Exploring Cross-Domain Bangla Text Summarization Using Large Language Models

by

Eshika Ebnat Kotha
21201318
Abtahi Bin Jahangir Chowdhury
21201426
Rafiyad Khan Anan
21201094
Subha Naj Showkat
21201398
Sayed Afridi
21201772

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
Brac University
January 2026.

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Eshika Ebnat Kotha

21201318

Subha Naj Showkat

21201398

Rafiyad Khan Anan

21201094

Sayed Afridi

21201772

Abtahi Bin Jahangir Chowdhury

21201426

Approval

The thesis/project titled "Exploring Cross-Domain Bangla Text Summarization Using Large Language Models" submitted by

1. Eshika Ebnat Kotha (21201318)
2. Abtahi Bin Jahangir Chowdhury (21201426)
3. Subha Naj Showkat (21201398)
4. Rafiyad Khan Anan (21201094)
5. Sayed Afridi (21201772)

of Spring 2025 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science in Summer 2025.

Examining Committee:

Supervisor:
(Member)

Dr. Farig Yousuf Sadeque

Associate Professor
Department of Computer Science and Engineering
BRAC University

Co-Supervisor:
(Member)

Md. Saiful Bari Siddiqui

Lecturer
Department of Computer Science and Engineering
Brac University

Thesis Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam

Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi, PhD

Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

Automatic text summarization is a critical tool for managing the growing volume of digital content, yet effective summarization remains challenging for low-resource languages such as Bangla. This thesis investigates the capability of large language models (LLMs) to perform cross-domain Bangla text summarization under a strictly zero-shot setting. Rather than proposing a new summarization model, the study focuses on a systematic and reliable evaluation of existing models across heterogeneous domains. Summarization outputs are generated from two distinct datasets: a real-world Bangla news corpus (Prothom Alo) and the benchmark XL-Sum (Bangla) dataset. A diverse set of encoder–decoder and decoder-only LLMs is evaluated using a multi-layered assessment framework that combines traditional automatic metrics, blind LLM-as-a-Judge evaluation, SBERT-based semantic similarity analysis, and an automated error taxonomy. We assume that we need a more robust comparison beyond surface level lexical matching, which is found ineffective for Bangla abstractive summarization. However, our experimental results show that those lexical metrics (such as ROUGE and BLEU) are generally insufficient to reflect semantic quality for Bangla summary since near zero scores ('0'scores) appear in the case of coherent summarization. In contrast, the semantic analysis shows that Bangla-specific encoder-decoder models including BanglaT5 and mT5 significantly better perform than both the multilingual and decoder-only in domains. Decoder-only models are observed to behave erratically and incline towards either ungrammatical extraction or hallucination, as is systematically verified using semantic similarity patterns and error taxonomy analysis. The findings show that fine summarization in Bangla is insensitive to surface fluency or lexical overlap but dependents on semantic abstraction and faithfulness. We believe that by presenting an exhaustive and behavior-aware evaluation framework, we are able to give practical advice for the future Bangla summarization work so as to demonstrate the importance of language-wise evaluation methodologies especially for low resource languages.

Keywords: Bangla Text Summarization, Cross-Domain Summarization, Large Language Models, Zero-Shot Learning, Semantic Evaluation, LLM-as-a-Judge, SBERT, Error Taxonomy, Low-Resource Languages

Acknowledgement

First and foremost all thanks and praises to Allah (S.W.T) for by His mercy and blessing we managed to complete our thesis without major problems.
We are grateful to our thesis Supervisor Dr. Farig Yousuf Sadeque for his guidance, encouragement and useful suggestions in all the phases of research. We are grateful to him for his generous assistance and faithful support throughout this work.
We would also like to express our heartfelt thanks to our co-supervisor, Md. Saiful Bari Iftu, whose constructive feedback, technical support, and motivation consistently guided us toward achieving our objectives.
Finally, we are deeply grateful to our parents, whose unwavering support, sacrifices, and prayers have always been our source of strength. Without their continuous encouragement, this accomplishment would not have been possible.

Table of Contents

Declaration i

Approval ii

Ethics Statement iv

Abstract iv

Dedication v

Acknowledgment v

Table of Contents vi

List of Figures ix

List of Tables x

Nomenclature x

1 Introduction 1
- 1.1 Background . . . 1
- 1.2 Rationale of the Study . . . 1
- 1.3 Problem Statement . . . 2
- 1.4 Objectives . . . 2
- 1.5 Methodology in Brief . . . 3
- 1.6 Scope and Challenges . . . 4

2 Literature Review 5
- 2.1 Preliminaries . . . 5
- 2.2 Review of Existing Research . . . 7
- 2.3 Summary of Key Findings . . . 17

3 Requirements, Impacts and Constraints 21
- 3.1 System Requirements and Specifications . . . 21
  - 3.1.1 Task Definition . . . 21
  - 3.1.2 Experimental Setting . . . 21
  - 3.1.3 Computational and Software Requirements . . . 22
- 3.2 Evaluation Constraints and Design Considerations . . . 23
  - 3.2.1 Limitations of Lexical Overlap Metrics . . . 23

3.2.2 Semantic Evaluation Using LLMs as Judges . . . 24
3.2.3 Embedding-Based Similarity Analysis . . . 24
3.3 Societal Impact . . . 24
3.4 Ethical and Environmental Considerations . . . 25
3.4.1 Ethical Considerations . . . 25
3.4.2 Environmental Impact . . . 25
3.5 Project Constraints and Risk Analysis . . . 25
3.5.1 Technical Constraints . . . 25
3.5.2 Risk Mitigation Strategies . . . 25
3.6 Economic Considerations . . . 26

4 Methodology and Implementation 27
4.1 Methodology Overview . . . 27
4.2 Datasets and Preprocessing . . . 28
4.2.1 Data Collection . . . 28
4.2.2 Standardization and Storage Format . . . 29
4.2.3 Data Cleaning . . . 29
4.2.4 Data Transformation, Integration, and Reduction . . . 30
4.3 Model Description . . . 30
4.3.1 Transformer Architectures . . . 31
4.3.2 Architectural Comparison . . . 31
4.3.3 Model Specification Table . . . 32
4.4 Experimental Design . . . 32
4.4.1 Zero-Shot Summarization Protocol . . . 32
4.4.2 Output Collection and Organization . . . 33
4.5 Evaluation Framework . . . 33
4.5.1 Lexical and Embedding-Based Metrics . . . 33
4.5.2 Evaluation Framework Overview . . . 35
4.5.3 Blind LLM-as-a-Judge Evaluation . . . 37
4.5.4 SBERT Cosine Similarity Analysis . . . 37
4.5.5 Automated Error Taxonomy . . . 40
4.6 Chapter Summary . . . 41

5 Results and Analysis 42
5.1 Overview of Evaluation Results . . . 42
5.2 Automatic Metric Results . . . 42
5.3 LLM-as-a-Judge Evaluation Using Qwen2.5 . . . 43
5.3.1 Five-Model Comparison . . . 44
5.3.2 Four Bangla-Oriented Model Comparison . . . 45
5.4 Blind LLM-as-a-Judge Evaluation Using ChatGPT and Gemini . . . 45
5.4.1 Blind Evaluation Protocol . . . 45
5.4.2 Post-hoc Model Identity Mapping . . . 46
5.4.3 LLM-as-a-Judge Results (Side-by-Side Comparison) . . . 46
5.4.4 Cross-Judge Agreement and Interpretation . . . 47
5.4.5 Comparative Analysis Under Alternative LLM-as-a-Judge Criteria . . . 47
5.5 Semantic Similarity Analysis and Automated Error Taxonomy . . . 49
5.5.1 Case Studies (Qualitative Analysis) . . . 54

6 Conclusion **59**
6.1 Summary of Contributions 59
6.2 Implications for Bangla Summarization Research 60
6.3 Limitations and Directions for Future Work 61
6.4 Concluding Remarks 61

Bibliography **63**

List of Figures

1.1 Workflow diagram illustrating the methodological approach used in this thesis. 3

4.1 Example of a JSONL record containing `article` and `summary` fields. . 29
4.2 Side-by-side comparison of encoder–decoder and decoder-only transformer architectures used in this study. 31
4.3 Overview of the multi-stage evaluation framework 35
4.4 SBERT Model Archiecture Diagram 39

5.2 Qwen2.5 judge-based evaluation of Bangla-oriented models across datasets. 45
5.3 Per-model SBERT evaluator comparison on the Prothom Alo dataset. 50
5.4 Per-model SBERT evaluator comparison on the XL-Sum dataset. . . 50
5.5 Normalized error composition by model across all generations. 52
5.6 Error type prevalence comparison between Prothom Alo and XL-Sum datasets. 52
5.7 Error propensity heatmap identifying dominant failure modes per model. 53
5.8 Case Study A (XL-Sum): Reference and BanglaT5 outputs. 54
5.9 Case Study A (XL-Sum): mT5 and decoder-only outputs. 55
5.10 Case Study B (Prothom Alo): Reference and BanglaT5 outputs. . . . 55
5.11 Case Study B (Prothom Alo): mT5 and decoder-only outputs. 56
5.12 Case Study C (XL-Sum): Reference and BanglaT5 output. 56
5.13 Case Study C (XL-Sum): mT5 and decoder-only output (semantic misalignment). 57
5.14 Case Study D (Prothom Alo): Reference and BanglaT5 output. . . . 57
5.15 Case Study D (Prothom Alo): mT5 and decoder-only output (unsupported detail risk). 58

List of Tables

2.1 Summary Table of Important Papers 17

3.1 Hardware specifications used for experimentation 23
3.2 Software tools and libraries used in the study 23
3.3 Evaluation methods used in the study 24

4.1 Overview of datasets used for Bangla and cross-lingual summarization. 28
4.2 Model specifications and architectural characteristics (compact variants) 32
4.3 Automated error taxonomy used in this thesis 40

5.1 ROUGE, BLEU, and BERTScore results on the Prothom Alo dataset. 43
5.2 ROUGE, BLEU, and BERTScore results on the XL-Sum dataset. . . 43
5.3 Mapping between anonymized evaluation IDs and actual models . . . 46
5.4 Gemini blind evaluation scores across datasets 46
5.5 ChatGPT blind evaluation scores across datasets 47
5.6 Top-5 summaries under originality-dominant blind evaluation 48
5.7 Top-5 summaries under faithfulness-oriented blind evaluation 49

Chapter 1

Introduction

1.1 Background

Summarization is an important problem in information processing because it allows large-scale textual content to be condensed into short, meaningful representations that are easier for humans to consume. In an era of information overload, summarization supports faster comprehension and more efficient decision-making by presenting the main points of long documents in a compact form. Text summarization can be broadly categorized into two approaches: extractive and abstractive. Extractive summarization selects and reproduces sentences directly from the source, while abstractive summarization generates novel sentences that paraphrase and synthesize information from the original text. Abstractive approaches such as [1], [2], although more flexible and expressive, commonly suffer from factual inconsistency, redundancy, and hallucination, which makes both generation and evaluation more challenging.

The field has progressed further through multilingual and cross-lingual modeling, where systems aim to generate summaries using shared multilingual representations and transfer learning. These approaches diminish reliance on extensive task-specific data and enable knowledge to transfer from high-resource languages. Meanwhile, large language models (transformers) have shown impressive generalization ability on a variety of natural language processing tasks even under zero-shot settings. But for low-resource languages like Bangla, summarization is still challenging due to lack of large-scale annotated corpora, domain mismatch between training and test data, unstable evaluation metrics which calculate the overlap based on surface level text.

1.2 Rationale of the Study

Bangla, one of the most spoken languages, is yet under-represented in the state-of-the-art NLP research in terms of large datasets, benchmark coverage and evaluation robustness. The majority of Bangla summarization works are focused on the news domain and the practice summaries in most datasets have variable quality. These constraints challenge the supervised training robustness and hinder evaluation under task shifts, especially domain shift. Consequently, gains seen in benchmark tests may not carry over to normal usage. One major motivation behind this thesis is that summarization research in case of Bangla is in need of not just stronger models, but

also better evaluation frameworks which are not only more robust, but also interpretable. Rich morpho-syntax, flexible word ordering and frequent paraphrasing in Bangla decrease the literal token overlap between summaries and references. This means that popular word-overlap based metrics, such as ROUGE or BLEU, often underestimate semantic quality. This in turn motivates the deeper investigation of evaluation methodology. Instead of being driven to achieve good generator quality, this study seeks to learn about how the different evaluation paradigm for Bangla and its impact on the conclusions drawn, particularly when evaluated in cross-domain setting.

1.3 Problem Statement

This thesis investigated how effectively large language models could generate high-quality Bangla summaries across different domains in a strictly zero-shot setting, and how such outputs could be evaluated reliably when traditional lexical metrics failed.

In simpler terms, the research problem was to determine to what extent and by what means different model families were able to produce Bangla summaries when the document domain changed and no task-specific adaptation was allowed. This included evaluating summarization robustness on two datasets with different characteristics: Prothom Alo (real-world Bangla news) and XL-Sum (Bangla) (benchmark-style summaries). In addition, the problem explicitly addressed the evaluation challenge that arose when reference-based lexical metrics collapsed to near-zero values, requiring alternative semantically grounded evaluation strategies to compare models fairly and interpret results in a meaningful way.

1.4 Objectives

To address the stated problem, this research sets out the following objectives:

- **1. Evaluate zero-shot Bangla summarization across model families:** Systematically evaluate nine models under the same zero-shot prompting protocol, covering encoder–decoder and decoder-only architectures, including multilingual baselines and Bangla-oriented models, on Prothom Alo and XL-Sum (Bangla), in order to assess cross-domain robustness.
- **2. Empirically examine the reliability of traditional automatic metrics for Bangla:** Report and analyze the observed behavior of ROUGE, BLEU, and BERTScore on Bangla summarization outputs, and empirically demonstrate why these metrics fail to reflect semantic quality in morphologically rich languages.
- **3. Apply semantic evaluation using LLM-as-a-Judge under controlled conditions:** Use Qwen2.5 for criterion-wise semantic scoring and conduct blind ranking with ChatGPT and Gemini to reduce evaluator bias and increase alignment with human judgment principles.

- **4. Quantify semantic alignment with embedding-based similarity:** Use SBERT-based cosine similarity analysis to measure Reference–Generated and Article–Generated alignment, enabling differentiation between abstractive summarization and extractive copying.
- **5. Provide interpretable diagnosis of failure modes:** Implement an automated error taxonomy to systematically categorize dominant failure patterns and explain model weaknesses beyond aggregate scores.

1.5 Methodology in Brief

A controlled experimental setting is adopted in this thesis with respect to zero-shot summarization and multi-stage evaluation methodology. Uniform zero-shot prompt and fixed generation are imposed to all the models in order to make a fair cross-model and dataset comparison. No fine-tuning is performed for any task and no prompt adaptation is applied. A bird's eye view of the process from top to bottom is depicted in figure 1.1.

