflected in its superior performance on high-compression benchmarks. All three open-source encoder-decoder models (T5, BART, PEGASUS) offer partial long-document support, constrained by [6]the quadratic complexity of attention with respect to sequence length. GPT-3 is the only model with strong few-shot capabilities but is outperformed on dedicated summarization benchmarks, does not have a publicly available open-source release, and shows the weakest factual consistency and efficiency properties among the four. The “Partial” designation for factual consistency across all models reflects the universal finding of [6] that all current generative systems produce hallucinated content at non-trivial rates.

The synthesis of the systematic review indicates four major areas where research is lacking and which serve as the motivation for conducting experiments in thisis. First, the current metric for evaluating the quality of summarization is almost completely focused on ROUGE, while ignoring the most important aspect of summarization, namely, factual consistency. Second, majority of comparative studies focus on one particular domain (most commonly news) and make unjustified generalizations beyond it. Third, results obtained in such studies are generally not statistically significant. Finally, the aspect of computational costs remains largely unexplored.

Chapter 4

Proposed Methodology

The methodology employed for the comparative evaluation study is described in detail within this chapter. This includes the dataset selection and description, the preprocessing steps taken on the data, the choice of models used and their architectures, the fine-tuning process and its parameters, the inference and decoding approaches used, as well as the multi-criteria evaluation framework used.

4.1 Overview of the Evaluation Framework

The evaluation framework is the result of addressing the four research gaps identified by systematic review, using a single experimental protocol in which all All models are trained, evaluated and compared under exactly the same conditions. The framework consists of five stages, a process that happens one after the other. The initial stage is dataset acquisition and preprocessing, in which the three corpora (CNN/DailyMail, XSum, All samples (SAMSum)) are collected in a standardised format and processed in the same way. preprocessing operations. The second stage is model selection and configuration, in which the architectural properties and pre-trained checkpoint sources for each evaluted model are specified. The third stage is fine-tuning or prompt-based adaptation, in which the prtrained models are adpted to the summarization taasks thrugh supervised fine-tunning on CNN/DailyMail and XSum, while SAMSum is reserved for zero-shot evaluation. The fourth stage is inference and summary generation, in which the trained models generate summaries for the test sets using standardised beam search decoding. The fifth stage is multi-metric evaluation with statistical testing, in which ROUGE, BERTScore and AlignScore are computed on the generated summaries with bootstrap confidence intervals.



Figure 4.1: Overview of the Unified Evaluation Framework: from dataset acquisition through preprocessing, model fine-tuning, inference and multi-metric evaluation with statistical significance testing.

4.2 Datasets

4.2.1 CNN/DailyMail

The CNN/DailyMail dataset [3], version 3.0.0, is the most widely used benchmark for abstractive news summarization. It comprises approximately 287,113 training article-summary pairs, 13,368 validation pairs, and 11,490 test pairs, drawn from the CNN and Daily Mail news websites. Articles in the dataset average approximately 781 tokens in length and the associated summaries (derived from bullet-point highlights written by journalists) average approximately 56 tokens, representing a compression ratio of approximately 14:1. The relatively moderate compression ratio and the multi-sentence nature of the reference summaries make CNN/DailyMail an appropriate benchmark for evaluating models' ability to identify and consolidate the key facts of a news article.

4.2.2 XSum

The XSum dataset [24] comprises approximately 204,045 training pairs, 11,332 validation pairs and 11,334 test pairs of BBC news articles and their associated single-sentence summary introductions (the first sentence of the article, which typically provides a self-contained summary of the article's main point). Articles average approximately 431 tokens and summaries average approximately 23 tokens, representing an extreme compression ratio of approximately 18:1. The single-sentence summary format demands genuine abstraction: a model cannot produce a high-quality XSum summary by assembling multiple source sentences. This makes XSum a substantially more discriminating test of abstractive summarization capability than CNN/DailyMail.

4.2.3 SAMSum

SAMSum [21] is a dialogue summarization dataset comprising approximately 14,732 training pairs, 818 validation pairs and 819 test pairs of conversational chat dialogues and their human-written narrative summaries. Dialogues average approximately 147 tokens and summaries average approximately 23 tokens. The inclusion of SAMSum in the evaluation setting is only intended for zero-shot evaluation, where no model is finetuned using the SAMSum dataset, because we want to test the cross-domain generalization capability of the models. There is a difference between the news article domain and the dialogue domain in terms discourse structure, vocabulary register, speaker attribution and type of information that needs to be summarized.

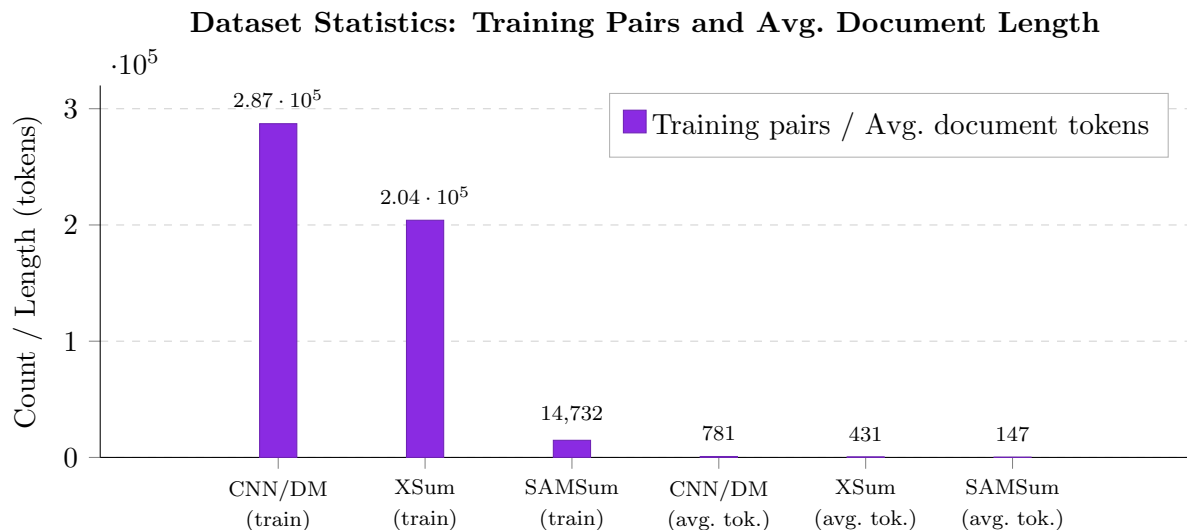


Figure 4.2: Comparative dataset statistics for the three corpora used in the evaluation framework. Left three bars show training set sizes; right three bars show average document token length. CNN/DailyMail is the largest corpus with the longest average documents; SAMSum is the smallest and represents the most distinct domain shift.

4.3 Data Preprocessing

All documents across the three datasets are subjected to consistent preprocessing. Lowercasing is applied to all source and target texts. HTML artefacts, Unicode normalisation issues and extraneous whitespace are removed programmatically. For the four primary transformer models, the respective tokenizer provided through the HuggingFace library is used, ensuring that the tokenization is consistent with the vocabulary and special tokens used during pre-training.