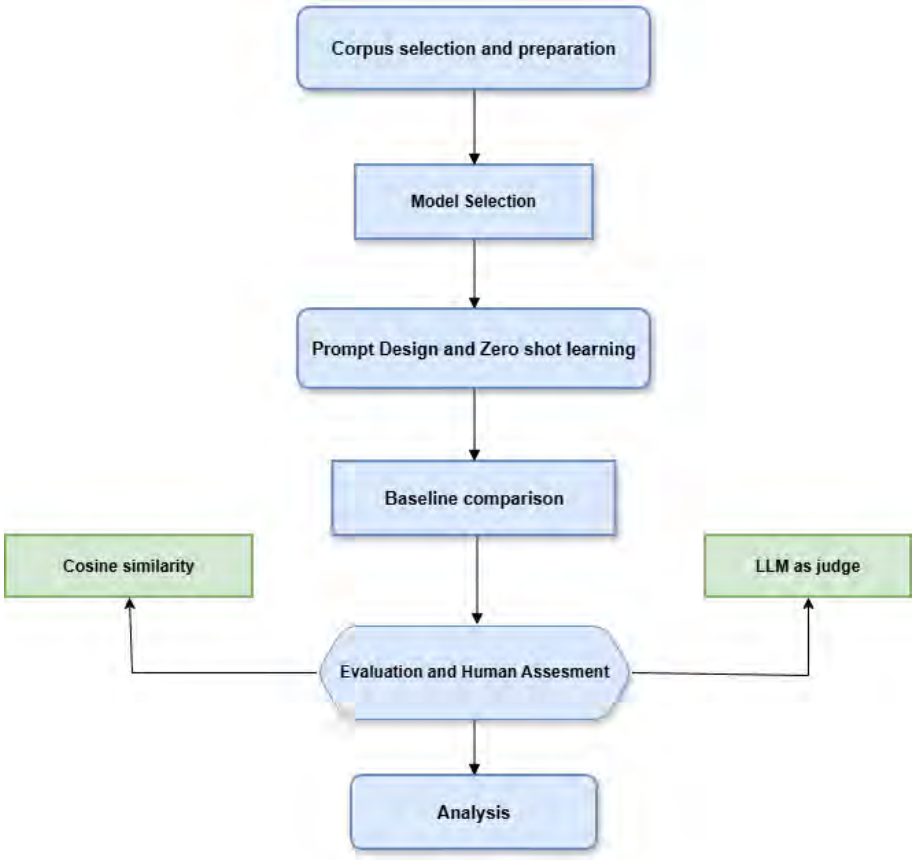


Figure 1.1: Workflow diagram illustrating the methodological approach used in this thesis.

The core steps of the methodology are summarized as follows:

- **Corpus selection and preparation:** Two datasets are used: Prothom Alo, representing real-world Bangla news articles, and XL-Sum (Bangla), representing benchmark-style summaries. All source articles, reference summaries,

and generated outputs are stored in structured CSV files to support blind evaluation, automated analysis, and reproducibility.

- **Model selection:** Nine models are evaluated to capture architectural diversity and varying degrees of Bangla specialization. This includes strong encoder–decoder baselines (BanglaT5 and mT5) as well as multiple decoder-only models marketed for multilingual or Bangla-specific use.
- **Zero-shot summarization:** All models generate summaries without in-context examples, fine-tuning, or domain adaptation. This isolates inherent model generalization ability and enables fair cross-model comparison under identical conditions.
- **Evaluation and analysis:** Due to the unreliability of lexical metrics for Bangla, evaluation proceeds in multiple layers: Qwen2.5 LLM-as-a-Judge scoring, blind ranking using ChatGPT and Gemini, SBERT-based semantic similarity analysis, and automated error taxonomy. These complementary perspectives allow triangulation of findings and increase confidence in conclusions.

1.6 Scope and Challenges

Scope of the Study: In this work, we focused on single-document abstractive summarization for Bangla under zero-shot condition. The task was comparative processes and not training or optimizing a model. Cross-domain robustness was investigated using two datasets of distinct feature. The focus was on evaluation methodology/interpretability rather than achieving the highest possible absolute generation quality.

Key Challenges and Constraints: One of the key challenges was the inadequacy of traditional lexical overlap metrics for a Bangla summarization evaluation. Paraphrasing and morphological variation lead ROUGE and BLEU to converge toward zero even for semantically correct summaries. Another constraint was that some evaluated models produce unstable outputs in Bangla, including excessive extractive copying, hallucination, or technical noise. These behaviors necessitate diagnostic analysis beyond aggregate scores. Finally, because each semantic evaluator may introduce its own bias, the study relies on triangulation across judges, embedding models, and taxonomy-based analysis to ensure robust and defensible conclusions.

Chapter 2

Literature Review

2.1 Preliminaries

The presented literature review aims at exploring and investigating some articles that have been published in 2017-2025 with a focus on the major Bangla text summarization contributions of the world and the region. The works that were looked at adopt various methodologies, from acknowledged statistical models to innovative deep neural networks and large language models (LLMs) representing the latest developments. In this section, we are giving a brief overview of the fundamental theoretical ideas, definitions, and models which are the cornerstones of our inquiry. It also contains a description of various summarization kinds, an attempt to clarify the interdisciplinary nature of the summarization (mainly crossing languages), a short introduction to large language models and their relation to our project.

Text Summarization Fundamentals: Text summarization is the process of computer-supported creation of short versions of lengthy texts wherein the important part of the original text is kept intact. It is broadly divided into:

- **Extractive summarization** chooses important sentences or phrases from the original text.
- **Abstractive summarization** formulates primary sentences that represent the gist of the material.

Cross-Lingual Summarization (CLS): CLS means making a summary of a different language than the original one for the document thus it is a combination of both tasks, summary and translation. To illustrate, providing a summary of an English text in Bangla implies the selection of the most important information and then translating it in a correct and fluent way. Generally, CLS has thus far been solved with pipeline methods:

- **(i)** either translating the whole document before doing the summarization (translate-then-summarize)
- **(ii)** first do the summarization followed by the translation (summarize-then-translate).

Although these two methods are very easy, they still have the problem that as errors are amplified, one can lead another to amplifying the final output. In the recent past, end-to-end models such as mBART and mT5 have been employed to translate words from one language to another directly for summarization purposes. Such models may be able to circumvent some of the problems faced by pipeline approaches as they acquire the ability to juggle both tasks at the same time. On the other hand, these models necessitate extensive, varied cross-lingual datasets for training, which have been rather scarce. Resources like CrossSum that are relatively new, are broadening the language coverage and providing bigger datasets which in turn makes end-to-end CLS more attainable and potent. These sets can also be used to explore new languages, thereby giving the ability to train cross-lingual models for more languages.

Large Language Models (LLMs): A few of the examples of LLMs are GPT-3, GPT-4, and PaLM; they are just enormous neural models which are trained on extensive text data. One of the most important characteristics of recent LLMs lies in-context or prompt-based learning, thus they can easily do any of the tasks like translation or summarization by simply giving instruction in prompt without further training. LLMs support:

- **Zero-shot learning**, where no examples are presented for a task, yet models like GPT-4 perform well, including in cross-lingual summarization.
- **Few-shot learning**, where the input of a couple of samples in the prompt enables the output quality to be improved.
- **Instruction tuning and RLHF**, where models are fine-tuned to consider human instructions, this being so they can be more useful for directed tasks, for example, producing short summaries or focusing on certain content.

Such features make LLMs potential instruments for adaptive and efficient summarization even in less-resourced languages.

Relevance of LLMs to Bangla Summarization: LLMs are directly related to Bangla summarization tasks in that they constitute a powerful alternative for low-resource tasks like Bangla summarization while at the same time they significantly cut down the need for large datasets that are Bangla-specific. Existing research such as Wang et al. (2023) demonstrates that with appropriate prompting, GPT-4 can perform as well as or even better than zero-shot at cross-lingual summarization, though it tends to be verbose [12]. Park et al. (2024) shows that few-shot prompting largely contributes to better results in low-resource languages and that bigger models are always ahead of the race when compared to smaller ones [17]. Li et al. (2025) presented a stepwise prompting approach named SITR (Summarization, Improvement, Translation, Refinement), revealing that leading LLMs step-by-step with a structured framework improves summarization accuracy and fluency [20]. Evaluation is still difficult. Although ROUGE, BLEU, and METEOR give some indication, innovations such as LaSE employ multilingual embeddings to determine how good the summary is without having to check with reference translations [8], [18]. In the end, human evaluation is still the most reliable way to decide whether

the Bangla summaries are clear and helpful [22]. Individual research contributions and their methodologies are summarized in detail in the next sections.

2.2 Review of Existing Research

In 2023, Bhattacharjee and colleagues designed BanglaT5, a domain-specific encoder-decoder language model tailored for Bangla natural language generation (NLG) [9]. Together with BanglaT5, the authors came up with a comprehensive benchmark suite called BanglaNLG for the evaluation of their model. The benchmark comprises six different NLG tasks: machine translation, text summarization, question answering, multi-turn dialogue, headline generation, and cross-lingual summarization. Abhik Bhattacharjee, Tahmid Hasan, Wasi Uddin Ahmad, and Rifat Shahriyar are among the authors who introduced a new multi-turn dialogue dataset by translating and human-curating the DailyDialog corpus for Bangla. To achieve consistent high performance on various tasks, the authors decided to pre-train BanglaT5 on Bangla2B+, a 27.5GB clean Bangla corpus. They also used a span corruption (denoising) objective, which is a part of the T5 training framework. They preferred this objective over BART's text-infilling or ProphetNet's future n-gram prediction as it gave them more flexibility in generation tasks. They demonstrated that BanglaT5 consistently outperforms multilingual models like mT5, mBART-50, XML-ProphetNet, and IndicBART in five of six tasks. Particularly strong results are shown in QA and machine translation, even though the model had minimal English data exposure. They have attributed the success of BanglaT5 to the high quality of the pretraining corpus curated by Bhattacharjee et al., vocabulary construction which preserved non-Bangla tokens, and efficient training on TPUs, making the model both effective and resource-friendly.

Another paper by Bhattacharjee et al. (2023) presents CrossSum, a massive multilingual cross-lingual summarization dataset, which is a collection of over 1.68 million article-summary pairs across more than 1,500 language pairs the largest publicly available dataset of its kind [8]. The authors used a cross-lingual retrieval method to find matching articles in different languages, enabling them to obtain high-quality alignments validated through human evaluation. To address language imbalance and resource scarcity, they proposed a multistage language sampling (MLS) algorithm, enabling models to generalize across language pairs efficiently. Furthermore, they introduced LaSE, an embedding-based evaluation metric that is highly correlated with ROUGE and can evaluate summaries in low-resource languages without requiring reference summaries. Their experiments confirmed that models trained on CrossSum outperform baselines, and the dataset's multilingual richness enables inclusive cross-lingual summarization research beyond English-centric constraints.

In 2023, Wang and his colleagues aimed to test the zero-shot cross-lingual summarization (CLS) abilities of large language models (LLMs), mainly GPT-4 and ChatGPT, thus emphasizing their possibility to create target-language summaries without fine-tuning [12]. In the beginning, the authors (Wang et al., 2023) designed different LLMs prompting methods – those direct and the chain-of-thought ones – in order to enable the LLMs to carry out the whole CLS process (Section 2.1).

They checked the models' performance on a number of datasets that covered various domains and languages, thus showing that GPT-4 still achieves the best results in the zero-shot settings, which are often equal to the performance of the fine-tuned models, like mBART-50 (Section 4.1). Significantly, Wang et al. (2023) report interactive prompts as a useful means for models to enhance the control of their informativeness versus conciseness, although the changes in LLM-based evaluation scores that they cause are not clear . This study moves beyond traditional pipeline and task-specific models to the new frontier of zero-shot general-purpose LLMs for CLS by conducting a systematic survey (Section 4.1) . Nevertheless, the authors highlight that the models are not yet perfect in generating concise summaries and their results are quite unstable when testing different LLMs. They announce that future research should target improving prompt creation, increasing model scalability, and coming up with better evaluation tools, thus making CLS a suitable benchmark for testing LLM's ability in the future (Section 5).

Rony and Islam [22] Rony and Islam Perform an evaluation on a Bangla news summarization task using source code which LLMs (GPT-4) is best suited for the BANS and BNLPC dataset. They used human written summaries from student annotators for comparison in order to circumvent the limitations of previous evaluations with poor quality reference summaries. They show that GPT-4 was able to generate summaries in zero-shot settings which were on par with human-written ones in terms of coherence, faithfulness, and relevance. They also noticed that the effect of instruction tuning was more significant for summarization than model size. Human evaluations with faithfulness, coherence, and relevence scores showed that GPT-4 exceeded or equalled the quality of student-written summaries which illustrated the potential of LLMs for low-resource languages like Bangla. The analysis highlights the necessity of high-quality reference summaries and demonstrates that GPT-4 can achieve state-of-the-art performance even in an unfine-tuned setting.

Li et al. (2024) introduced an iterative four-stage zero-shot prompting framework, termed SITR (Summarization, Improvement, Translation, and Refinement), to enhance the performance of large language models (LLMs) for cross-lingual summarization (CLS) in low-resource language settings [20]. Unlike traditional pipeline approaches that suffer from cumulative error propagation, SITR performs meta-generation by incrementally refining summaries and translations through multiple controlled stages. On CrossSum and WikiLingua, SITR models (especially GPT-4) demonstrate a large leap in performance over strong finely-tuned models like mBART-50 and mT5 as well as standard LLM baselines under both ROUGE and BERTScore metrics. The research also shows the open-source models like LLaMA 3.1 and GEMMA-2 get high performance boost from SITR, which sometimes could even outperform some GPT-4 variants. Ablation experiments confirm that each stage of the framework contributes meaningfully to performance gains, with the Improvement and Refinement stages playing a particularly critical role. Overall, the findings highlight the robustness of the SITR framework and its effectiveness in enabling high-quality cross-lingual summarization for resource-scarce languages without task-specific fine-tuning.

In [16], Sanzana Karim Lora and Rifat Shahriyar (2023) in their paper "CON-

VERSUM: A Contrastive Learning Based Approach for Data-Scarce Solution of Cross-Lingual Summarization Beyond Direct Equivalents" introduce a new way to perform CLS when we don't have access to large-scale parallel corpora. In this work, the authors propose ConVerSum, a contrastive learning based approach for multilingual candidate summary generation and ranking. In particular, Lora (2023) describes how even mT5 and FLAN-T5 seq2seq models with training using contrastive ranking loss lead to diverse candidate summaries in the summarization process which are good for semantic similarity estimation. Shahriyar (2023) also highlights that such evaluation metrics are important in order to rank candidate summaries and can assist the model in generating more accurate cross-lingual summaries. The method carefully pairs its positive and negative summaries to help low-resource CLS performance, surpassing the full size GPT-4 using this low-resource setting as study case. In sum, this work well shows that contrastive learning can serve as an effective data-efficient strategy for CLS without heavy dependency on large parallel corpora, therefore opening the door to more accessible multilingual summarization.

Park et al. (2024) explore the few-shot learning capabilities of large language models (LLMs) for low-resource crosslingual summarization (XLS) using a direct approach [17]. The experiment compared GPT-3. 5, GPT-4 (w9kda) and an open-source Mistral-7B-Instruct-v0. 2 for different language pairs with one and two-shot prompts. The results demonstrate that GPT-3.5 and GPT-4 notably reinforced performance with few-shot learning thus, many-to-one situations (summarizing into English) that is, outperforming zero-shot setups. On the other hand, Mistral-7B was most limited and struggled to perform various one-to-many cases inconsistently with more than one example. The authors opine that although few-shot learning elevates XLS performance in LLMs, it is still futile in quite low-resource language directions (like English to Pashto). As a result, it is necessary to explore new architectures, prompt strategies, and transfer learning methods which are specially designed for such tasks.

Paulus et al. (2017) introduce a neural network architecture that is designed for the purpose of abstractive summarization and is a solution to the problem of generated summaries that mainly are repetitive and incoherent [1]. The model is an intra-decoder attention mechanism as well, which, according to the authors, helps establish coherence for longer summaries by re-using the memory of previously attended input tokens in experiments on the CNN/Daily Mail dataset. Furthermore, the authors go further to combine supervised learning with reinforcement learning (RL) (following Nallapati et al.'s 2016 approach) as a method of solving the problem of bias, and thus they get more readable and more diverse output. Their training scheme relies on RL to fine-tune sequence-level metrics besides allowing their system to keep the integrity of the individual word predictions, which results in the best performance to date, like a ROUGE-1 score of 41.16 on CNN/Daily Mail. The model was also tested on the New York Times dataset, which is different from the CNN/Daily Mail dataset, thus demonstrating the model's robustness even when it is given multiple datasets and different summary styles.

Chen and Bansal (2018) presented a new two-stage summarization method called

Fast Abstractive Summarization with Reinforce-Selected Sentence Rewriting, that is based on the utilization of both extractive and abstractive techniques for better efficiency and performance [2]. The first part of their model picks up the most important sentences by a reinforcement learning-based extractor and then the latter is changed to an abstracted one using a sequence-to-sequence model employing attention and a copy mechanism. The hierarchical approach is the same as the way people do document summarization finding the most relevant parts of the document and then just rephrasing those parts in the words of the summary. The authors implemented a sentence-level policy gradient technique to transfer the non-differentiable piece of extraction and abstraction, thus allowing them to train the model end-to-end without losing language fluency. Their model by far is ahead of previous models on the CNN/Daily Mail dataset evaluated using ROUGE and METEOR metrics, and also shows good generalization to the DUC-2002 dataset. To put it short, the output of the model was more abstractive (i.e., it contained a higher number of new n-grams), repetitions were reduced, and it was remarkably quicker than the previous ones 10–20 times faster in the inference stage and 4 times faster in training than the models of See et al. (2017). The final version of the model with reranking boosted the summary further to the top and provided better human evaluation scores in both relevance and readability . Their works highlight the adoption of sentence-level reinforcement learning for summarization tasks, the design of a hybrid extractive-abstractive architecture that supports parallel decoding, as well as setting a new state-of-the-art performance record in terms of both accuracy and efficiency.

Pernes et al. (2024) present the completely new task of multi-target cross-lingual summarization (MTXLS), which is about producing summaries that are semantically consistent in several languages simultaneously, a problem that is still almost untouched by previous research [18]. To this end, they come up with a language-neutral re-ranking method called NeutralRR, which not only makes it possible to find sets of summaries that are mutually semantically similar across languages, but also gets rid of the necessity for a pivot language and hence allows unbiased multilingual outputs. In addition, the authors of the paper enlarge their framework by utilizing a dynamic programming-based optimization method that takes them to the solution most likely to be the one that matches their hypothesis on semantic similarity transitivity, thereby quickly searching for the optimum set of candidate summaries . Furthermore, they transform the CrossSum dataset into groups of multilingual document-summary pairs, thus enabling their methods to be tested across languages. Their observations establish that the models like M2MS are powerful in their ability to meet traditional readability criteria including ROUGE; however, they seldom have cross-lingual semantic coherence, which is notably enhanced by the introduced re-ranking strategies, thus illustrating the significance of their method for dependable multilingual summarization systems. In general, the authors Pernes et al. present a solid conceptual framework and evaluation protocol that facilitate the improvement of more semantically consistent cross-lingual summarization models, thus filling in the missing parts in the current literature .

G-Eval: NLG Evaluation with Better Human Alignment by Liu et al.(2023) is a

new paradigm of improvements to reference-free evaluation of natural language generation (NLG) systems and is based on the use of the large language models (LLM) as an evaluator with a chain-of-thought (CoT) and form-filling paradigm. Existing evaluators with BLEU and ROUGE show little correlation with human judgments, particularly with creative or open-ended tasks, such as summarization and dialogue generation, and the current LLM-based evaluators are comparatively weak to medium-scale neural. G-Eval solves these shortcomings by asking GPT-4 to generate CoT evaluation steps based on an introduction of the task and the evaluation criteria, then generating structured CoT evaluation steps, and finally scoring the outputs using the evaluation steps in a form-filling way. The experiments on standard benchmarks show that G-Eval using GPT-4 is associated with Spearman correlation of 0.514 with human ratings in summarization tasks, which is much higher than past methods, and also does well on dialogue evaluation. The research contains the study of possible evaluator behavior, in particular, the dependence of the evaluator on the results produced by similar models can be observed. The given finding demonstrates the potential and limitations of using LLM as automatic assessors of NLG quality and informs future studies of the relevance of calibration and sound evaluation design. The G-Eval method is applicable to the tasks that do not require reference texts and where surface textual overlap is not the most crucial feature but semantic judgment, which explains its direct applicability to the studies that use the quality of text summarization estimated by using LLM-based evaluation.

The article by Zheng et al.(2021), J judging LLM-as-a-Judge with MT-Bench and Chatbot Arena, which systematically studies the application of large language models (LLMs) as automated judges to rate other generative models The authors pay attention to the evaluation paradigm of LLM-as-a-Judge and its abilities and limitations, especially position bias (prefer one answer position to another), verbosity bias (prefer longer answers), and self-enhancement bias (prefer the results of similar or the same model family). To empirically measure these problems and determine the practical usefulness of LLM judges, the paper presents two assessment levels: an MT-Bench (a multi-turn question set) and Chatbot Arena (a crowdsourced platform, where human preferences are used as a ground truth of comparative judgments). Findings indicate that well-performing LLM judges like GPT-4 can hit human preference at above 80 percent agreement, which is close to human-to-human agreement in generative tasks, indicating that LLM judges can be easily scaled and explained as alternatives to expensive human evaluation. The analysis also proves that traditional benchmarks can be supplemented by commercial semantic assessments by LLM judges, who can give subtle semantic assessment as opposed to surface-based ones. Nevertheless, the bias examination suggests that there are considerable difficulties to reliability and fairness in automated evaluation designs, and bias reduction methods are required. Generally, this paper gives evidence of how promising the use of LLM judges is in the evaluation pipeline and important insight into what causes systematic biases that need to be overcome to have a strong deployment.

Tatsu-Lab (`tatsu-lab-alpaca_eval`) by Dubois et al. (2024) introduces AlpacaEval, an automated evaluation system that involves the use of large language models

(LLMs) to evaluate their performance by having the outputs of the model evaluated in a pairwise preference interface without requiring human intervention to do so. AlpacaEval is an LLM judge that is usually GPT-4, and adjudicates two model responses to the same instruction based on which one is preferable, giving out win-rate scores that compare the quality of the output of different models used. This system has been tested on large scale human preferences (=20 K annotations), and shows a high correlation to human judgments, with much faster, less expensive and more repeatable results than human judgment. The framework also has mechanisms to randomize output ordering to reduce simple biases and leader board generation in many models, which is useful in rapid benchmarking when developing models and doing research. One important extension, which is the AlpacaEval 2.0, adds length-controlled win rates to eliminate verbosity bias, the propensity of automatic judges to rate longer outputs more highly, and makes model rankings even more robust and interpretable. AlpacaEval is now a standard benchmark in the research community around the LLM to evaluate the quality of models based on the quality of their instruction-following, and an alternative to the expensive human evaluation pipelines, although residual judgment biases still exist and results need to be carefully interpreted. The framework is also applicable to the context of summarization evaluation research since it is one of the examples of how the LLM can serve as an automated judge of generative outputs, providing clues to the automated semantic quality evaluation that may be applied to your own work on the subject of the LLM-based scoring and assessment using other evaluation criteria.

Suggested by Kryscinski et al(2020). another major drawback of the classical measures, e.g. ROUGE, which rely on superficial overlap between the summary and the original text but not its accuracy, is overcome by a weakly supervised and model-based framework to assess the factual consistency of abstractive text summaries. The article presents FactCC that is trained based on artificial data produced by running rule-based transformations on original documents to obtain consistent and inconsistent sentence summaries without expensive human annotations at scale. The model collectively carries out three functions, which include classifying a summary sentence as factually consistent with the source, extracting supporting evidence spans out of the source in consistent cases, and determining erroneous spans in inconsistent summaries. Experiments on the outputs of several neural summarization systems indicate that this method greatly performs better than the methods previously based on natural language inference and question answering when error of fact is being detected. Human assessments further show span-extraction goals are better in interpretability and accuracy in verification. Annotated datasets, training code and pretrained models also are released by the authors rendering this work a pillar in automated factuality evaluation in abstractive summarization and very relevant to studies on semantic and faithfulness-based evaluation structures.

Deode et al.(2023) suggest L3Cube- IndicSBERT: A simple cross-lingual sentence representation learning method based on multilingual BERT proposes a more practical approach of learning sentence-level embeddings of multiple Indic languages with fine-tuning standard multilingual BERT models in an SBERT form. The authors note that recent vanilla vanilla multilingual BERT models are not

well-posed to semantically represent sentences, which are required by similarity and clustering tasks, particularly low-resource languages and that training vanilla SBERT models per language is both data-intensive and complex. To deal with this, they create synthetic NLI and STS datasets of ten large Indic languages and carry out SBERT-style fine-tuning of a base multilingual model not by using explicit cross-lingual training data. The result of this process is L3Cube-IndicSBERT, a set of SBERT models adapted to the languages Hindi, Marathi, Kannada, Telugu, Malayalam, Tamil, Gujarati, Odia, Bengali, and Punjabi, which is a mapping of semantically similar sentences to similar vector representations in an overlapping space. It has been shown experimentally that this fine-tuned SBERT method is both much more accurate than baseline multilingual representations (such as LaBSE and LASER) at cross-language and monolingual sentence similarity tasks, and is also comparable to monolingual SBERT models. The authors also publish monolingual SBERT variants in each of the languages and measure the performance in terms of embedding similarity scores and classification accuracy, which prove the efficiency of the synthetic corpus fine-tuning. The work has contributed to semantic evaluation and representation learning in low-resource languages and it is a useful resource to tasks that need meaningful sentence representations including clustering, semantic similarity, and retrieval in multilingual or Bengali NLP resources.

Kabir et al. (2024) describe BanglaEmbed: Efficient Sentence Embedding Models to a Low-Resource Language using Cross-Lingual Distillation Techniques, in which two lightweight sentence transformers are created that are specifically trained on the Bangali language to solve the issue of effective sentence embedding models in low-resource languages. The authors determine that high quality sentence embedding models of languages like English are not easily transferred to Bangali owing to the scarcity of data and resources; traditional multilingual models tend to perform poorly because they are not language-specific in semantic subtlety. To address this, they exploit a cross-lingual knowledge distillation model that depends on parallel English-Bangla machine translations information and a teacher student training system, where a high-performing English sentence transformer (teacher) is used to advise the training of two Bangla student models, called BanglaEmbed-MSE and BanglaEmbed-MNR. Such distillation brings the Bangali sentence embeddings and the respective English analogies into a shared semantic space, without using a large corpus of Bangali, implying that the approach is suitable in low-resource conditions. The two models have different loss functions, mean squared error (MSE) and multiple negative ranking (MNR) to optimize the quality of embedding. Comparison with paraphrase detection, semantic textual similarity (STS), and hate speech clustering tasks demonstrate that the proposed models, particularly BanglaEmbed-MSE, are superior to the existing Bangla sentence transformers, are less weighty and more effective, their inference time is smaller, and the performance indicators are comparable. The paper shows that cross-lingual distillation can produce high-quality sentence embeddings on the languages such as Bangla and others, and this study is valuable as an important methodology to the downstream semantic tasks and enables increased semantic assessment and retrieval when using the existing systems of semantic evaluation and retrieval in Bangla NLP systems.

Zhang et al (2019) introduce BERTScore: Text Generation Evaluation with BERT Text generation evaluation Text generation evaluation is an assessment method suggesting a new metric of evaluation of text generation tasks, based on contextual embeddings of pretrained language models rather than on surface-level overlap of tokens. The classical metrics such as BLEU, ROUGE and METEOR compare generated text to references using the exact matching of n-grams which does not always reflect semantic similarity particularly where other word selections can convey the same meaning. BERTScore fills this gap by matching tokens in the candidate and reference texts with contextual embeddings of BERT-like models and calculating a similarity score based on the cosine similarity of token embeddings. Instead of only considering the exact matches, the algorithm uses each token in one text and finds the best match in the other text to obtain the precision, recall, and F1 scores, thus identifying more in-depth semantic correlations. Different experiments and tasks in the teaching of text generation (translation using the machine, summarization and captioning) indicated that BERTScore is more strongly correlated with scores made by humans than conventional measures particularly in languages that exhibit high paraphrasing or expression differences. Another idea the authors also reveal is that the evaluation performance is affected by the use of various pretrained models (RoBERTa or XLNet) which implies that the sensitivity and granularity of semantic comparison depend on the underlying embedding model used. It has since been used as a popular semantic evaluation measure in NLG studies as it is more sensitive to meaning preservation and content relevance, and is a relevant reference when evaluating portions of summarization systems that are sensitive to semantic fidelity and coherence. This will help you to prepare your semantic evaluation of summaries and will emphasize the weakness of lexical measures over embedding based measures.

Zhong(2022), introduce UniEval: A Unified Multi-Dimensional Evaluator of Text Generation, that suggests a comprehensive evaluation framework that is aimed at testing generated text on a number of levels, including coherence, relevance, consistency, fluency, and factuality, instead of using single or superficial measures. Since the authors acknowledge the shortcomings of the conventional automatic evaluation systems such as BLEU and ROUGE that are largely unrelated to the actual human judgments, particularly during the summarization or dialogue generation tasks, they propose a single evaluator that would be much closer to the way humans perceive the quality of the text. UniEval does this by training a multi-task scoring model which co-learns a number of quality dimensions with the backbone being language models pretrained on language and with each evaluation criterion having specialized task heads. The model is trained on a wide range of annotated data that includes different text generation tasks giving it the capacity to be more general across domains. The authors have performed large experiments on the standard benchmarking that proves that UniEval is more correlated when compared to the existing single-dimension metrics as well as any other learned evaluator. The proposed structure also facilitates dimension-specific diagnostics, whereby the researchers can know which attribute of quality is causing the performance differences among the models e.g. the coherence versus factual consistency. Such a multi-dimensional approach is especially pertinent when the process involves the evaluation of summarization where various aspects of quality are important at the same time. The structure and outputs of UniEval indicate

that a single and trained judge should produce more human-centric and coherent assessments than other metrics and present a solid foundation into future studies of automatic, semantic and multi-facet assessment of text generation systems.

Scialom et al (2021) present QuestEval: Summarization Asks for Fact-based Evaluation, a reference-less automatic evaluation metric designed to better correlate with human judgments across multiple quality dimensions for text summarization without relying on ground-truth reference summaries. Traditional automatic metrics like ROUGE and BERTScore measure surface-level overlap between generated summaries and human references, which often fails to capture deeper semantic or factual quality, while other QA-based methods had not consistently outperformed ROUGE prior to this work. QuestEval instead directly assesses a summary by generating and answering questions based on both the source document and the candidate summary, combining precision and recall–based QA signals to evaluate whether essential information from the source is both present and accurately represented in the summary. Extensive experiments on standard summarization benchmarks show that QuestEval has significantly higher correlation with human judgment on key qualitative measures (consistency, coherence, fluency and relevance) than reference-based metrics, which makes it a more reliable substitute for human evaluation in many summarization scenarios. Moreover, the framework's reference-less property makes it applicable when human references are not present, which is a frequent limitation of evaluation approach. Furthermore, QuestEval's question-answering manner introduces a semantic, fact-based basis for the decision on summarization quality by judging the faithfulness of summary to source content rather than based purely on lexical matching. This makes QuestEval a useful metric for those who focus on semantic evaluation and factuality during model evaluations.