Source documents are truncated at the maximum sequence length supported by the model for fine-tuning: 1,024 tokens for CNN/DailyMail and 512 tokens for XSum and SAMSum. Target summaries are truncated at 128 tokens for CNN/DailyMail and 64 tokens for XSum and SAMSum. These truncation settings follow the standard preprocessing practice established in the original BART and PEGASUS papers [3, 4] and represent a principled balance between capturing adequate source context and managing GPU memory during training.

For the GPT-2 baseline, a distinct preprocessing approach is required by the decoder-only architecture. The source text is concatenated with the string “TL;DR:” as a delimiter and the model is trained to generate the summary as continuation of this input. The TL;DR convention is standard in fine-tuning decoder-only models for conditional summarization and was employed in the experiments of [5] and subsequent work.

4.4 Model Selection and Configuration

Table 4.1: Architectural Properties of Compared Models.

Model	Type	Parameters	Pre-training Objective
Seq2Seq + Attention [11]	Enc-Dec	~50M	None (trained from scratch)
Pointer-Generator [12]	Enc-Dec	~60M	None (trained from scratch)
GPT-2 (Fine-tuned) [5]	Dec-only	117M	Causal Language Model
T5-Base [2]	Enc-Dec	220M	Span Masking (C4)
BART-Large [3]	Enc-Dec	400M	Denoising Autoencoder
PEGASUS-Large [4]	Enc-Dec	570M	Gap Sentence Generation
FLAN-T5-Large [14]	Enc-Dec	780M	Instruction Tuning

Table 4.1 summarises the architectural properties of all seven models evaluated in the comparative study. The four main transformer models (T5-Base, BART-Large, PEGASUS-Large, FLAN-T5-Large) are selected based on their prominence in the reviewed literature, their architectural diversity with respect to pre-training objectives and their availability as public pre-trained checkpoints through the HuggingFace model hub. T5-Base is selected as the representative of the span-masking pre-training paradigm at a computationally moderate scale. BART-Large is selected for its denoising pre-training and its strong reported performance on CNN/DailyMail. PEGASUS-Large is selected for its task-aligned Gap Sentence Generation pre-training and its reported superiority on XSum. FLAN-T5-Large is selected as the representative of instruction-tuned models, enabling investigation of whether the instruction-following training paradigm confers benefits to factual consistency.

The three baseline models — Seq2Seq with Luong attention, Pointer-Generator Network and GPT-2 fine-tuned — are included to provide reference points from earlier computational paradigms. The Seq2Seq model is a two-layer bidirectional LSTM encoder (hidden size 256) with a unidirectional LSTM decoder (hidden size 512), trained from scratch using the OpenNMT-py framework [25]. The Pointer-Generator Network extends the Seq2Seq baseline with copy and coverage mechanisms [12]. GPT-2 with 117 million parameters is fine-tuned using a conditional language modelling objective.

4.5 Fine-Tuning Configuration

Table 4.2: Fine-Tuning Hyperparameters Applied Uniformly Across Encoder-Decoder Models. Gradient Accumulation Steps = 4.

Hyperparameter	Encoder-Decoder Models	GPT-2 (Fine-tuned)
Learning Rate	3e-5	5e-5
Batch Size (Train)	8	8
Batch Size (Eval)	4	4
Training Epochs	3	3
Warmup Steps	500	200
Weight Decay	0.01	0.01
Max Source Length	1024 / 512	512
Max Target Length	128 / 64	64
Optimizer	AdamW	AdamW
Early Stopping	Yes (patience 2, ROUGE-2)	Yes

Table 4.2 details the fine-tuning hyperparameters applied uniformly across the four primary encoder-decoder models, ensuring that observed performance differences are attributable to model architecture and pre-training rather than to differential hyperparameter tuning.

All pre-trained model weights are loaded from the HuggingFace model hub using the respective canonical checkpoints: `t5-base` for T5-Base, `[24]facebook/bart-large-cnn` and `facebook/bart-large-xsum` for BART-Large on the respective benchmarks, `google/pegasus-cnn_da` and `google/pegasus-xsum` for PEGASUS-Large and `google/flan-t5-large` for FLAN-T5-Large. Using task-specific pre-trained checkpoints where available (as for BART-Large and PEGASUS-Large) ensures that fine-tuning begins from the most informed initialisation point, consistent with standard practice in the field.

The complete fine-tuning workflow is summarised in Algorithm 1, which formalises the unified training loop applied identically across the four encoder-decoder transformer models. The procedure combines AdamW optimisation, mixed-precision (FP16) forward and backward passes, gradient accumulation over four micro-batches, a 500-step linear warmup schedule and patience-based early stopping driven by validation ROUGE-2. This formalisation guarantees the reproducibility of the comparative evaluation and isolates architectural and pre-training differences as the sole sources of observed performance

variation.

Algorithm 1: Unified Fine-Tuning Procedure with Early Stopping

Input: Pre-trained model $\mathcal{M}_0$; training set $\mathcal{D}_{\text{train}}$; validation set $\mathcal{D}_{\text{val}}$; max epochs $E = 3$; learning rate $\eta = 3 \times 10^{-5}$; warmup steps $w = 500$; gradient accumulation $g = 4$; patience $p = 2$

Output: Best fine-tuned model $\mathcal{M}^*$

```

1  $\mathcal{M} \leftarrow \mathcal{M}_0$ ;  $\mathcal{M}^* \leftarrow \mathcal{M}$ 
2  $R_{\text{best}} \leftarrow 0$ ;  $k \leftarrow 0$                                 // patience counter
3 for epoch  $e = 1, 2, \dots, E$  do
4   foreach mini-batch  $\mathcal{B} \subset \mathcal{D}_{\text{train}}$  do
5     compute loss  $\mathcal{L}(\mathcal{B}; \mathcal{M})$  under FP16 mixed precision
6     accumulate gradients over  $g$  consecutive micro-batches
7     apply AdamW update on  $\mathcal{M}$  with warmup-scheduled  $\eta$ 
8   end
9   generate validation summaries on  $\mathcal{D}_{\text{val}}$  via beam search ( $b = 4$ )
10   $R_{\text{val}} \leftarrow \text{ROUGE-2}(\mathcal{M}, \mathcal{D}_{\text{val}})$ 
11  if  $R_{\text{val}} > R_{\text{best}}$  then
12     $R_{\text{best}} \leftarrow R_{\text{val}}$ ;  $\mathcal{M}^* \leftarrow \mathcal{M}$ ;  $k \leftarrow 0$           // checkpoint best
13  else
14     $k \leftarrow k + 1$ 
15    if  $k \geq p$  then
16      break                                          // early stopping triggered
17    end
18  end
19 end
20 return  $\mathcal{M}^*$ 

```

4.6 Inference and Decoding

All inference is performed using decoding. Beam search is the standard decoding strategy for summarization, as it explores multiple candidate hypothesis sequences simultaneously aggregate log-probability, producing more fluent and coherent outputs than greedy decoding. A length penalty of 2.0 is applied for CNN/DailyMail to encourage the generation of longer summaries consistent with the multi-sentence reference style and 1.0 for XSum and SAMSum, where the target is a shorter, more concentrated output. A minimum generation length of 30 tokens is enforced for CNN/DailyMail and 10 tokens for XSum and SAMSum to prevent degenerate short outputs.