Asking and Answering Questions to guarantee Factual Consistency of Abstractive Summaries (QAGS) Chien et al (2020) was an automatic evaluation procedure, proposed by Wang et al. (2020), to address a bottleneck problem of abstractive summarization which is factual consistency present in generated summaries but never show up on traditional ROUGE or BLEU metrics. The primary concept of QAGS is based on the approaches of question answering (QA) and question generation (QG): when a summary contains the information presented in the original document, questions asked in relation to the summary are supposed to be responded to by the questions asked in the text. QAGS follows three steps: it is an automatic question generator that builds questions on the basis of the candidate summary, applies QA models to these questions individually on the summary and on the original document, and then compares the pairs of answers using similarity metrics to determine factual congruence. The result of this process is a factual consistency score which has much stronger correlation with human judgments than classical overlap-based measures, especially on benchmark datasets like CNN/DailyMail and XSum. Significantly, interpretability is also provided by QAGS since it states specific tokens or content portions which are inconsistent, thereby enabling diagnostic conclusions on the errors of summarization. The authors test QAGS on the compatibility with human ratings of the factual consistency, and also show its usefulness as a more semantically-based assessment of

summarization systems. This protocol is especially applicable to the studies of the semantic evaluation and factuality analysis of text summarization, which provides a framework that extends beyond surface matches to evaluate whether generated summaries are a meaningful capture of underlying source information- one of the factors to consider when developing trusted NLP systems.

Raihan et al. (2024) introduce *A Family of Bangla Large Language Models*, presenting a systematic effort to develop and analyze large language models specifically tailored for the Bangla language. Motivated by the limited coverage of Bangla in existing multilingual LLMs, the authors train multiple Bangla-focused models across different parameter scales using curated Bangla corpora. Their work emphasizes linguistic adequacy, vocabulary coverage, and downstream task performance rather than relying solely on cross-lingual transfer. The paper tests our proposed Bangla LLM family out on a variety of natural language understanding and generation tasks, including text classification, question-answering and text generation. We find that Bangla-specific pretraining results in outputs which are more stable and linguistically coherent compared to multilingual baselines, especially for tasks which are more sensitive to syntax and generation. The authors additionally investigate scaling, finding that larger models are better but with diminishing returns on some tasks. This work brings to attention the significance of language-specific model development for low-resource languages, and empirically demonstrates that monolingual or language-tuned LLMs can outperform multilingual models under Bangla-centric conditions. The results encourage investigating native Bangla architectures and also serves as a basis for measuring the quality of Bangla summarization and generation beyond surface fluency.

Zehady et al. (2024) introduce BanglaLLaMA, a version of LLaMA for Bangla language modelling. In this paper, we investigate the possibility of scaling up the strong decoder-only large language model (except training from initial checkpoint) language models to Bangla by further pretraining and instruction tuning on Bangla specific corpora. Unlike the encoder-decoder style architectures typically used for summarization, BanglaLLaMA is designed with a focus on autoregressive generation and open-ended text modeling. We evaluate BanglaLLaMA on various Bangla NLP tasks, such as text generation, question answering and instruction-following scenarios. Results suggest that BanglaLLaMA performs better in terms of fluency and lexical appropriateness in comparison to zeroshot multilingual LLaMA baselines. There is some variability in the model's performance with many factual consistency and abstraction-heavy tasks, though this is expected due to the common deficiencies of decoder-only models in low-resource settings. In this work, we show that further pretraining of large decoder-only models is able to offer significant improvements in Bangla language coverage and uncover problems such as hallucination and extractive bias. For BanglaLlMA, we contribute by providing a valuable benchmark for the Bangla LLM development that could guide future comparative studies on decoder-only and encoder-decoder models in Bangla summarization and evaluation pipelines.

2.3 Summary of Key Findings

In this article, we dissect various individual studies through which we summarize the major results and patterns that appear, putting emphasis on the collective information that they reveal about using LLMs for Bangla and cross-domain summarization.

Table 2.1: Summary Table of Important Papers

SL	Papers	Authors	Year	Method/Dataset	Key Findings
1	**BanglaNLG and BanglaT5: Benchmarks and Resources for Evaluating Low-Resource Natural Language Generation in Bangla**	Bhattacharjee et al.	2023	Six-task benchmark (BanglaNLG), Pretrained BanglaT5 model (T5-based), compared BanglaT5 with 5 multilingual models. Datasets: BanglaNMT, XL-Sum, BQA, DailyDialog (translated), CrossSum	BanglaT5 outperforms multilingual models and is efficient & strong for low-resource settings.
2	**CrossSum: Beyond English-Centric Cross-Lingual Summarization for 1,500+ Language Pairs**	Bhattacharjee et al.	2023	Cross-lingual retrieval from XL-Sum, multistage language sampling (MLS), LaSE metric. Dataset: CrossSum (1.68M samples, 1,500+ language pairs)	CrossSum is not centered on English; LaSE effectively evaluates low-resource summaries; models trained on CrossSum outperform baselines.
3	**Zero-Shot Cross-Lingual Summarization via Large Language Models**	Lin et al.	2023	LLM-based zero-shot summarization (e.g., mT5, GPT). Datasets: English + Bengali articles	Demonstrated feasibility of Bangla summarization without native training using multilingual LLMs.
4	**Evaluating Large Language Models for Summarizing Bangla Texts**	Rony et al.	2025	LLMs (GPT-3.5, GPT-4, PaLM-2) for abstractive cross-lingual summarization (EN→X, X→EN, X→X); zero-shot and few-shot prompts. Datasets: CrossSum (news/summaries, 12 languages)	GPT-4 and PaLM-2 outperform mT5/mBART50 in zero- and few-shot settings for Bangla; GPT-4 generates fluent, abstractive summaries for low-resource languages.
5	**Think Carefully and Check Again! Meta-Generation Unlocking LLMs for Low-Resource Cross-Lingual Summarization**	Li et al.	2025	Four-step zero-shot method (SITR: Summarization, Improvement, Translation, Refinement) for LLMs; meta-generation, no task-specific fine-tuning. Datasets: CrossSum, WikiLingua	SITR enables high-quality summarization in low-resource languages, state-of-the-art ROUGE or BERTScore, robust across LLMs.

SL	Papers	Authors	Year	Method/Dataset	Key Findings
6	**CONVERSUM: A Contrastive Learning Based Approach for Data-Scarce Solution of Cross-Lingual Summarization Beyond Direct Equivalents**	Lora et al.	2023	Contrastive learning with candidate summary generation via seq2seq (mT5, FLAN-T5), contrastive ranking loss, evaluated with LaSE/BERTScore. Datasets: CNN/DailyMail, XL-SUM (Bengali, Thai, Burmese, Tigrinya), CrossSum	Outperforms baseline models for low-resource and cross-lingual settings, effective in low-data, beats LLMs in some low-resource CLS tasks.
7	**Low-Resource Cross-Lingual Summarization through Few-Shot Learning with Large Language Models**	Park et al.	2024	Few-shot learning for XLS using GPT-3.5, GPT-4, Mistral-7B-Instruct-v0.2; in-context prompting; end-to-end approach. Dataset: CrossSum	Improves multilingual summary quality by removing pivot languages, enhances coherence/robustness in low-resource, higher computational cost.
8	**A Deep Reinforced Model for Abstractive Summarization**	Paulus et al.	2017	Encoder-decoder with attentional mechanisms; intra-decoder attention; supervised + reinforcement learning. Datasets: CNN/Daily Mail, New York Times	Intra-decoder attention improves longer summaries, RL enhances readability/diversity, achieved SOTA ROUGE-1.
9	**Fast Abstractive Summarization with Reinforce-Selected Sentence Rewriting**	Chen et al.	2018	Two-stage hybrid: RL-based extractor + abstractive abstractor (seq2seq, attention, copy). Datasets: CNN/Daily Mail, DUC-2002	Hybrid model bridges extractive/abstractive; more abstractive, fewer repetitions, 10-20x faster inference, new SOTA.
10	**Multi-Target Cross-Lingual Summarization: a novel task and a language-neutral approach**	Pernes et al.	2024	MTXLS framework: multi-target CLS, clustering, language-neutral re-ranking (NeutralRR, PivotRR), semantic coherence metrics. Dataset: CrossSum (7 languages, clusters for MTXLS)	Boosts semantic coherence/trust in multilingual summaries, removes pivot language, improves quality but increases computational cost.
11	**G-Eval: NLG Evaluation using GPT-4 with Better Human Alignment**	Liu et al.	2023	LLM-based evaluation framework using GPT-4 as an automatic judge; prompt-guided scoring across NLG tasks	GPT-4-based evaluation shows strong correlation with human judgments and outperforms traditional metrics in assessing semantic quality.

SL	Papers	Authors	Year	Method/Dataset	Key Findings
12	**Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena**	Zheng et al.	2023	LLM-as-a-Judge evaluation using MT-Bench (multi-turn QA) and Chatbot Arena (crowdsourced human preference comparisons)	Achieving over 80% agreement with human preferences. Identifies major evaluation biases and shows that benchmark design and blind comparison are essential for fair LLM-based evaluation.
13	**Length-Controlled AlpacaEval**	Li et al.	2024	Pairwise LLM-as-a-Judge evaluation with verbosity control; preference-based scoring	Length-controlled evaluation reduces verbosity bias and improves robustness of LLM-based automatic evaluation.
14	**Evaluating the Factual Consistency of Abstractive Text Summarization**	Kryściński et al.	2020	Weakly supervised factual consistency classifier (FactCC) using synthetic data generation	Effectively detects hallucinations and factual errors in abstractive summaries without human annotation.
15	**L3Cube-IndicSBERT**	Deode et al.	2023	Sentence-BERT fine-tuning for Indic languages using multilingual BERT	Improves semantic similarity representation for low-resource Indic languages, supporting semantic evaluation tasks.
16	**BanglaEmbed: Efficient Sentence Embedding Models for Bangla**	Kabir et al.	2024	Cross-lingual distillation to build Bangla sentence embeddings	Outperforms multilingual baselines on Bangla STS tasks and enables reliable semantic evaluation for Bangla NLP.
17	**BERTScore: Evaluating Text Generation with BERT**	Zhang et al.	2020	Contextual embedding similarity using BERT; reference-based evaluation	Demonstrates stronger alignment with human judgments than lexical overlap metrics such as ROUGE.
18	**UniEval: Towards a Unified Multi-Dimensional Evaluator for Text Generation**	Zhong et al.	2022	QA-based unified evaluator covering coherence, relevance, fluency, and consistency	Provides multi-dimensional evaluation with high agreement to human assessment across generation tasks.

SL	Papers	Authors	Year	Method/Dataset	Key Findings
19	**Semantic Evaluation of Summarization Beyond ROUGE**	Scialom et al.	2021	QA-based and reference-free semantic evaluation methods	Highlights limitations of ROUGE and promotes semantic metrics for abstractive summarization.
20	**Asking and Answering Questions to Evaluate the Factual Consistency of Summaries**	Wang et al.	2020	QA-based factual consistency evaluation (QAGS)	Better captures faithfulness errors and correlates more strongly with human judgments than ROUGE.
21	**A Family of Bangla Large Language Models**	Raihan et al.	2025	Bangla-specific large language models trained at multiple scales; evaluated on Bangla NLU and NLG tasks including text generation and classification.	Demonstrates that Bangla-native LLMs outperform multilingual baselines in linguistic coherence and stability, highlighting the importance of language-specific pretraining for low-resource languages.
22	**BanglaLLaMA: LLaMA for Bangla Language**	Zehady et al.	2024	Decoder-only LLaMA architecture adapted for Bangla via continued pretraining and instruction tuning; evaluated on Bangla generation and instruction-following tasks.	Improves fluency and lexical accuracy over zero-shot multilingual LLaMA, but exhibits extractive bias and hallucination, indicating limitations of decoder-only models for abstractive Bangla summarization.

Conclusion: The reviewed works indicate that Bangla text summarization has undergone important progression from the common extractive and pipeline based methods to state- of-the-art neural and large language model techniques. The introduction of multilingual resources such as CrossSum and benchmark models such as BanglaT5 has expanded the research horizon allowing better cross-domain and cross-lingual summarization. Yet, some longstanding problems persist, namely the lack of highquality annotated data, difficulties in evaluating abstractive summaries and handling factual consistency across domains and languages. The recent developments in LLMs, specially zero-shot and few-shot learning abilities, have demonstrated potential for low-resource settings but tuning these models for Bangla has yet to be explored. Together, these conclusions highlight both successes and continuing challenges that drive present and future research in Bangla text summarization.

Chapter 3

Requirements, Impacts and Constraints

3.1 System Requirements and Specifications

This chapter provides the goals, system demands, evaluation methodology and practical difficulties of our analysis. The paper is organized based on the **controlled experimental case study** used to explore Large Language Models (LLMs) doing **cross-domain Bangla text summarization** under a well-defined zero-shot setting. Instead of tuning a summary system to perform best in any particular domain, the focus is on studying how different models behave in comparison with respect to robustness and semantic consistency level. This kind of perspective is crucial for low resource languages, such as Bangla where evaluation noise, dataset scarcity and linguistic variation can make the differences between model families less meaningful.

3.1.1 Task Definition

The task addressed in this study the problem considered is to produce compact Bangla summaries from diverse Bangla source documents. All models are tested under the same zero-shot prompts without fine-tuning on task-specifics, in-context example or domain adaptation. By restricting the task in this way, the study teases out the natural generalization ability of each model. This design decision makes it possible to realistically evaluate summarization performance in deployed settings, where labeled Bangla resources, curated prompts or domain-specific fine-tuning are not available. The main objective of the method is to understand if, and to what extent, current LLMs can support transferable summarization capacities across Bangla domains and effects of architecture and training variations on abstraction quality in low-resource scenarios.

3.1.2 Experimental Setting

The models are assessed with an uniform prompt-like format, decoding settings and generation constraints to enable comparability. Nine Large Language Models are chosen to cover various architectural paradigms **encoder–decoder and decoder-only**, training protocols **Bangla-specific, multilingual, and general-purpose**, as well as parameter sizes. This choice of model is intended to be representative

of a range of behaviour around that displayed by contemporary LLM while still being computationally realistic for the researcher. The use of Bangla-native models in multilingual and decoder-only systems allows specific investigation into language-specific pretraining effects as well as syntactic limitations including extractivity bias and hallucination.

The experimental system is required to satisfy the following functional properties:

- Accept Bangla (Unicode-compliant) documents as input.
- Generate Bangla summaries exclusively via zero-shot prompting.
- Support evaluation across multiple datasets and content domains.
- Enforce identical inference and evaluation conditions across all models.
- Produce outputs suitable for both quantitative scoring and qualitative inspection.

In addition to functional correctness, the system must satisfy several non-functional requirements to ensure scientific rigor:

- **Reproducibility:** Fixed prompts, decoding parameters, and evaluation protocols.
- **Comparability:** Uniform evaluation conditions across models and datasets.
- **Scalability:** Efficient evaluation of multiple models and documents.
- **Traceability:** Explicit linkage between model outputs, evaluation scores, and reported findings.

Together, these requirements prioritize methodological integrity over absolute performance maximization and support robust comparative analysis.

3.1.3 Computational and Software Requirements

All experiments were performed on a commodity computer cluster. The hardware and software environment was designed to enable multi-model inference and semantic interpretation, while remaining realistic in an academic research context. No distributed training or cloud-based deployment is necessary, since we consider only inference-time assessment in the study.

Hardware Configuration

The hardware configuration used for model inference and evaluation is summarized in Table 3.1.

Table 3.1: Hardware specifications used for experimentation

Component	Specification
CPU	AMD Ryzen 9 5950X (16 cores, 32 threads)
GPU	NVIDIA RTX 3080 Ti (12 GB VRAM)
RAM	64 GB DDR4
Storage	1 TB SSD
Power Supply	850 W PSU

This model design also makes zero-shot inference, document level batching and embedding based evaluation hardwarefree across models.

Software Environment

The experimental pipeline is implemented using widely adopted machine learning and numerical computing libraries. All models are executed within the same software environment to guarantee fairness and reproducibility. A summary of the software stack is provided in Table 3.2.

Table 3.2: Software tools and libraries used in the study

Component	Description
Operating System	Linux-based environment
Deep Learning Framework	PyTorch
Numerical Computing	NumPy
CUDA Toolkit	CUDA 11.8
Supporting Libraries	Tokenization, embedding, and evaluation utilities

Model-specific fine-tuning is deliberately excluded to preserve the integrity of the zero-shot evaluation framework.