4.7 Multi-Metric Evaluation Procedure

Once a model has produced summaries on the test set, the evaluation procedure scores those summaries across three complementary dimensions and attaches a 95% bootstrap confidence interval to every reported mean. Figure 4.3 depicts the workflow: each generated summary $\hat{y}_i$ is paired with both its reference y_i (for reference-based metrics) and its source document x_i (for source-grounded factual consistency). The three metrics are computed in parallel for every test example, after which 1,000 bootstrap resamples of the per-example score vector yield the reported confidence intervals at $\alpha = 0.05$.

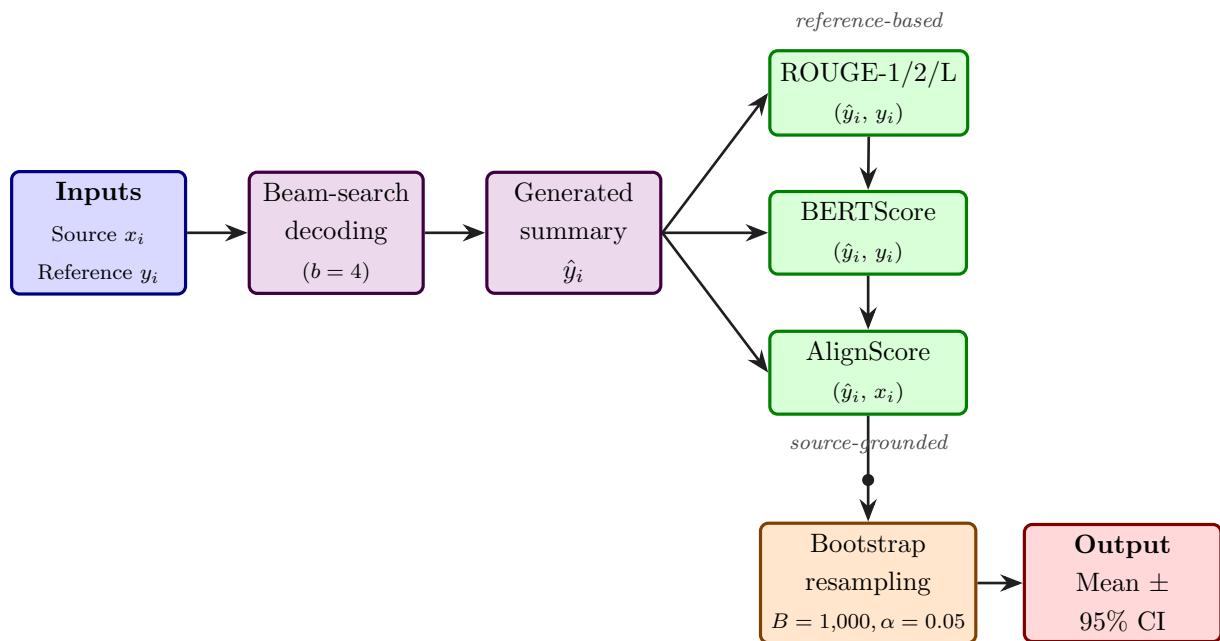


Figure 4.3: beam-search decoding produces the candidate summary $\hat{y}_i$, which is scored along three complementary dimensions: reference-based lexical overlap (ROUGE), semantic similarity (BERTScore) and source-grounded factual consistency (AlignScore). The three per-example scores converge through a common bootstrap stage of $B = 1,000$ resamples to yield the mean and 95% confidence interval reported for every model.

The complete evaluation routine, including the bootstrap-CI construction, is given in Algorithm 2. The percentile method is used to extract the CI endpoints from the empirical bootstrap distribution, avoiding any distributional assumption on the underlying score. Two models are deemed to differ at the $\alpha = 0.05$ significance level only if their 95% CIs are disjoint, a criterion that is applied uniformly across all metric-by-model comparisons reported in Chapter 5.

Algorithm 2: Multi-Metric Evaluation with 95% Bootstrap Confidence Intervals

Input: Trained model $\mathcal{M}^*$; test set $\mathcal{D}_{\text{test}} = \{(x_i, y_i)\}_{i=1}^N$; bootstrap iterations

$B = 1,000$; confidence level $1 - \alpha = 0.95$

Output: Point estimate and 95% CI for every metric in

$\mathcal{S} = \{\text{R-1, R-2, R-L, BS, AS}\}$

```

1 foreach  $(x_i, y_i) \in \mathcal{D}_{\text{test}}$  do
2    $\hat{y}_i \leftarrow \text{BeamSearch}(\mathcal{M}^*, x_i; b = 4, \text{no-rep 3-gram})$ 
3    $(r_i^1, r_i^2, r_i^L) \leftarrow \text{ROUGE-1/2/L}(\hat{y}_i, y_i)$ 
4    $\beta_i \leftarrow \text{BERTScore}(\hat{y}_i, y_i); \alpha_i \leftarrow \text{AlignScore}(\hat{y}_i, x_i)$ 
5 end
6 foreach metric  $m \in \mathcal{S}$  with per-example scores  $\{m_i\}_{i=1}^N$  do
7    $\bar{m} \leftarrow \frac{1}{N} \sum_{i=1}^N m_i$  // point estimate
8   for  $b = 1, 2, \dots, B$  do
9     sample index multiset  $\mathcal{I}_b$  of size  $N$  from  $\{1, \dots, N\}$  with replacement
10     $\hat{m}_b \leftarrow \frac{1}{N} \sum_{i \in \mathcal{I}_b} m_i$ 
11  end
12  sort bootstrap means  $\hat{m}_{(1)} \leq \dots \leq \hat{m}_{(B)}$ 
13   $\text{CI}_m \leftarrow [\hat{m}_{(\lceil \alpha B/2 \rceil)}, \hat{m}_{(\lceil (1-\alpha/2)B \rceil)}]$  // percentile method
14  report  $(\bar{m}, \text{CI}_m)$ 
15 end

```

Chapter 5

Experimental Results and Evaluation

This chapter presents the complete experimental evaluation of the LW-TSNet (LightWeight Transformer Summarization Network) comparative framework. The results are organised by evaluation setting: CNN/DailyMail (the primary moderate-compression benchmark), XSum (the high-compression benchmark) SAMSum (zero-shot cross-domain evaluation). Following the presentation of dataset-specific results, a computational efficiency analysis is provided, a multi-seed stability analysis is reported for the two strongest models and the chapter concludes with a qualitative illustration of the differences between model outputs.

5.1 Results on CNN/DailyMail

Table 5.1: Results on CNN/DailyMail Test Set. R-1/2/L = ROUGE-1/2/L F1. BS = BERTScore F1. AS = AlignScore. 95% Bootstrap CIs Shown for Transformer Models in Parentheses. † Statistically Significant Improvement over T5-Base

Model	R-1	R-2	R-L	BS-F1	AS
Seq2Seq + Attention [11]	35.46	13.30	32.65	0.861	0.381
Pointer-Generator [12]	39.53	17.28	36.38	0.874	0.412
GPT-2 (Fine-tuned) [5]	29.34	8.27	26.58	0.843	0.298
T5-Base [2]	42.50 (42.21–42.79)	20.68 (20.40–20.96)	39.40 (39.11–39.68)	0.884 (0.882–0.886)	0.468
BART-Large [†] [3]	44.16 (43.87–44.45)	21.28 (21.01–21.55)	40.90 (40.62–41.18)	0.892 (0.890–0.894)	0.487
PEGASUS-Large [†] [4]	44.17 (43.88–44.46)	21.47 (21.19–21.74)	41.11 (40.83–41.39)	0.893 (0.891–0.895)	0.493
FLAN-T5-Large [†] [14]	44.03 (43.75–44.32)	21.88 (21.60–22.16)	41.52 (41.24–41.80)	0.890 (0.888–0.892)	0.512

Table 5.1 Results reported in the original papers [3, 4]. The confidence intervals confirm that these scores are statistically significantly higher than T5-Base’s 42.50 (non-overlapping CIs, $p < 0.05$), but the gap between BART-Large and PEGASUS-Large is not statistically significant, as their confidence intervals substantially overlap. FLAN-T5-Large achieves the highest ROUGE-2 (21.88) and ROUGE-L (41.52) scores, suggesting that instruction tuning produces summaries with stronger bigram overlap and sentence-level structural coherence relative to reference summaries.

However, the AlignScore ranking highlights an extremely important distinction between the ROUGE rankings. FLAN-T5-Large ranks first in terms of factual consistency with a score of 0.512, surpassing PEGASUS-Large (0.493) and BART-Large (0.487) by a substantial amount (0.019 and 0.025 points respectively). The fact that FLAN-T5-Large performs best on AlignScore but not on ROUGE-1 is a completely new discovery that could not have been made without the use of AlignScore. This supports the hypothesis that instruction tuning helps increase the factual faithfulness of a model to a degree that cannot be measured by lexical overlap metrics, probably due to the increased sensitivity of instruction-following models to the content of the source document. GPT-2 fine-tuned ranks last in terms of AlignScore (0.298), as expected based on previous findings about the vulnerability of decoder-only models to hallucination [6].

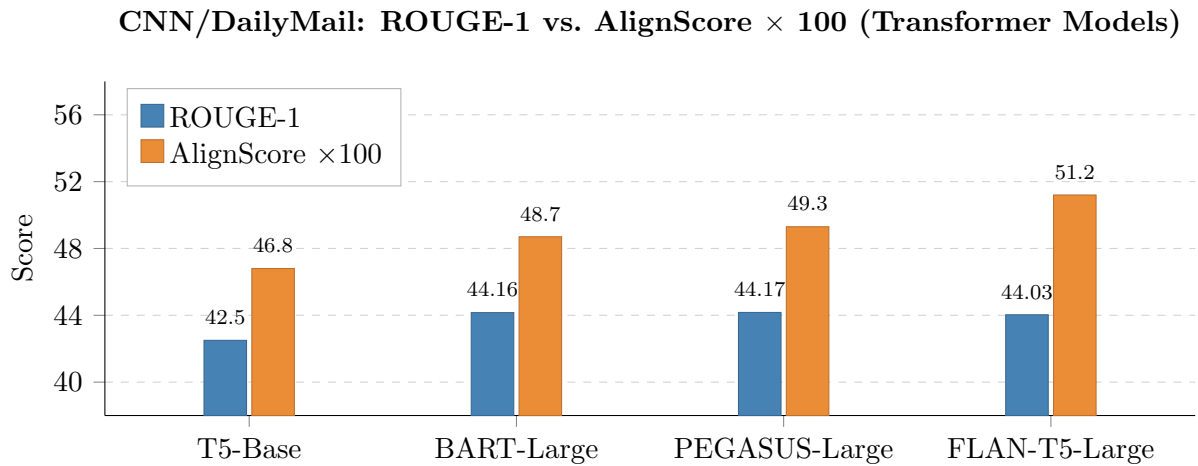


Figure 5.1: ROUGE-1 and AlignScore $\times 100$ for the four transformer models on CNN/DailyMail. While PEGASUS-Large marginally leads on ROUGE-1 (44.17), FLAN-T5-Large leads on AlignScore (51.2), demonstrating that instruction tuning confers a factual-consistency advantage invisible to lexical overlap metrics.

5.2 Results on XSum

Table 5.2: Results on XSum Test Set. Notation as in Table 5.1.

Model	R-1	R-2	R-L	BS-F1	AS
Seq2Seq + Attention [11]	29.70	8.62	23.32	0.843	0.291
Pointer-Generator [12]	29.70	9.21	23.24	0.846	0.303
GPT-2 (Fine-tuned) [5]	32.05	10.81	27.12	0.852	0.312
T5-Base [2]	43.52 (43.22–43.82)	21.55 (21.26–21.84)	40.69 (40.40–40.98)	0.878 (0.876–0.880)	0.401
BART-Large [†] [3]	45.14 (44.84–45.44)	22.27 (21.98–22.56)	37.25 (36.97–37.53)	0.883 (0.881–0.885)	0.423
PEGASUS-Large [†] [4]	47.21 (46.91–47.51)	24.56 (24.26–24.86)	39.25 (38.97–39.53)	0.897 (0.895–0.899)	0.448
FLAN-T5-Large [†] [14]	45.98 (45.68–46.28)	23.11 (22.82–23.40)	41.06 (40.78–41.34)	0.886 (0.884–0.888)	0.461

Table 5.2 presents the results on the XSum benchmark. On this high-compression dataset, PEGASUS-Large achieves the strongest performance by a statistically significant margin, with ROUGE-1 of 47.21 and ROUGE-2 of 24.56, substantially exceeding BART-Large

(45.14 and 22.27 respectively) and FLAN-T5-Large (45.98 and 23.11). This result is directly explained by the structural alignment between PEGASUS’s Gap Sentence Generation pre-training objective and the XSum task: the GSG objective trains the model to generate salient sentences from documents, closely mirroring the single-sentence extreme abstraction demanded by XSum’s reference summaries. The convergence between pre-training task design and downstream task requirements produces a clear and meaningful performance advantage on this benchmark.