3.2 Evaluation Constraints and Design Considerations

Evaluation design in summarization is a critical choice, especially for low-resource languages where stylistic varia- tion, paraphrasing and reference quality are significant challenges. This section anno-tates the methodological limitations that determined metric choice and provides motivation for supplementary semantic eval-uation methods.

3.2.1 Limitations of Lexical Overlap Metrics

Lexical overlap metrics such as ROUGE are commonly employed in summarization evaluation. However, preliminary experiments reveal that ROUGE scores for Bangla abstractive summarization are consistently near zero across all evaluated models and

domains. This behavior arises from Bangla's morphological richness and the high degree of paraphrasing produced by modern LLMs. As a result, ROUGE fails to capture semantic equivalence and meaning preservation in this context. Consequently, lexical overlap metrics are included only as baseline diagnostics and are not treated as primary indicators of summarization quality.

3.2.2 Semantic Evaluation Using LLMs as Judges

To overcome the limitations of lexical metrics, this study incorporates **LLM-as-a-Judge** evaluation. Two independent pretrained models (ChatGPT and Gemini) are employed to assess generated summaries based on semantic coverage, coherence, and overall quality. Using multiple judges mitigates idiosyncratic scoring behavior and aligns with recent findings that ensemble-style evaluation improves robustness. LLM judgments are interpreted as informed semantic signals rather than absolute ground truth and are complemented by additional evaluation paradigms.

3.2.3 Embedding-Based Similarity Analysis

Besides the judge-based assessment measure, we also define cosine similarity between sentence embeddings as principle of semantic similarity. Three pretrained SBERT-based models are used to evaluate semantic alignment and intermodel agreement of the summaries derived from the same source document. This reference-free methodology offers insight into abstraction, extractiveness, and convergence behavior across model families. This relatively high inter-model similarity can be an indicator of template or extractive response in comparison to lower similarity values, which may represent more abstraction and stylistic variety.

Table 3.3: Evaluation methods used in the study

Method	**Purpose**
ROUGE (baseline)	Illustrate limitations of lexical overlap metrics
LLM-as-a-Judge	Assess semantic quality and coherence
SBERT Similarity	Measure semantic consistency across models

3.3 Societal Impact

Bangla is one of the most spoken languages in the world but it is severely under-represented in state-of-the-art natural language processing research and development. This bias obstructs the Bangla-speaking to access information technologies in an equitable way. Robust summarization methods in Bangla will improve the information access for various fields like education, journalism, governance and digital media. It can reduce cognitive load, aid in understanding of the information, and better support dissemination to a larger audience in knowledge societies. But of course, there will be costs associated with these positives. Errors or partial summaries can lead to misinformation and parts of the message, particularly in sensitive areas, may be missing. These are all reasons why robust assessment frameworks and responsible deployment is so important.

3.4 Ethical and Environmental Considerations

3.4.1 Ethical Considerations

The function of LLMs as both generative language model and evaluator raises ethical concerns regarding subjectivity and bias. To mitigate this possibility, we employ multiple raters and heterogenous paradigms of evaluation to avoid reliance on any particular score or model in this study. All datasets used are publicly available, and were used under an academic license. The project is an analysis rather than a use, hence ethics are limited to the interpretation of the experiment.

3.4.2 Environmental Impact

From an environmental perspective, the computational footprint of huge language models is something that we should be increasingly worried about. This paper deliberately confines experimentation to inference-time performance with zero-shot prompting, not involving energy-costly training or fine-tuning. By means of re-utilizing produced summaries across evaluation stages, the work reduces the need for redundant computation and supports insightful comparison, which is inline with guidelines in sustainable ML research.

3.5 Project Constraints and Risk Analysis

3.5.1 Technical Constraints

Some technical limitations determine both conduct and analysis of this study. First, most of the benchmarked Large Language Models are proprietary systems and little is known about their underlying architecture, training data, or optimization method. Consequently, the analysis is necessarily from a black-box perspective of evaluation: it only considers observable input-output behavior of the model rather than its internal mechanisms.

Next, the evaluation presented such LLM-based methods may be variable themselves. Ratings may vary based on variations in decoding noise, prompt sensitivity, or idiosyncratic rater behaviors. While such variability does not preclude comparative analyses, it places bounds on the interpretability of absolute-rated scores and requires stringent experimental control.

Model sensitivity Finally, embedding-based similarity metrics intrinsically depend on the choice of the embedding model. Different versions of SBERT might focus on different semantic dimensions, which might affect the distribution of similarities and inter-model com-parisons. This fragility constrains the degree to which any given embedding model can be used as a ground truth for semantics..

3.5.2 Risk Mitigation Strategies

To address these limitations, the analysis gives precedence to the trends of comparisons than to the absolute values of performance. Model behavior is contrasted

across a variety of evaluators, datasets and semantic representations in order to find consistent patterns rather than ranking it via single point scores. Multiple LLM judges and embedding models are used to triangulate results, and decreases reliance on a single judge or similarity space. All prompts, decoding parameters, datasets and evaluation procedures have been explicitly provided for the sake of reproducibility. In some cases, qualitative examination and error taxonomy analysis also complement quantitative results by revealing systematic failure modes which the scalar metrics alone cannot reveal. With these mitigation measures, the study aims to reconcile methodological stringency and operational constraints, typical of evaluation of large-scale LLM.

3.6 Economic Considerations

The main economic cost involved in this research is the use of APIs for LLM inference and LLM-as-a-Judge evaluation. By limiting the experimental design to zero-shot inference, and by reusing generated summaries across all evaluation steps we closely manage those costs and limit the amount of infinite model invocations.
No search or training as part of fine-tuning is conducted during this method; thus, the computational cost and infrastructure demand are significantly decreased. So the whole project remains realizable within standard academic resource limitations while still supporting full multi-model, multi-metric evaluation. This cost-effective scheme improves the feasibility and reproducibility of our evaluation framework.

Chapter 4

Methodology and Implementation

4.1 Methodology Overview

This study adopts a controlled, evaluation-centric experimental methodology designed to systematically analyze the zero-shot Bangla text summarization capabilities of large language models across heterogeneous domains. Rather than proposing or training a new summarization architecture, the methodology is explicitly structured to ensure fair comparison, reproducibility, and interpretability of model behavior under identical conditions. The primary focus is on understanding how different model families behave when generating Bangla summaries and how their outputs can be evaluated reliably when traditional lexical metrics fail.

The methodological pipeline is organized into five sequential stages: dataset preparation, model selection, zero-shot generation, multi-layered evaluation, and analytical interpretation. Each stage is carefully constrained to minimize confounding factors and isolate the intrinsic generalization ability of the models.

First, two complementary datasets are selected to represent different summarization conditions. The Prothom Alo dataset consists of real-world Bangla news articles with naturally written summaries, capturing practical domain characteristics such as stylistic variability and editorial inconsistency. In contrast, the XL-Sum (Bangla) dataset provides benchmark-style article–summary pairs with more standardized summary structure. This dual-dataset setup enables the assessment of cross-domain robustness without introducing additional task-specific adaptation.

Second, a diverse set of nine large language models is selected to capture architectural and linguistic variation. These include encoder–decoder models optimized for Bangla or multilingual generation, as well as decoder-only models marketed for multilingual or Bangla-centric use. All models are treated as black-box generators, and no fine-tuning, parameter updates, or domain adaptation is performed. This ensures that performance differences arise from model architecture and pretraining rather than task-specific optimization.

Third, all models operate under a strictly zero-shot summarization protocol. A single, uniform prompt template is applied across all models and datasets, with fixed generation parameters to ensure comparability. No in-context examples,

prompt engineering variants, or adaptive decoding strategies are introduced. This design choice isolates inherent model capabilities and avoids inflating performance through prompt sensitivity.

Fourth, evaluation is conducted using a multi-stage framework that combines complementary perspectives on summary quality. Lexical overlap metrics (ROUGE and BLEU) are computed to establish baseline behavior, despite their known limitations for Bangla. These are supplemented by embedding-based semantic similarity analysis using SBERT to quantify semantic alignment between generated summaries, reference summaries, and source articles. In parallel, LLM-as-a-Judge evaluation is employed under controlled and blind conditions to capture qualitative judgments related to coherence, faithfulness, and abstraction. Finally, an automated error taxonomy is applied to systematically categorize dominant failure modes such as hallucination, unsupported detail insertion, and excessive extraction.

Finally, the outputs of all evaluation stages are jointly analyzed to triangulate findings and avoid over-reliance on any single metric or evaluator. Quantitative results are interpreted alongside qualitative case studies to provide a behavior-aware understanding of model performance. This methodology enables robust comparison across model families and datasets while maintaining transparency, interpretability, and reproducibility.

4.2 Datasets and Preprocessing

4.2.1 Data Collection

We want to state that the datasets that we used, were only collected from publicly available and trustworthy (NLP) repositories having work on Bangla and cross lingual summarization model. As the objective of this study is to develop and evaluate summarization models that can handle both Bangla and bilingual (Bangla-English) content, we selected datasets focusing on linguistic quality as well as topical coverage and diversity

No raw data scraped from the web or collected manually was used. Instead, all datasets were downloaded from open research resources **Hugging Face Datasets** and **Kaggle**, where they have pre-processed article-summary aligned pairs. This promoted ethics-focused data use, transparency and replicability.

Sources of Data

Table 4.1: Overview of datasets used for Bangla and cross-lingual summarization.

Dataset	**Origin/Source**	**Language**	**Nature**
XL–Sum (Bangla subset)	Hugging Face – `bbc_xlsum`	Bangla	News articles with human-written summaries

Dataset	Origin/Source	Language	Nature
CrossSum (English–Bangla & Bangla–English)	Hugging Face	English ↔ Bangla	Parallel articles and summaries across multiple languages

Data Collection Method

Datasets were downloaded using both the **Kaggle API** and the **Hugging Face Datasets** library. For larger archives (e.g., CrossSum), manual extraction from compressed files was performed after download. Each dataset was stored in an organized directory hierarchy to support reproducibility.

Data Verification and Validation

To ensure integrity and research compliance, the following validation steps were performed:

- File sizes and checksums were compared with repository metadata where available to confirm completeness.
- All text resources were converted to UTF-8 to ensure correct Bangla rendering.
- Article–summary pairs were checked to confirm valid fields and readable content.
- Datasets were verified to be publicly available and usable for academic research, and checked to avoid personally identifiable information.

4.2.2 Standardization and Storage Format

All datasets were standardized into JSON Lines (JSONL) format with two core fields: `article` and `summary`. This format ensures compatibility with tokenizers, data loaders, and evaluation pipelines.

```
{"article": "...", "summary": "..."}
```

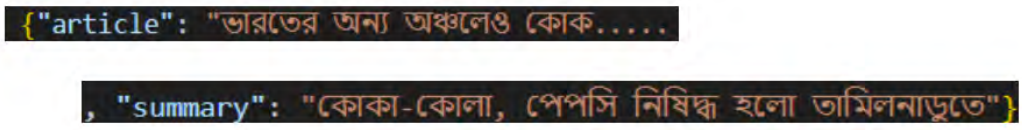


Figure 4.1: Example of a JSONL record containing `article` and `summary` fields.

4.2.3 Data Cleaning

A consistent cleaning pipeline was used on all datasets to ensure the quality. The pipeline discards incomplete records, outliers of extreme length, standardized Bangla Unicode rendering and prune noise down (such as HTML/Garbage tokens) and tuning duplicates using SHA-256 hash.

Handling Missing and Incomplete Values

All data-sets were validated so that all entries had an article and corresponding summary. Records lacking one of these two fields were eliminated. Entries consisting only of whitespace, control characters, or corrupted characters were removed. All text was stored in UTF-8.

Handling Outliers (Length Filtering)

To balance informativeness and stability, character-length thresholds were applied:

- Articles: 80–6000 characters
- Summaries: 15–1000 characters

Text Normalization

The unicodedata library was used for applying Unicode normalization (NFC). Zero-width spaces, HTML tags and Non-language characters are filtered out. Punctuation (spaces before and after the Bangla danda) was regularized.

Duplicate Detection and Removal

SHA-256 hashes were calculated for the de-duplicated article text and a global hash set was employed in order to detect and remove duplicates across all datasets.

Error Correction and Validation

The readability, Bangla rendering and uniform formatting of the schema were reviewed in random samples. Pre-processed datasets were exported in a common JSONL format.

4.2.4 Data Transformation, Integration, and Reduction

After cleaning, all datasets were transformed into a unified JSONL schema, validated for structural integrity, and prepared for optional integration. Data reduction strategies (global de-duplication, length filtering, and optional proportional sampling) were applied to reduce redundancy while maintaining diversity.

4.3 Model Description

This study evaluates **compact (small-scale) variants** of each model family to examine which smaller model size provides the best balance between semantic adequacy and robustness under practical constraints. All models were evaluated using a consistent **zero-shot** prompt.

4.3.1 Transformer Architectures

Encoder–Decoder Transformers

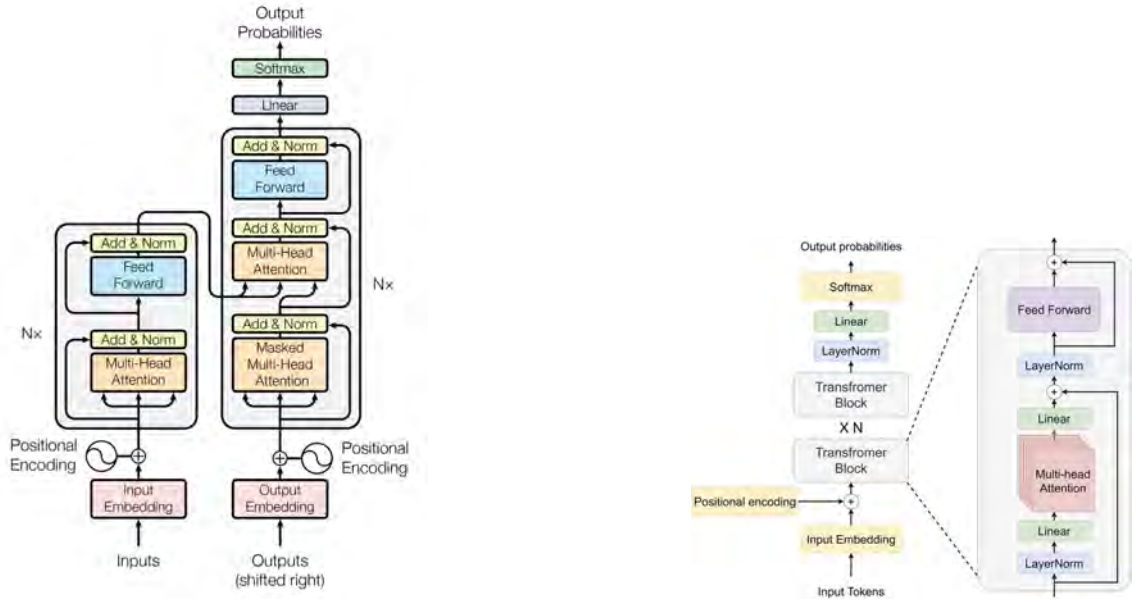
Encoder–decoder models perform conditional generation: the encoder encodes the input article into contextual representations, and the decoder generates the summary conditioned on those representations. This explicit mapping is typically suitable for summarization.

Decoder-Only Large Language Models

Decoder-only models generate text autoregressively by predicting the next token given previous context. For summarization, the article and instruction prompt are provided as context, and the model generates a summary based on instruction-following behavior and learned language representation.

4.3.2 Architectural Comparison

To clarify the structural differences between the two model families evaluated in this study, Figure 4.2 presents a side-by-side comparison of encoder–decoder and decoder-only transformer architectures. This visualization highlights the distinct information flow and generation mechanisms underlying each paradigm.