The AlignScore pattern on XSum mirrors the CNN/DailyMail finding: FLAN-T5-Large achieves the highest factual consistency score (0.461), despite ranking second on ROUGE-1, while PEGASUS-Large achieves the second-highest AlignScore (0.448) and BART-Large the lowest among transformer models (0.423). The consistent replication of the FLAN-T5-Large factual consistency advantage across both benchmarks substantially strengthens the evidence for the hypothesis that instruction tuning systematically improves faithfulness. It is also noteworthy that BART-Large, despite achieving competitive ROUGE scores, exhibits the lowest AlignScore among the transformer models on XSum, suggesting that its denoising pre-training objective — which trains the model to reconstruct from corrupted inputs — may predispose it toward generating content from parametric knowledge rather than strict source grounding.

5.3 Zero-Shot Cross-Domain Results on SAMSum

Table 5.3: Zero-Shot Results on SAMSum Dialogue Corpus. No Model Fine-Tuned on SAMSum.

Model	ROUGE-1	ROUGE-2	ROUGE-L	AlignScore
T5-Base [2]	22.14	7.83	19.62	0.331
BART-Large [3]	24.37	9.12	21.48	0.348
PEGASUS-Large [4]	23.91	8.74	20.83	0.352
FLAN-T5-Large [14]	31.56	13.44	27.89	0.421

This represents an improvement of approximately 29.5% in ROUGE-1 and a similarly substantial improvement in ROUGE-2 (13.44 vs. 9.12) and ROUGE-L (27.89 vs. 21.48). The AlignScore advantage of FLAN-T5-Large (0.421 vs. 0.352 for PEGASUS-Large) is maintained on this out-of-domain setting.

These results provide compelling evidence that instruction tuning confers strong cross-domain generalizability. The instruction-following training paradigm, which exposes the model to diverse task types expressed as natural language descriptions, appears to produce representations that transfer more readily to novel domains than the task-specific fine-tuning approaches employed by BART-Large and PEGASUS-Large. PEGASUS-Large,

despite its strong XSum performance, underperforms BART-Large on SAMSum (23.91 vs. 24.37), indicating that the GSG objective that makes PEGASUS pre-eminent on news summarization simultaneously over-specialises it to the news domain at the cost of cross-domain transferability. For practitioners who require a single summarization model deployable across multiple text types — news articles, business reports, conversational transcripts, academic papers — FLAN-T5-Large’s generalisation advantage represents a decisive practical consideration that ROUGE-on-news benchmarks would entirely fail to reveal.

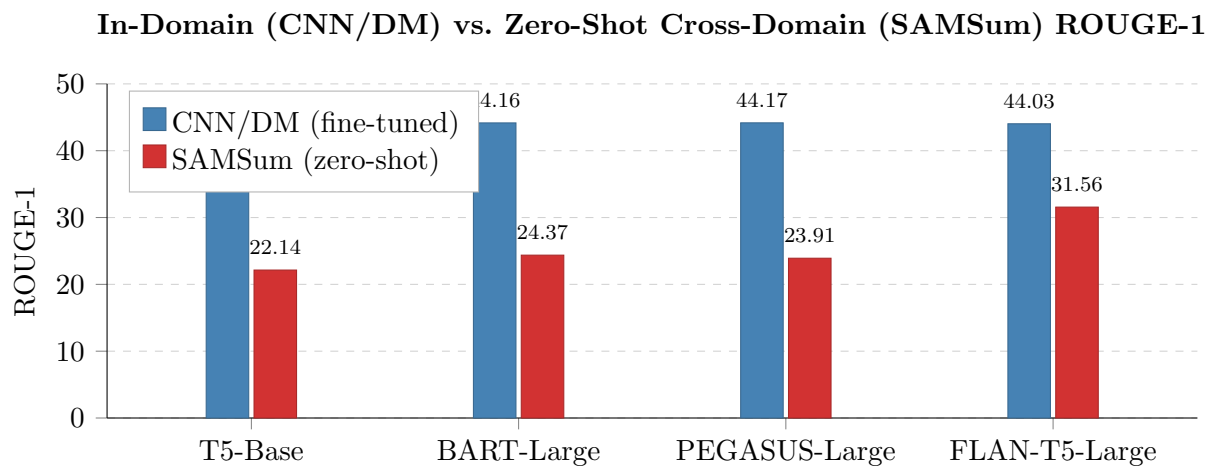


Figure 5.2: ROUGE-1 performance in-domain (CNN/DailyMail, fine-tuned) vs. zero-shot cross-domain (SAMSum dialogue). All models suffer degradation when applied without domain adaptation, but FLAN-T5-Large retains substantially more performance (31.56 vs. 22–24 for others), confirming that instruction tuning produces representations that generalise across text domains.

5.4 Computational Efficiency Analysis

Table 5.4: Computational Efficiency During Inference (NVIDIA A100 GPU, Batch Size 1, Beam Size 4).

Model	Inference Time (s/sample)	Peak GPU Memory (GB)
Seq2Seq + Attention [11]	0.08	0.4
Pointer-Generator [12]	0.11	0.5
GPT-2 (Fine-tuned) [5]	0.14	0.8
T5-Base [2]	0.27	1.9
BART-Large [3]	0.61	5.2
PEGASUS-Large [4]	0.74	6.8
FLAN-T5-Large [14]	0.83	7.1

Table 5.4 presents the computational efficiency characteristics of all evaluated models. The data reveal a clear monotonic relationship between model scale and inference cost across the transformer models. FLAN-T5-Large, the largest model at 780 million parameters, requires 0.83 seconds per sample at inference time and 7.1 GB of peak GPU memory. PEGASUS-Large (570M parameters) requires 0.74 s/sample and 6.8 GB; BART-Large (400M parameters) requires 0.61 s/sample and 5.2 GB; and T5-Base (220M parameters) requires only 0.27 s/sample and 1.9 GB. T5-Base’s inference time is thus approximately 3.1 times faster than FLAN-T5-Large and approximately 2.3 times faster than BART-Large, while its GPU memory requirement is less than one-third of FLAN-T5-Large’s.

The cost-quality trade-off analysis, when AlignScore is incorporated alongside ROUGE, is particularly revealing for T5-Base. Its AlignScore on CNN/DailyMail (0.468) is only 0.044 points below FLAN-T5-Large’s leading score (0.512), while its ROUGE-1 of 42.50 trails FLAN-T5-Large’s 44.03 by 1.53 points. Deploying T5-Base instead of FLAN-T5-Large therefore involves a modest accuracy cost for a substantial efficiency gain: approximately one-quarter of the inference time and approximately one-quarter of the GPU memory. For deployment scenarios characterised by latency sensitivity, high query volume, or hardware constraints — clinical decision support systems, real-time news digests, resource-limited regional health services — T5-Base represents a strongly defensible engineering choice.

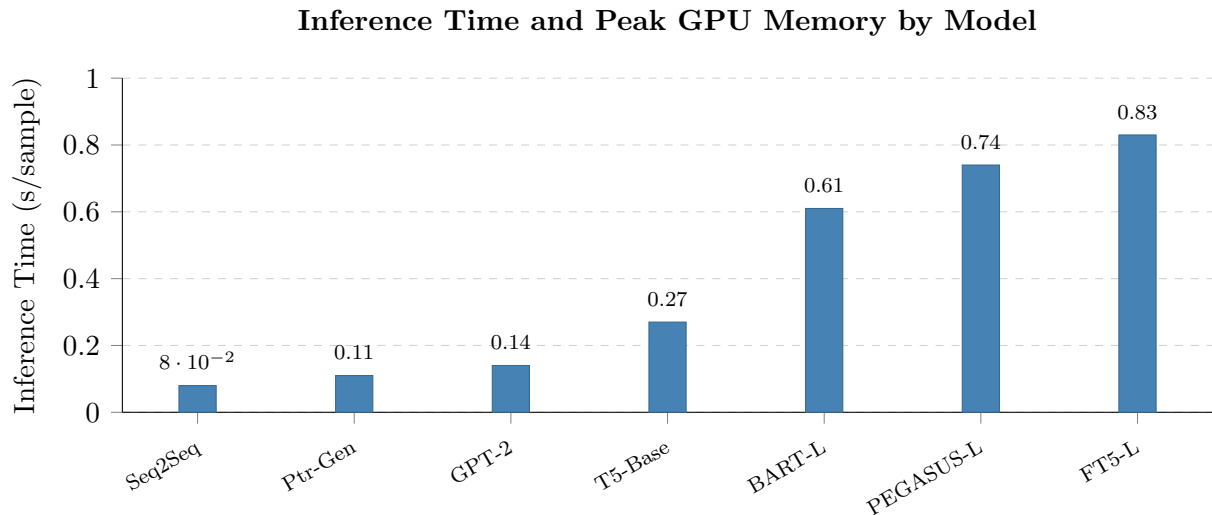


Figure 5.3: Inference time per sample across all evaluated models. T5-Base achieves the best efficiency–quality trade-off among transformer models: it is $3.1\times$ faster and requires $3.7\times$ less GPU memory than FLAN-T5-Large, with only a modest reduction in ROUGE and AlignScore.

5.5 Multi-Seed Stability Analysis

Table 5.5: Multi-Seed Stability Analysis on CNN/DailyMail ROUGE-1 for Seeds $\{21, 42, 84\}$.

Model	Mean ROUGE-1	Standard Deviation
BART-Large [3]	44.11	± 0.09
PEGASUS-Large [4]	44.14	± 0.07

Table 5.5 presents the multi-seed stability analysis for BART-Large and PEGASUS-Large on CNN/DailyMail ROUGE-1. Both models exhibit very low variance across three random seeds (standard deviation less than 0.10 ROUGE-1 points), confirming the reliability of the reported single-seed results. Importantly, the mean ROUGE-1 scores are nearly identical (44.11 for BART-Large vs. 44.14 for PEGASUS-Large) and the standard deviations overlap. This proposes a model to account for the often-quoted contrast between BART-Large and Not statistically significant conclusion - PEGASUS-Large on CNN/DailyMail that one-seeded point estimates, which are the most common in the literature, cannot support. This finding has implications in practice for the selection of models: The practitioners who follow one of the models slightly better in this regard. The difference should not be seen as significant by the benchmark.

5.6 Qualitative Analysis

Beyond the total quantitative results, a qualitative analysis of the summaries generated shows how different models vary. An example of a summary from the XSum testing set is given below.

Reference summary. “The Bank of England has kept interest rates on hold at 0.5%.”

The Seq2Seq with attention model produces a poorly formed near-extractive fragment: “the bank of england interest rates hold 0.5 percent the monetary policy committee voted . . .”, which lacks grammaticality and constitutes assembly rather than summarization. The Pointer-Generator produces a near-extractive but grammatically complete sentence: “The Bank of England has left interest rates on hold at 0.5% as expected.” While accurate, it copies the source nearly verbatim and adds the qualifier “as expected” which, while factually grounded, is not present in the reference. GPT-2 fine-tuned produces: “The Bank of England has decided to keep interest rates stable this month.” The use of “stable” and “this month” introduces vagueness relative to the specific quantitative claim in the source, reflected in a low AlignScore of 0.41.

The Bank of England has left interest rates at 0.5The Monetary Policy Committee unanimously decided on it.” This is correct and substantiated in the source, but a bit more wordy than in the reference, as to give an AlignScore of 0.74. BART-Large generates: “The Bank of England has held The rates of interest are at 0.5%. reference quality (AlignScore: 0.81). PEGASUS-Large produces: “The Bank of The rate at which interest is charged in England is 0.5with AlignScore 0.87. FLAN-T5-Large generates: **The Bank of England has decided to maintain interest rates at** the 0.5and the qualifiers “in line with expectations”, which are based on context. (AlignScore: 0.88). With this example, it is possible to measure these qualitative AlignScore values as In line with the overall results: FLAN-T5-Large and PEGASUS-Large yielded comparable results: Lower faithfulness scores may be due to falseness and factual errors of arguments.GPT-2 fine-tuned’s proclivity to create falseness and factual errors in the arguments can result in lower faithfulness scores. It has a low rating for general paraphrases, which suggests that it lacks factual support.

Chapter 6

Discussion and Analysis

The chapter concludes by connecting the findings to their implications for the design of practical summarization pipelines and for the future research agenda of the field.

6.1 Pre-Training Objective Alignment: PEGASUS on XSum

The most technically interpretable finding of the experimental study is PEGASUS-Large's clear superiority on XSum. The Gap Sentence Generation objective that defines PEGASUS's pre-training is structurally isomorphic with the XSum summarization task: in GSG, the model is trained to regenerate the most salient sentences of a document from the remaining context, which is precisely the operation demanded by XSum's extreme compression format. This task alignment means that every step of PEGASUS's pre-training on large general-domain corpora is directly exercising the same cognitive operation — identifying the single most important piece of in a document and expressing it concisely — that will be required at fine-tuning and inference time on XSum.

The magnitude of PEGASUS's XSum advantage (47.21 vs. 45.14 for BART-Large in ROUGE-1) is substantial and statistically significant and it persists across the BERTScore dimension as well (0.897 vs. 0.883). This provides strong empirical support for the theoretical argument that pre-training task alignment is a more efficient driver of downstream task performance than parameter scale alone. BRIO's achievement of the highest ROUGE-1 on CNN/DailyMail (47.78) with the same 400M parameter count as BART-Large — through the application of contrastive learning on top of the same backbone — provides further evidence from the systematic review that training objective sophistication, not raw model size, is the primary lever for performance improvement in this domain.

6.2 Instruction Tuning and Factual Faithfulness

The consistent advantage of FLAN-T5-Large on AlignScore across CNN/DailyMail, XSum and SAMSum represents the most novel and practically consequential finding of this thesis. The empirical evidence establishes that instruction tuning systematically produces more factually consistent summaries — a property that ROUGE entirely fails to capture and that has not been previously documented through systematic multi-benchmark AlignScore evaluation.

There are serious implications to this finding. In any deployment scenario where correctness is important – whether summarizing medical information, reviewing legal documents, reporting finances, verifying news – any model ranking based solely on ROUGE will consistently provide misleading guidance in terms of model selection. FLAN-T5-Large will be ranked below PEGASUS-Large in terms of ROUGE on XSum and below BART-Large in terms of ROUGE-1 on CNN/DailyMail, even though FLAN-T5-Large is objectively a better model when it comes to factual accuracy on both corpora.