(a) Encoder–decoder transformer architecture.

(b) Decoder-only transformer architecture.

Figure 4.2: Side-by-side comparison of encoder–decoder and decoder-only transformer architectures used in this study.

4.3.3 Model Specification Table

Table 4.2: Model specifications and architectural characteristics (compact variants)

Model	Architecture	Parameters	Notes
BanglaT5	Encoder–decoder	247M	Bangla-focused summarization baseline
mT5	Encoder–decoder	582M	Multilingual summarization baseline
BLOOMZ	Decoder-only	1.7B	Multilingual instruction-tuned LLM baseline
LLaMA3.2	Decoder-only	1.23B	Fine-tuning attempt (compact variant)
Mistral7B	Decoder-only	7.3B	Decoder-only baseline; Bangla output issues observed
TigerLLM	Decoder-only	1B	Bangla-oriented compact LLM
TituLLM	Decoder-only	1B	Bangla-oriented compact LLM
Lunatic_bongLlama	Decoder-only	1.1B	Bangla-oriented compact LLM
BanglaLlama	Decoder-only	1B	Bangla-oriented LLM

4.4 Experimental Design

This section discusses the experimental methodology used for this thesis, which consists of the summary protocol and output collection and organization techniques. The design favors controlled comparison, reproducibility and fairness in which any improvement is attributed to model behavior and not experimental artifacts.

4.4.1 Zero-Shot Summarization Protocol

We evaluated all our models in a completely zero-shot summarization setting test condition where no in-context examples, demonstrations or task-specific fine-tuning were provided. All models were given the same Bangla prompt template and asked to generate short-form summary of a Bangla source document. We use this design to decompose the generalization capability of each model without potential confounds from prompt engineering or few-shot example selection.

Zero-shot is very significant in case of low-resource languages such as Bangla, where there is a lack of well-formed training data available and task adaptation sometimes inclines toward the phenomenon of visible robustness. Applying the same prompts and generation constraints to each of several models, the study allows for direct comparison of architectural options, training regimes and language specialization

given realistic deployment scenarios. Decoding settings were kept as similar as possible among models (whenever it was feasible given implementation constraints) to mitigate any generation bias.

4.4.2 Output Collection and Organization

All the summary outputs were accumulated as structured CSV files for further evaluation and analysis. For each record we include the source article, its respective reference summary (if available), and model-generated output under zero-shot conditions. This common format helps the community for fast blind evaluation, cross-model comparison, automatic scoring and visualization.

To facilitate blind LLM-as-a-Judge, we eliminated or anonymized model identities in the judging phase to prevent judges from being biased based on previous knowledge of models architectures or origins. We used the same output files for all evaluation methods (LLM-based judgment, SBERT cosine similarity) and automated error taxonomy. This reuse ensures that all evaluation methods work with exactly the same model output, thereby enhancing reproducibility, traceability and methodological soundness.

4.5 Evaluation Framework

4.5.1 Lexical and Embedding-Based Metrics

This study initially employed established automatic evaluation metrics—ROUGE, BLEU, and BERTScore—to quantify the overlap and similarity between generated summaries and reference summaries. For completeness and reproducibility, the definitions and scoring principles of these metrics are summarized below.

ROUGE

ROUGE measures overlap between a candidate summary C and a reference summary R. ROUGE-N is based on matching N-grams, while ROUGE-L is based on the Longest Common Subsequence (LCS).

ROUGE-N:

$$\text{ROUGE-N} = \frac{\sum_{g \in \mathcal{G}_N(R)} \min\Big(\text{Count}_C(g),\, \text{Count}_R(g)\Big)}{\sum_{g \in \mathcal{G}_N(R)} \text{Count}_R(g)} \tag{4.1}$$

where $\mathcal{G}_N(R)$ denotes the set of N-grams in R and $\text{Count}(\cdot)$ is the frequency of an N-gram.

ROUGE-L (LCS-based):

$$\text{ROUGE-L} = \frac{\text{LCS}(C, R)}{|R|} \tag{4.2}$$

where $\mathrm{LCS}(C, R)$ is the length of the longest common subsequence between C and R and $|R|$ is the reference length.

BLEU

BLEU is a precision-oriented metric using clipped n-gram counts and a brevity penalty. For N-gram order N, BLEU is:

$$\mathrm{BLEU} = \mathrm{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{4.3}$$

where p_n is the clipped n-gram precision, w_n are weights (often uniform), and the brevity penalty is:

$$\mathrm{BP} = \begin{cases} 1, & |C| \geq |R| \\ \exp\left(1 - \frac{|R|}{|C|}\right), & |C| < |R| \end{cases} \tag{4.4}$$

BERTScore

BERTScore computes similarity using contextual embeddings and reports Precision, Recall, and F1. For token embeddings $\{\mathbf{c}_i\}$ from C and $\{\mathbf{r}_j\}$ from R, with cosine similarity $\cos(\cdot, \cdot)$:

Precision:

$$P = \frac{1}{|C|} \sum_{i=1}^{|C|} \max_j \cos(\mathbf{c}_i, \mathbf{r}_j) \tag{4.5}$$

Recall:

$$R = \frac{1}{|R|} \sum_{j=1}^{|R|} \max_i \cos(\mathbf{r}_j, \mathbf{c}_i) \tag{4.6}$$

F1:

$$F1 = \frac{2PR}{P + R} \tag{4.7}$$

Why These Metrics Underperformed for Bangla

In Bangla summarization, morphological variation, compounding, flexible word order, and paraphrasing reduce literal token overlap. As a result, ROUGE and BLEU often collapse to near-zero even when the summary is semantically correct. BERTScore can also become unreliable when token alignment is distorted by language-specific segmentation and paraphrastic divergence. This motivated the transition to semantic evaluation using LLM-as-a-Judge and SBERT cosine similarity. The numerical results obtained from ROUGE, BLEU, and BERTScore are reported and analyzed in Chapter 5 to illustrate their empirical behavior across datasets and models.

4.5.2 Evaluation Framework Overview

The evaluation framework adopted in this study is intentionally multi-layered and diagnostic in nature, designed to overcome the limitations of any single evaluation paradigm when applied to low-resource, highly abstractive Bangla summarization. Rather than relying on a single metric or evaluator, the framework integrates complementary evaluation signals that jointly assess semantic quality, abstraction behavior, and systematic failure modes.

Figure 4.3 presents an overview of the complete evaluation pipeline used in this work. The framework is structured as a staged process in which model outputs are progressively examined using increasingly semantic- and error-aware evaluation mechanisms.

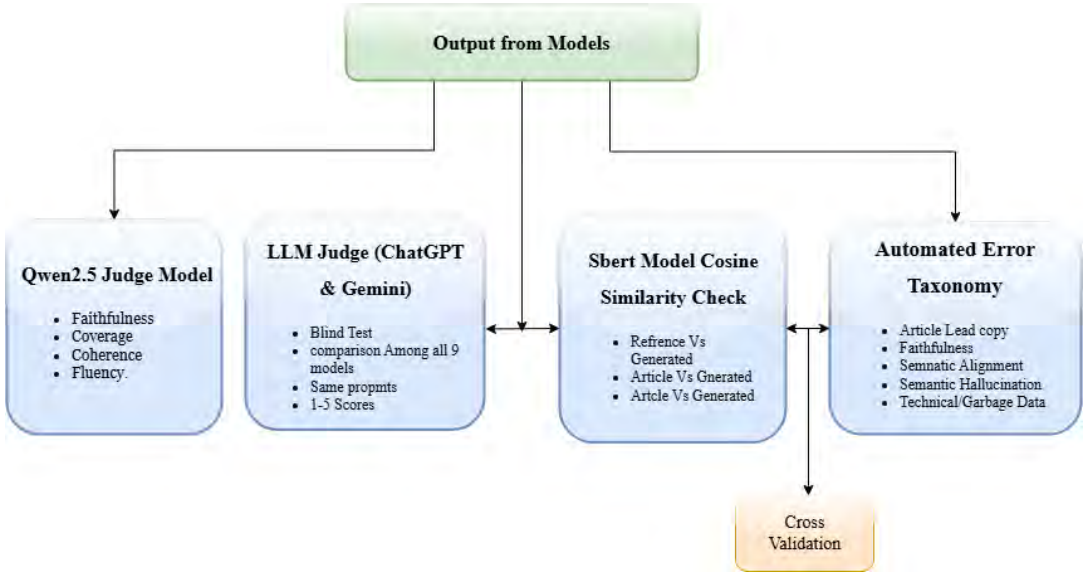


Figure 4.3: Overview of the multi-stage evaluation framework

The assessment is carried out in four main recurrent phases. To do this, it firstly evaluates LLM -as-a-Judge model based summaries (Qwen2. 5-1. 5B) which yields scalar values along a number of semantic axes (e.g., faithfulness, and coverage, coherence, and fluency). This step provides an interpretable semantic baseline and demonstrates the feasibility of judge-based evaluation for Bangla summarization, as conventional lexicalbased metrics are unproductive.

Second, in order to minimize the bias of the evaluator and enhance credibility of the results,**blind LLM-as-a-Judge evaluation** is performed blind by ChatGPT and Gemini. For this stage, model identitiesare anonymized and summaries are shown in random order, similar to the controlled human evaluation methodology followed in summarization literature.

Third, **embedding-based semantic similarity analysis** is performed with a collection of SBERT-family models. In this way, the stage gives a reference-free quantitative image of semantic alignment between generated summaries and either reference summaries or source articles and allows for examination of abstraction

versus extractive behavior across various model families.

Finally, an **automated error taxonomy** is used to classify and measure salient failure modes : article-lead copying, hallucination, faithfulness errors (faith),technical noise. This diagnostic layer offers explanative insight for why certain models obtain low judge scores or high similarity values, grounding semantic evaluations in interpretable and recurrent error patterns.

When conditioned on a judge-based scoring, embedding similarity and structured error analysis allow for much more robust triangulation of results. Consistency across these layers increases confidence in results, while inconsistency indicates bias of the evaluator or problems with the model. This architecture is intended to guarantee that any conclusion is not an artifact of a unique evaluation metric, but responds to coherent semantic trends from several complementary points of view.The following sections describe each component of the evaluation framework in detail, beginning with an analysis of why traditional lexical metrics fail for Bangla and motivating the transition toward semantic and judge-based evaluation.

Qwen2.5-1.5B: Judge Model Description: The initial LLM-as-a-Judge evaluation in this study employs **Qwen2.5-1.5B**, a decoder-only transformer-based large language model developed as part of the Qwen family. Qwen2.5-1.5B is designed for instruction-following, reasoning, and multilingual text generation, making it suitable for semantic judgment tasks such as summary evaluation.
Architecturally, Qwen2.5-1.5B follows a standard autoregressive transformer design, consisting of stacked self-attention and feed-forward layers with causal masking. The model is pretrained on a large-scale multilingual corpus and subsequently instruction-tuned to improve alignment with human preferences in tasks involving evaluation, comparison, and qualitative assessment. Unlike encoder–decoder summarization models, Qwen2.5 as a text-based evaluator only which does not produce summaries in this paper.

The less extended nature of Qwen2. 5-1. 5B is a good compromise in terms of the ability to perform semantic reasoning and computational load. This makes it especially appropriate for controlled academic experiments which require many iterations of evaluation on numerous summaries. Being instruction-tuned, it can follow well-structured evaluation prompts, and provide scalar judgments on various dimensions of quality in a consistent way.

Role of Qwen2.5 in the Evaluation Pipeline Qwen2.5-1.5B is utilized solely as an automated judge rather than a candidate summarization model. The system is provided with an input article, a generated summary and a fixed evaluation prompt: It scores for semantic faithfulness to the source text, coherence of the response as well as coverage and fluency. These scores are then aggregated and analyzed in the context of models with similar architecture. This first phase (evaluation of the five models) has a dual aim. First, it confirms the plausibility of LLM-based semantic evaluation in the context of Bangla summarization, something conventional metrics do not support. Second, it establishes interpretable benchmark performance that can be used to inform the next iteration of expanded evaluation (with a broader set

of models and adjudicators) as described below.

Extension with Additional Bangla-Oriented Models Extension with More Bangla-Oriented Models In addition to studying if the Bangla-specialized models could outperform multilingual baselines, four more Bangla-oriented LLMs (Tiger-LLM, TituLLM, Lunatic_bongLlama, and BanglaLlama) were presented. The models were evaluated with the **same zero-shot prompt** and assessed using **Qwen2.5 as the judge**, preserving consistency with its previous semantic analysis. It is to be noted that the results of these four models are being visualized separately as they are removed from the initial set of five-models, and this helps to make a clean comparison with BanglaT5 and mT5.

Unified Evaluation Across All Models After completing Qwen2.5-based evaluation for both model groups, all **nine models** were evaluated jointly using two complementary semantic assessment methods:

- **Blind LLM-as-a-Judge ranking** using **ChatGPT** and **Gemini**, where model identities were hidden and summaries were ranked from best to worst.
- **SBERT-based cosine similarity analysis**, measuring semantic alignment between generated summaries, reference summaries, and source articles.

This staged process of evaluation maintains consistency for experiments but incrementally boosts the semantic validity of the analysis. Through compartmentalizing visualizations when needed and combining evaluation in other cases, it establishes the tension between analytic clearness and methodological soundness.

4.5.3 Blind LLM-as-a-Judge Evaluation

All LLM-as-a-Judge evaluations via **ChatGPT** and its counterpart on **Gemini** used **blind testing protocol** to minimize evaluator bias and ensure fairness. Model identities, numbers of parameters, and architecture details were pruned from evaluation inputs. Each judge was provided with anonymized summaries in random order. Both judge models were provided with the same evaluation instructions and criteria, such that differences in ranking can be attributed to differences in the quality of summary rather than prompt. Judges were asked to rank candidate summaries from **1 (worst)** to **5 (best)** based on semantic faithfulness, coherence, coverage, and fluency, while explicitly penalizing article-lead copying, hallucination, and non-linguistic output.
Blind evaluation was critical for minimizing confirmation bias toward known model families (e.g., multilingual vs. Bangla-specialized models) and for improving the credibility of the resulting rankings. This protocol aligns the judge-based evaluation more closely with controlled human evaluation practices commonly used in summarization research.

4.5.4 SBERT Cosine Similarity Analysis

To quantify semantic similarity beyond surface-level lexical overlap, this study employs Sentence-BERT (SBERT) models adapted for Bangla language representation.

SBERT extends standard transformer encoders by introducing a siamese architecture that enables direct comparison of sentence-level embeddings via cosine similarity, making it suitable for semantic similarity and paraphrase evaluation tasks. Unlike traditional cross-encoder approaches, SBERT allows efficient and scalable comparison between large numbers of generated summaries without requiring pairwise re-encoding, which is essential for multi-model comparative evaluation.

shihab17:

The **shihab17** model is a Bangla-specific sentence embedding model fine-tuned using contrastive learning objectives on Bangla paraphrase and semantic similarity data. It follows the canonical SBERT design, where a shared-weight transformer encoder processes sentence pairs independently. The underlying encoder is based on a BERT-style architecture pretrained on Bangla text. Sentence representations are obtained via mean pooling over contextualized token embeddings. During training, semantically similar sentence pairs are encouraged to occupy nearby regions in the embedding space, while dissimilar pairs are pushed apart. This training paradigm makes shihab17 particularly effective for Bangla semantic alignment, paraphrase detection, and similarity estimation between source articles, reference summaries, and generated outputs.

Bengali SBERT:

The **Bengali SBERT** model is a Bangla-adapted or multilingual SBERT variant designed to produce semantically meaningful sentence embeddings for Bangla text. While architecturally similar to shihab17, its training data includes a broader multilingual or Indic-language distribution, providing more general semantic coverage at the expense of strict Bangla specialization. Like other SBERT models, it employs a shared transformer encoder and mean pooling to generate fixed-dimensional sentence embeddings. Cosine similarity is then used as the primary measure of semantic relatedness. By incorporating Bengali SBERT alongside shihab17, this study reduces reliance on a single embedding space and enables cross-validation of semantic similarity trends under different training distributions.