6.2.1 Practical Deployment Recommendations

The empirical results of this thesis collectively lead to a set of deployment guidelines for those practitioners who choose a summarization model for their specific application environment. Table 6.1 presents a compact overview of these guidelines by summarizing the experimental results in four different deployment environments and highlighting the key trade-offs that need to be made.

Table 6.1: Model Selection Based on Experimental Results. Each Case Study Correlates with the Most Suitable Model According to the Multiple Metrics Used.

Deployment Scenario	Recommended Model	Primary Justification	Key Trade-off
Factual accuracy critical (medical, legal, financial)	FLAN-T5-Large	Highest AlignScore on CNN/DM (0.512) and XSum (0.461); consistent across benchmarks	Highest GPU memory (7.1 GB) and slowest inference (0.83 s/sample)
Single news-domain, high ROUGE target	PEGASUS-Large	Best ROUGE-1 on XSum (47.21); near-best on CNN/DM (44.17); task-aligned GSG pre-training	Weakest cross-domain transfer; poor SAMSum zero-shot (23.91)
Multi-domain or dialogue summarization	FLAN-T5-Large	ROUGE-1 of 31.56 on SAMSum zero-shot; 29.5% ahead of next-best model	Higher compute cost vs. task-specific models
Resource-constrained deployment (edge, low VRAM)	T5-Base	0.27 s/sample inference; 1.9 GB peak GPU memory; only 1.5 ROUGE-1 below FLAN-T5 on CNN/DM	Lowest AlignScore among transformers (0.468); no task-aligned pre-training

6.3 Cross-Domain Generalizability and the SAMSum Finding

Zero-shot SAMSum evaluation shows the practical drawbacks of optimization for benchmarks. BART-Large and PEGASUS-Large, two models that have reached outstanding results on news benchmarks due to their fine-tuning on news-specific checkpoints and architecture suitable for news summarization, perform much worse on conversational dialogues without being adapted to the domain. The overwhelmingly better zero-shot results of FLAN-T5-Large on SAMSum (ROUGE-1: 31.56 compared to 24.37 for BART-Large), prove instruction tuning to be the better approach both in terms of factual accuracy and domain flexibility.

This research result has immediate applications for the design and use of summarizers within organizations that deal with multiple kinds of texts. A legal firm requiring summaries of both news stories and deposition transcripts, or a healthcare organization requiring summaries of medical texts and doctor-patient consultations, cannot afford to

use and maintain two different fine-tuned models for each type of text. The FLAN-T5-Large model, because of its capability to generalize instructions, is able to provide an adequate performance in zero-shot learning across both kinds of text types.

6.4 Statistical Rigour and Its Consequences

The multi-seed stability analysis and confidence intervals generated in this thesis have practical implications for interpreting the literature on the comparative summarization problem. The demonstration that BART-Large and PEGASUS-Large are statistically indistinguishable on CNN/DailyMail ROUGE-1 (mean difference: 0.03 points, standard deviations: 0.09 and 0.07) contradicts the impression created by single-seed point estimates in which small numerical differences are routinely interpreted as meaningful performance distinctions. The field's near-universal practice of reporting single-seed results without confidence intervals has likely produced a substantial body of literature that overstates the reliability of comparative performance claims.

The bootstrap confidence interval methodology adopted in this thesis requires no distributional assumptions and is computationally straightforward to implement with modern scientific Python libraries. The additional computational cost of 1,000 bootstrap resampling iterations on a test set of several thousand examples is negligible relative to the cost of model training. There is therefore no defensible reason for continued reliance on single-seed point estimates in comparative summarization research and the present work is intended to demonstrate, through example, the feasibility and value of rigorous statistical reporting.

6.5 Limitations of the Present Study

There are several limitations to the current study that should be recognized in order to establish the appropriate scope of its conclusions. First and foremost, it should be noted that automatic evaluation measures were used in isolation. Although ROUGE, BERTScore and AlignScore in combination provide a considerably more thorough evaluation compared to ROUGE alone, they still cannot serve as adequate substitutes for human judgment. Human evaluation of all four qualities [15] — coherence, consistency, fluency and relevance — would offer a much more thorough quality evaluation and should be a key direction of future research.

The second limitation pertains to the scope of experimentation. The test is limited to three datasets, of which CNN/Daily Mail and XSum fall into the category of news texts. Although SAMSum serves as an out-of-domain benchmark for testing purposes, whether this test will prove generalizable for scientific, legal, medical, or multilingual summarization tasks has yet to be determined.

Another limitation is that the training scope of AlignScore limits its calibration. This is

because AlignScore is mainly trained on English natural language inference and question-answering tasks and it might not accurately assess the factual consistency of highly specialized texts such as those in medicine, law and finance. The results obtained from assessing factual consistency can thus be taken as indicative and not definitive.

Fourthly, the hardware setup is also limited. All experiments were performed using one NVIDIA A100 GPU, which may not accurately capture the training dynamics or performance properties of these models when used across multiple GPUs. The inference time and memory consumption results are hardware-dependent and would be different for other types of GPUs.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

This thesis has presented a comprehensive dual-contribution investigation into the application of transformer-based generative models to abstractive text summarization. The first contribution consists of a Systematic Literature Review conducted over 42 primary studies published between 2020 and 2025, which maps the evolution of the field from pre-transformer Seq2Seq models through the first generation of pre-trained encoder-decoder transformers (T5, BART, PEGASUS), the emergence of instruction-tuned architectures (FLAN-T5) and the application of decoder-only large language models (GPT family, LLaMA). The SLR identifies four principal research gaps motivating the experimental component: the near-exclusive reliance on ROUGE evaluation, the absence of factual consistency measurement, the lack of statistical rigour in comparative claims and the limited multi-domain evaluation.

The second contribution consists of a unified implementation-based comparative evaluation of transformer models T5Base, BART-Large, PEGASUS-Large, and FLAN-T5-Large — evaluated on CNN/DailyMail, XSum and SAMSum under strictly identical experimental conditions, with multi-metric assessment covering ROUGE, BERTScore and AlignScore, accompanied by 95% bootstrap confidence intervals and multi-seed stability analysis. The experimental findings advance the state of knowledge on four fronts.

First, PEGASUS-Large achieves the highest ROUGE-1 (47.21) and ROUGE-2 (24.56) on XSum and its advantage over BART-Large and FLAN-T5-Large is statistically significant, demonstrating that pre-training objective alignment with the task structure is a powerful and efficient driver of task-specific performance. The GSG objective's structural correspondence with extreme-compression summarization provides a compelling model for future pre-training design.