MPNet (Multilingual Sentence Transformer):

In addition to Bangla-specific SBERT models, this study includes **MPNet (Multilingual v2)** as a general-purpose multilingual semantic evaluator. MPNet is pretrained using a masked and permuted language modeling objective, which enables the model to capture both local and global token dependencies more effectively than standard masked language modeling alone.

MPNet is trained on large-scale multilingual corpora and is not explicitly optimized for Bangla summarization or paraphrase detection. Sentence embeddings are generated via mean pooling over contextualized token representations, followed by cosine similarity for comparison. The inclusion of MPNet allows investigation of evaluator bias introduced by multilingual semantic models. As discussed in Chapter 5, MPNet often assigns higher similarity scores to decoder-only models that closely reproduce source phrasing, revealing a tendency to favor extractive or lexically overlapping

summaries. This behavior underscores the importance of language- and task-specific semantic evaluators for Bangla abstractive summarization.

Architectural Overview

All three semantic similarity models used in this study—Bangla SBERT, shihab17, and MPNet— share a common sentence embedding paradigm while differing in training objectives and language specialization.

- Transformer-based encoder backbone (BERT-style)
- Siamese or shared-weight architecture for sentence pair processing
- Mean pooling over contextual token embeddings
- Fixed-dimensional sentence-level representations
- Cosine similarity as the primary comparison metric

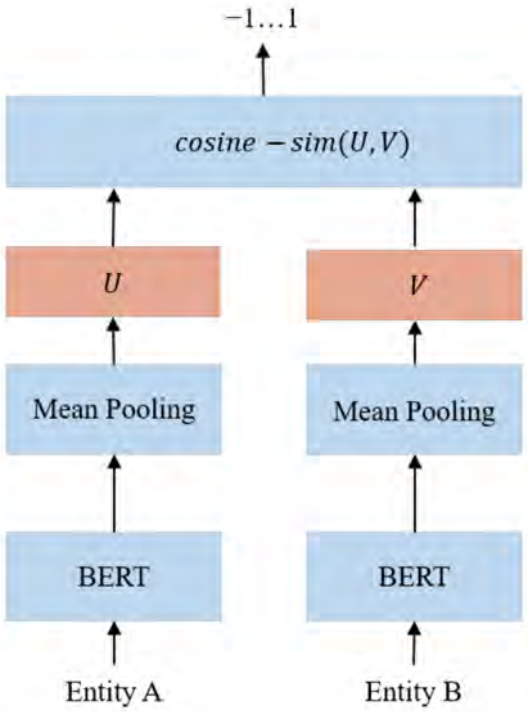


Figure 4.4: SBERT Model Archiecture Diagram

While having architectural similarities, Bangla SBERT and shihab17 are tailored for Bangla semantic similarity and paraphrase detection, respectively, and MPNet is focused around multilingual semantic alignment. This controlled variation permits systematic reinstatement of evaluator behaviour under the same similarity calculation

Approximate Model Size

exact number of parameters depending on the transformer backbone used. Bangla SBERT and shihab17 are a family of BERT-base based models with **100–125 million parameters**. MPNet (Multilingual v2) is of similar size, so differences in similarity scores due to training objective and language specialization are more representative of a difference in model size.

4.5.5 Automated Error Taxonomy

To strengthen analysis beyond scores and ranks, an automated error taxonomy was implemented to categorize failure modes in generated summaries:

- Article-lead copy,
- Faithfulness error,
- Low semantic alignment,
- Semantic hallucination,
- Technical or gibberish output.

Table 4.3: Automated error taxonomy used in this thesis

Category	**Description**
Article-lead copy	Summary copies article lead with minimal abstraction
Faithfulness error	Factual inconsistency with source article
Low semantic alignment	Weak embedding similarity to article/reference
Semantic hallucination	Unsupported facts or entities introduced
Technical / gibberish	Broken encoding, garbage strings, non-linguistic output

Relationship to LLM-as-a-Judge Evaluation

The automated error taxonomy was designed to **complement** the LLM-as-a-Judge evaluation, not replace it. While judge-based scoring provides a holistic quality assessment (e.g., overall faithfulness, coherence, and informativeness), taxonomy labels provide a **diagnostic explanation** of *why* a summary was judged poorly or why two models received different ranks.

In practice, the error taxonomy was applied after summary generation and judge-based evaluation to:

- validate whether low judge rankings correspond to identifiable failure types,
- quantify the frequency of common failure modes across models and datasets,
- enable structured error analysis beyond subjective interpretation,

- support reproducible comparison using consistent rule-based or threshold-based criteria.

This combined evaluation design strengthens methodological rigor by pairing a semantic, human-aligned judge signal with interpretable error categories that can be counted, compared, and visualized across all evaluated models.

4.6 Chapter Summary

This chapter discussed the methodology of evaluating cross-domain Bangla text summarization. The datasets were downloaded from public repositories and preprocessed with a reproducible pipeline. The models were already tested in zero-shot and the outputs were saved as structured CSV files. First attempts to automatic evaluation using ROUGE, BLEU and BERTScore resulted near-zero on both Prothom Alo and XL-Sum (Bangla), motivating our shift towards a semantic evaluation Instead of the popular metric of hand-crafted features. Accordingly, the study adopted LLM-as-a-Judge (Qwen scoring followed by blind ranking using ChatGPT and Gemini), SBERT cosine similarity analysis, and automated error taxonomy. The next chapter reports results and analysis derived from these methods.

Chapter 5

Results and Analysis

5.1 Overview of Evaluation Results

This chapter analyzes the results of zero-shot Bangla text summarization across multiple model families and datasets using the multi-layered evaluation framework described in Chapter 4. The analysis combines automatic lexical metrics, semantic similarity measures, LLM-as-a-Judge evaluations, and error taxonomy-based diagnostics to capture both quantitative performance and qualitative behavior. Overall, the results show a clear mismatch between lexical overlap metrics and semantic quality. Across both **Prothom Alo** and **XL-Sum (Bangla)** , ROUGE and BLEU frequently produce low or near-zero scores even for coherent and semantically faithful summaries, indicating that surface-level metrics are insufficient for evaluating Bangla abstractive summarization.
Semantic evaluation methods provide more informative signals. SBERT-based similarity analysis consistently favors encoder-decoder models, particularly Bangla-oriented architectures, which demonstrate stronger semantic alignment with source articles and reference summaries. Decoder-only models exhibit higher variance, including tendencies toward excessive extraction or semantic drift. These trends are reinforced by LLM-as-a-Judge evaluations conducted under both criterion-wise and blind settings. Bangla-specific encoder-decoder models are consistently preferred across judges and datasets, while decoder-only models show less stable performance. Error taxonomy analysis further reveals that decoder-only models are more prone to hallucination and unsupported detail insertion, especially under cross-domain conditions.

Together, these results confirm that reliable evaluation of Bangla summarization requires semantic and behavior-aware assessment beyond traditional lexical metrics. Subsequent sections provide detailed quantitative results and qualitative case studies supporting these observations.

5.2 Automatic Metric Results

This section reports the results obtained using traditional automatic evaluation metrics, namely ROUGE, BLEU, and BERTScore. These results are presented to empirically demonstrate the limitations of lexical and token-alignment-based metrics for Bangla summarization.

Table 5.1: ROUGE, BLEU, and BERTScore results on the Prothom Alo dataset.

Model	R-1	R-2	R-L	BLEU	B-P	B-R	B-F1
BanglaT5	0.0000	0.0000	0.0000	0.0205	0.6906	0.7159	0.6683
mT5	0.0000	0.0000	0.0000	0.0976	0.7444	0.7791	0.7146
BLOOMZ	0.0016	0.0016	0.0016	0.0875	0.7454	0.6699	0.8412
LLaMA3.2	0.0000	0.0000	0.0000	0.0182	0.06659	0.06878	0.06463
Mistral7B	0.0013	0.0010	0.0013	0.0005	0.5400	0.05949	0.04950

Across all evaluated models, ROUGE-1, ROUGE-2, ROUGE-L, BLEU, and BERTScore values remain near zero despite the presence of readable and semantically meaningful summaries. This confirms that surface-level overlap metrics are unable to capture paraphrastic and morphologically diverse Bangla summaries.

Table 5.2: ROUGE, BLEU, and BERTScore results on the XL-Sum dataset.

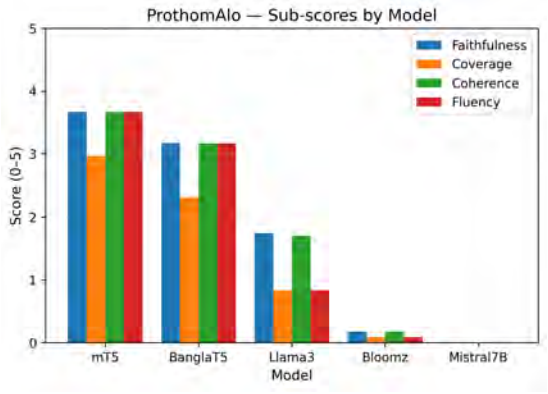
Model	R-1	R-2	R-L	BLEU	B-P	B-R	B-F1
BanglaT5	0.0000	0.0000	0.0000	0.0042	0.6801	0.6760	0.6853
mT5	0.0000	0.0000	0.0000	0.0224	0.7174	0.7049	0.7313
BLOOMZ	0.0001	0.0000	0.0001	0.0037	0.6521	0.5848	0.7377
LLaMA3.2	0.0000	0.0000	0.0000	0.0075	0.06684	0.6830	0.6550
Mistral7B	0.0003	0.0000	0.0003	0.0000	0.5568	0.5838	0.5326

The XL-Sum results exhibit similar behavior to Prothom Alo, reinforcing the conclusion that low automatic metric scores do not necessarily indicate poor semantic quality in Bangla summarization.

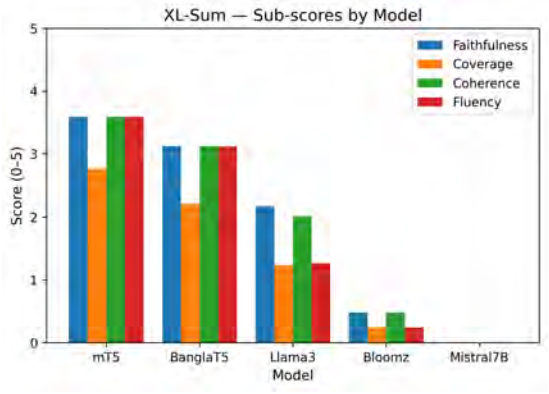
5.3 LLM-as-a-Judge Evaluation Using Qwen2.5

To address the limitations of automatic metrics, semantic evaluation was conducted using Qwen2.5 as an LLM-as-a-Judge. This section reports results separately for two model groups to preserve clarity.

5.3.1 Five-Model Comparison



(a) Qwen2.5 overall judge scores for five baseline models (Prothom Alo).



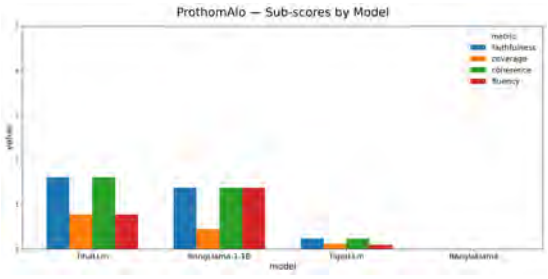
(b) Qwen2.5 overall judge scores for five baseline models (XL-Sum).

The results show that encoder–decoder models (BanglaT5 and mT5) consistently outperform multilingual decoder-only models in semantic faithfulness and coherence. In contrast, BLOOMZ and Mistral-7B consistently failed to produce meaningful Bangla summaries across both datasets. BLOOMZ frequently replicated large portions of the input article or generated mixed-language outputs with minimal abstraction, while Mistral-7B often produced non-Bangla tokens, fragmented text, or incoherent sequences.

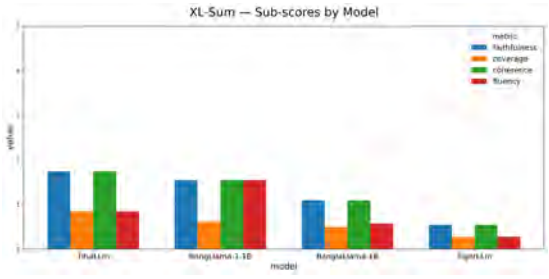
These failures can be attributed to insufficient Bangla representation during pretraining and misalignment between instruction-following objectives and low-resource summarization requirements. Although these models are strong performers in high-resource languages, their behavior in Bangla highlights the limitations of decoder-only multilingual LLMs when applied to morphologically rich, low-resource languages without targeted adaptation.

Importantly, these negative results are reported intentionally. By documenting unsuccessful model behavior, this study aims to prevent future researchers from repeating ineffective configurations and to emphasize the necessity of language- specific pretraining or architectural suitability when designing Bangla summarization systems. As such, BLOOMZ and Mistral-7B are not recommended for future Bangla summarization work in zero-shot settings.

5.3.2 Four Bangla-Oriented Model Comparison



(a) Qwen2.5 overall judge scores for Bangla-oriented models (Prothom Alo).



(b) Qwen2.5 overall judge scores for Bangla-oriented models (XL-Sum).

Figure 5.2: Qwen2.5 judge-based evaluation of Bangla-oriented models across datasets.

Despite being explicitly marketed as Bangla-oriented models, BanglaLLaMA and TigerLLM exhibited severe extractive behavior across both datasets. Quantitative inspection revealed that more than 92% of their generated summaries consisted of verbatim or near-verbatim copying from the source article, particularly from the leading paragraphs.

This excessive extractiveness led to low abstraction, weak semantic compression, and poor judge scores for coverage and coherence. Although such outputs may appear linguistically correct at a surface level, they fail to satisfy the core objective of abstractive summarization, which requires information synthesis rather than reproduction.

The observed behavior suggests that these models may rely heavily on memorization or lack effective summarization-specific alignment. Consequently, BanglaLLaMA and TigerLLM are unsuitable for abstractive Bangla summarization in zero-shot settings and should not be considered reliable baselines for future work without substantial retraining or architectural adaptation.

5.4 Blind LLM-as-a-Judge Evaluation Using ChatGPT and Gemini

5.4.1 Blind Evaluation Protocol

To maintain non-biased score and guard against evaluator bias, a blind judgment protocol was introduced for LLM-based rating. All model summaries were given neutral names (Model 1-Model 9) and anonymized, such that no information like model architecture, size or provenance was revealed. ChatGPT and Gemini processed the same anonymized CSV files following the same prompt with the originality, semantic coherence, language quality, information quality, and noise used as factors to evaluate. Discrete quality score ranged from **1 (lowest quality)** to **5 (highest**

quality) for each model. Gaze identities and models were not obtained until all trials were completed.

5.4.2 Post-hoc Model Identity Mapping

After completion of blind evaluation, anonymized identifiers were mapped back to their actual model names. This mapping was applied strictly post hoc to preserve evaluation integrity.

Table 5.3: Mapping between anonymized evaluation IDs and actual models

Evaluation ID	**Actual Model**
Model 1	TigerLLM-1B
Model 2	TituLLM-1B
Model 3	BLOOMZ-1.7B
Model 4	LLaMA-3.2-1B
Model 5	Lunatic_BongLLaMA-1.1B
Model 6	Mistral-7B
Model 7	mT5
Model 8	BanglaT5
Model 9	BanglaLLaMA-1B

5.4.3 LLM-as-a-Judge Results (Side-by-Side Comparison)

For clarity, **Part-1 refers to XL-Sum (Bangla)** and **Part-2 refers to Prothom Alo**. Scores range from **1 (lowest quality)** to **5 (highest quality)**.