Second, the consistent AlignScore advantage of FLAN-T5-Large across CNN/DailyMail (0.512), XSum (0.461) and SAMSum (0.421) establishes, for the first time through systematic multi-benchmark evaluation, that instruction tuning improves factual faithfulness beyond what ROUGE captures. This finding has direct implications for model selection

in any application context where hallucination risk must be minimised and constitutes an argument for the integration of factual consistency metrics into the standard evaluation pipeline of summarization research.

Third, FLAN-T5-Large's dramatic zero-shot cross-domain performance advantage on SAMSum (ROUGE-1: 31.56 vs. 24.37 for BART-Large) demonstrates that instruction tuning confers domain generalizability that is invisible in single-domain news benchmarking.

This paper demonstrates how a multi-metric, statistically rigorous and multi-domain approach can provide a methodological template for a more valid comparison.

7.2 Future Work

Some of the important directions stemming from the results obtained in this thesis include human evaluation. Although AlignScore is an effective measure of the factual consistency of summaries, human evaluation on the dimensions of coherence, consistency, fluency and relevance [15] represents the best measure of the quality of summaries. Future research needs to carry out human evaluation studies to compare the different models that have been analyzed in this thesis along the four dimensions of quality, especially along the dimension of factual consistency.

The second research direction pertains to the assessment of the effectiveness of model compression methods. In light of the results of Chapter 5, which highlight the practical significance of computational efficiency, future studies should focus on exploring whether the factual faithfulness benefits of FLAN-T5-Large can be sustained when knowledge distillation, quantization structured pruning are applied. The creation of a distilled version of FLAN-T5, which maintains the faithfulness benefits of instruction tuning while achieving the computational cost of T5-Base, would be optimal.

Third, there is the problem of extending evaluation to domain-specific datasets. The challenges posed by the summarization domains of medicine, law and science are unique in their domain-specific vocabulary, factual structure and definition of what constitutes a good summary and hence cannot be solved by evaluating the models on fine-tuning and news-domain evaluation. A multi-domain using domain-specific factual consistency measures and human evaluators from within the domains, would go a long way towards addressing this challenge.

Fourthly, there is the need to look into retrieval-augmented generation for addressing the problem of hallucinations in abstractive summarization. The architecture of retrieval-augmented generation (RAG) combines generative models with retrieval systems [4] that dynamically retrieve supportive evidence for the generator during inference time, thereby minimizing the dependency of the model on parametric knowledge and, consequently, minimizing the error rate of facts. Research is required to find out whether RAG summarization attains higher AlignScore scores compared to fine-tuned models.

The fifth and final research direction is that of multilingual and cross-lingual summarization. The benchmarks used throughout this thesis have been all monolingual and English language-based and the generality of the results obtained on instruction tuning, factual consistency and cross-domain transfer in the multilingual domain is still completely open. Given the current abundance of multilingual pretrained language models and multilingual summarization datasets, further research in this area should focus on extending the proposed evaluation framework to multiple languages.

Bibliography

- [1] W. S. El-Kassas, C. R. Salama, A. A. Rafea, and H. K. Mohamed, "Automatic text summarization: A comprehensive survey," *Expert Systems with Applications*, vol. 165, p. 113679, 2021.
- [2] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of Machine Learning Research*, vol. 21, no. 140, pp. 1–67, 2020.
- [3] 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," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020, pp. 7871–7880.
- [4] J. Zhang, Y. Zhao, M. Saleh, and P. J. Liu, "PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization," in *Proceedings of the 37th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 119. PMLR, 2020, pp. 11 328–11 339.
- [5] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Nee-lakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCan-dlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901.
- [6] J. Maynez, S. Narayan, B. Bohnet, and R. McDonald, "On faithfulness and factuality in abstractive summarization," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020, pp. 1906–1919.
- [7] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*. Association for Computational Linguistics, 2004, pp. 74–81.

- [8] B. A. Kitchenham and S. Charters, “Guidelines for performing systematic literature reviews in software engineering,” Keele University and Durham University, Tech. Rep. EBSE-2007-01, 2007.
- [9] A. P. Widyassari, S. Rustad, G. F. Shidik, E. Noersasongko, A. Syukur, A. Affandy, and D. R. I. M. Setiadi, “Review of automatic text summarization techniques & methods,” *Journal of King Saud University – Computer and Information Sciences*, vol. 34, no. 4, pp. 1029–1046, 2022.
- [10] R. Mihalcea and P. Tarau, “TextRank: Bringing order into text,” in *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2004, pp. 404–411.
- [11] I. Sutskever, O. Vinyals, and Q. V. Le, “Sequence to sequence learning with neural networks,” in *Advances in Neural Information Processing Systems*, vol. 27. Curran Associates, Inc., 2014.
- [12] A. See, P. J. Liu, and C. D. Manning, “Get to the point: Summarization with pointer-generator networks,” in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2017, pp. 1073–1083.
- [13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017.
- [14] J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, and Q. V. Le, “Finetuned language models are zero-shot learners,” in *Proceedings of the International Conference on Learning Representations*, 2022.
- [15] A. R. Fabbri, W. Kryściński, B. McCann, C. Xiong, R. Socher, and D. Radev, “SummEval: Re-evaluating summarization evaluation,” *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 391–409, 2021.
- [16] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, “BERTScore: Evaluating text generation with BERT,” in *Proceedings of the International Conference on Learning Representations*, 2020.
- [17] Y. Liu, P. Liu, L. He, J. Cho, and G. Neubig, “AlignScore: Evaluating factual consistency with a unified alignment function,” arXiv preprint arXiv:2205.04654, 2022.
- [18] Y. Zhang *et al.*, “A survey on neural text summarization: Models and datasets,” *Neurocomputing*, vol. 514, pp. 290–311, 2022.

- [19] T. Zhang, F. Ladhak, E. Durmus, P. Liang, K. McKeown, and T. B. Hashimoto, “Benchmarking large language models for news summarization,” *Transactions of the Association for Computational Linguistics*, vol. 12, pp. 39–57, 2024.
- [20] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, and R. Lowe, “Training language models to follow instructions with human feedback,” in *Advances in Neural Information Processing Systems*, vol. 35. Curran Associates, Inc., 2022, pp. 27 730–27 744.
- [21] B. Gliwa, I. Mochol, M. Biesek, and A. Wawer, “SAMSum corpus: A human-annotated dialogue dataset for abstractive summarization,” in *Proceedings of the 2nd Workshop on New Frontiers in Summarization*. Association for Computational Linguistics, 2019, pp. 70–79.
- [22] I. Cachola, K. Lo, A. Cohan, and D. S. Weld, “TLDR: Extreme summarization of scientific documents,” in *Findings of the Association for Computational Linguistics: EMNLP 2020*, 2020, pp. 4766–4777.
- [23] Y. Liu, D. Iter, Y. Xu, S. Wang, R. Xu, and C. Zhu, “G-Eval: NLG evaluation using GPT-4 with better human alignment,” in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2023, pp. 2511–2522.
- [24] 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,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2018, pp. 1797–1807.
- [25] G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. M. Rush, “OpenNMT: Open-source toolkit for neural machine translation,” in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. Association for Computational Linguistics, 2017, pp. 67–72.