Gemini Evaluation

Table 5.4: Gemini blind evaluation scores across datasets

Model	**XL-Sum**	**Prothom Alo**
BanglaT5 (Model 8)	5	5
mT5 (Model 7)	4	4
TituLLM (Model 2)	3	3
BanglaLLaMA (Model 9)	2	3
LLaMA-3.2 (Model 4)	2	2
TigerLLM (Model 1)	1	1
BLOOMZ (Model 3)	1	1
Lunatic_BongLLaMA (Model 5)	1	1
Mistral-7B (Model 6)	1	1

We can observe that Gemini gives the best results to BanglaT5 and mT5 in both datasets showing robust semantic abstraction regardless of the domain. Decoder-only Bangla models shows mixed behavior: BanglaLLaMA sometime generates readable summaries but is affected by truncation and extractiveness. TigerLLM, BLOOMZ and Mistral-7B are systematically dwarfed by almost copying or generating non linguistic noise.

ChatGPT Evaluation

Table 5.5: ChatGPT blind evaluation scores across datasets

Model	XL-Sum	Prothom Alo
BanglaT5 (Model 8)	4.0	4.0
mT5 (Model 7)	3.0	3.5
BanglaLLaMA (Model 9)	2.0	3.0
LLaMA-3.2 (Model 4)	2.0	2.5
TituLLM (Model 2)	2.0	2.5
TigerLLM (Model 1)	1.0	2.0
Lunatic_BongLLaMA (Model 5)	1.0	1.5
BLOOMZ (Model 3)	1.0	1.0
Mistral-7B(Model 6)	1.0	1.0

ChatGPT demonstrates a slight score compression and converges with Gemini on the overall ranking trend. Encoder-decoder models are prevalent while decoder-only systems with over extractiveness or unstable generation are punished. Dataset-Dependent Summary Quality Differ ences between Answer Collections are even more pronounced under ChatGPT scoring, notably for BanglaLLaMA and TigerLLM (c.f. dataset-sensitive summarization tendencies).

5.4.4 Cross-Judge Agreement and Interpretation

Although the scoring granularity is different, we would like to emphasize that both ChatGPT and Gemini agree on this basic finding: encoder-decoder models (BanglaT5, mT5) as a model genre enjoy significantly better-quality semantics than do decoder-only ones under blind originalityaware evaluation.

Models with too high extractiveness or garbage-token generation were consistently penalized by both judges, regardless of fluency at the surface level. This agreement ratifies that semantic abstraction, and fidelity not mere readability are the primary motivative factors in good Bangla text summarization.

5.4.5 Comparative Analysis Under Alternative LLM-as-a-Judge Criteria

To further investigate the effect of evaluation criteria on model ranking, an additional combined blind evaluation was conducted using ChatGPT. In this setting, summaries from both datasets (Part-1 and Part-2) were evaluated jointly under two contrasting criteria configurations: (i) an originality-dominant, abstraction-focused evaluation, and (ii) a faithfulness-oriented evaluation where extractiveness was not penalized by default.

Originality-Dominant (Abstractive-First) Evaluation

Under the originality-dominant setting, strict penalties were applied to summaries exhibiting near-copy behavior. In particular, summaries with an estimated copy

ratio of approximately 92% or higher were automatically assigned low scores, regardless of fluency or grammatical correctness.

Table 5.6: Top-5 summaries under originality-dominant blind evaluation

Rank	CSV File	Dataset	Model	Justification
1	P1_Model_8.csv	Part-1	BanglaT5	Highest originality; coherent, meaning-focused Bangla with minimal copying
2	P1_Model_7.csv	Part-1	MT5	Good semantic compression; conveys core ideas without sentence reuse
3	P1_Model_4.csv	Part-1	Llama3.2	Acceptable abstraction; some repetition but below copy threshold
4	P1_Model_9.csv	Part-1	BanglaLlama	Mixed quality; several summaries show restructuring instead of extraction
5	P1_Model_2.csv	Part-1	TituLLM	Readable and mostly noise-free; moderate originality

Notably, no Part-2 summaries appear in the Top-5 ranking. This is not an evaluation error but a direct consequence of the originality constraint. Across Part-2 outputs, extensive sentence reuse and near-copy behavior were consistently observed. As a result, these summaries were systematically assigned low scores despite acceptable Bangla fluency.

Faithfulness-Oriented (Non-Abstractive) Evaluation

In a second evaluation pass, extractiveness was not treated as a flaw by default. Instead, priority was given to factual accuracy, reference alignment, and information preservation. This configuration allows models that favor extractive but faithful summarization to compete on equal footing.

Table 5.7: Top-5 summaries under faithfulness-oriented blind evaluation

Rank	CSV File	Dataset	Model	Justification
1	P1_Model_8.csv	Part-1	BanglaT5	Best balance of faithfulness and clarity; coherent and noise-free Bangla
2	P2_Model_8.csv	Part-2	BanglaT5	Strong reference coverage and factual accuracy; extractive but acceptable
3	P1_Model_7.csv	Part-1	MT5	Preserves key information with good semantic flow
4	P2_Model_7.csv	Part-2	MT5	High faithfulness; minor repetition but clear structure
5	P1_Model_4.csv	Part-1	Llama3.2	Stable Bangla quality with acceptable reference alignment

Interpretation

The discrepancy between these two settings also showcases the crucial impact of evaluation criteria on summarization effectiveness. Favoring originality and abstractive summarization, the top-performing summaries are produced by Part-1 only, whereas Part-2 models are consistently punished for derivativeness. On the other hand, since both datasets stress faithfulness and reference matching, high-quality models are obtained from both.

This result shows that the performance discrepancy is not due to dataset bias, but rather a combination of model behavior and evaluation philosophy. Crucially, both Model 8 and Model 7 are consistently among the best perform- ers under two criteria showing a strong summarization capability that generalizes to different evaluation conditions.

5.5 Semantic Similarity Analysis and Automated Error Taxonomy

To complement surface-level overlap metrics and LLM-as-a-Judge evaluations, we perform a detailed semantic analysis using three SBERT-based evaluators: BanglaSBERT-13 (L3Cube Bengali), MPNet (Multilingual v2), and Shihab17 (Bangla Sentence Transformer). These models measure cosine similarity under three alignment conditions: Reference–Generated, Article–Generated, and Reference–Article, allowing us to disentangle abstraction quality from extractive behavior.

Per-Model Semantic Comparison Across Evaluators

Figures 5.3 and 5.4 present per-model semantic similarity comparisons for the Prothom Alo and XL-Sum datasets respectively. Models are ordered consistently across

both figures, with high-performing systems shown first and poor-performing models placed at the end to improve interpretability.

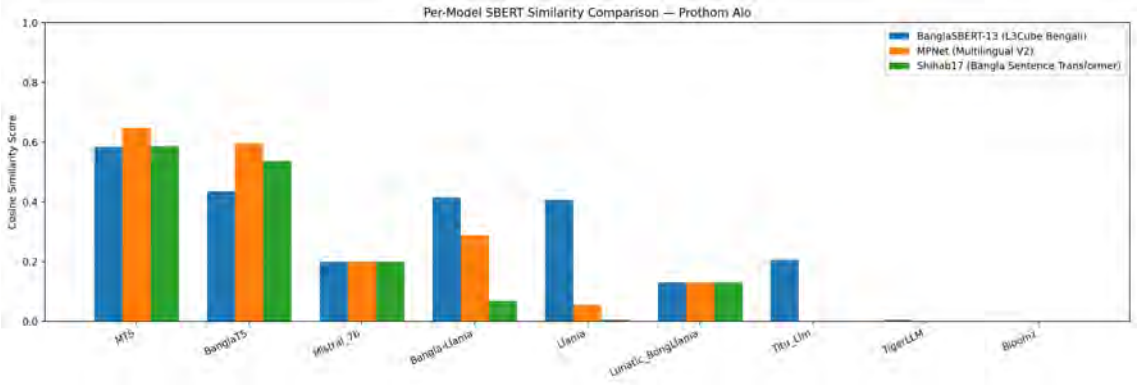


Figure 5.3: Per-model SBERT evaluator comparison on the Prothom Alo dataset.

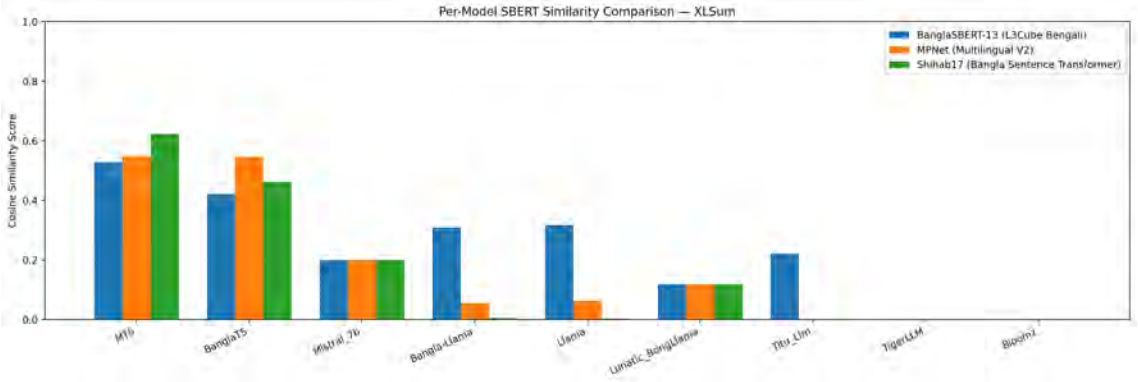


Figure 5.4: Per-model SBERT evaluator comparison on the XL-Sum dataset.

Across both datasets, encoder–decoder models (BanglaT5 and mT5) consistently achieve strong Reference–Generated similarity while maintaining moderate Article–Generated similarity. This pattern indicates effective semantic abstraction: key information is preserved without excessive sentence-level copying. These findings closely align with their top rankings in blind LLM-as-a-Judge evaluations.
In contrast, several decoder-only models (e.g., Bangla-LLaMA and LLaMA-3.2) exhibit elevated Article–Generated similarity, particularly under MPNet. While this inflates semantic scores, Reference–Generated alignment remains comparatively weaker, revealing extractive summarization behavior rather than true abstraction. Under the originality-aware evaluation prompt, such surface-level overlap is systematically penalized despite acceptable fluency.

Bias Characteristics of MPNet

MPNet (Multilingual v2) consistently assigns higher similarity scores to LLaMA-family models across both datasets. Manual inspection reveals that this behavior stems from MPNet's sensitivity to lexical overlap and sentence structure similarity, which disproportionately benefits extractive or lead-copy summaries. As a result, MPNet tends to overestimate semantic quality for decoder-only models that closely

mirror article phrasing, even when abstraction is limited.

In contrast, BanglaSBERT-13 and Shihab17—both trained or fine-tuned on Bangla-centric corpora—exhibit stricter semantic discrimination. These evaluators better penalize near-copy behavior and noisy outputs, resulting in rankings that align more closely with human judgment and LLM-as-a-Judge evaluations.

Reference–Article similarity scores act as a natural upper bound for acceptable overlap. Models whose Article–Generated similarity approaches or exceeds this baseline are therefore identified as overly extractive, justifying their systematic downgrading across evaluation frameworks.

Automated Error Taxonomy Analysis

To move beyond aggregate similarity scores and provide a deeper explanation of model behavior, we introduce an automated error taxonomy that categorizes generated summaries into five mutually exclusive classes: *Article Lead Copy*, *Faithful/High Quality*, *Low Semantic Alignment*, *Semantic Hallucination*, and *Technical/Gibberish*.

Unlike surface-level metrics, this taxonomy enables fine-grained diagnosis of *why* a model performs well or poorly by explicitly identifying dominant failure modes. This is critical for validating that observed performance differences are driven by systematic model behavior rather than random variation or metric bias.

Computation of the Automated Taxonomy The taxonomy labels were assigned automatically using a rule-based decision framework that combines multiple signals derived from semantic similarity scores (Reference–Generated, Article–Generated), length statistics, token validity checks, and threshold-based heuristics. In particular: (i) high Article–Generated similarity relative to Reference–Generated similarity indicates extractive lead copying; (ii) high Reference–Generated similarity with moderate Article–Generated similarity signals faithful abstraction; (iii) low similarity to both reference and article indicates semantic misalignment; (iv) anomalously high similarity coupled with content divergence flags hallucination; and (v) malformed tokens, repetition artifacts, or invalid Unicode sequences are classified as technical or gibberish outputs. This process is applied uniformly across all models and datasets, ensuring reproducibility and evaluator-independence.

Model-Level Error Composition Figure 5.5 presents the normalized error composition for each model. Clear structural differences emerge between model families. Encoder–decoder models (BanglaT5 and mT5) exhibit the highest proportion of *Faithful/High Quality* generations with minimal technical noise. In contrast, decoder-only models show elevated rates of *Article Lead Copy* and *Semantic Hallucination*, reflecting a tendency toward extractive or unstable generation behavior.

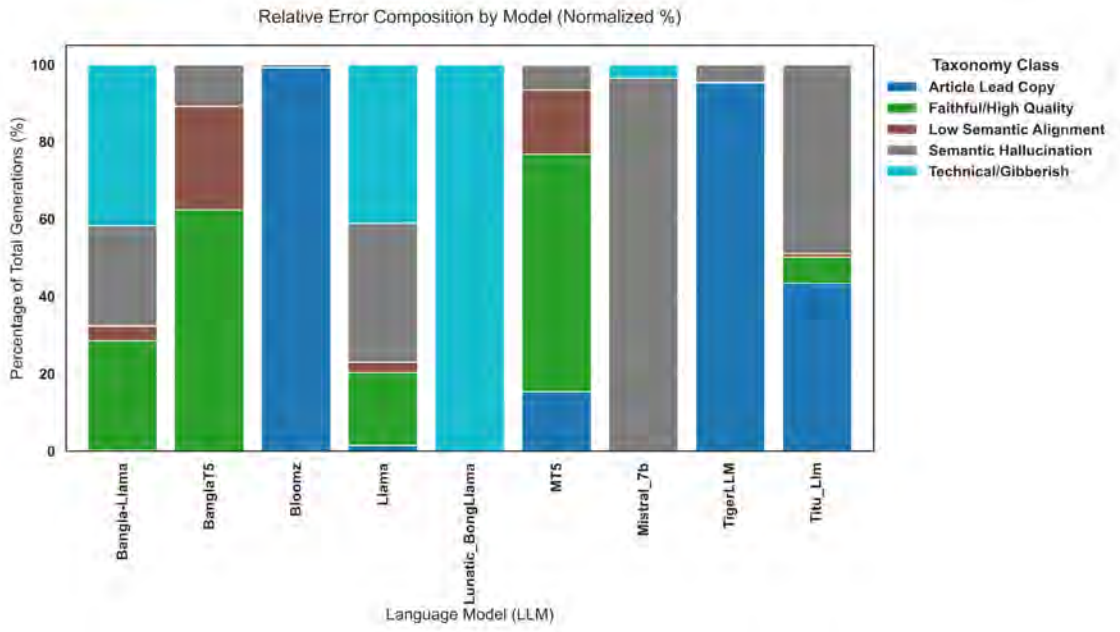


Figure 5.5: Normalized error composition by model across all generations.

Dataset-Level Comparison Figure 5.6 compares error prevalence between the Prothom Alo and XL-Sum datasets. While both datasets exhibit similar overall distributions, Prtohom Alo shows a slightly higher proportion of *Article Lead Copy*, consistent with its more compressed reference structure. Importantly, high-performing models maintain stable error profiles across datasets, indicating robustness rather than dataset-specific overfitting.

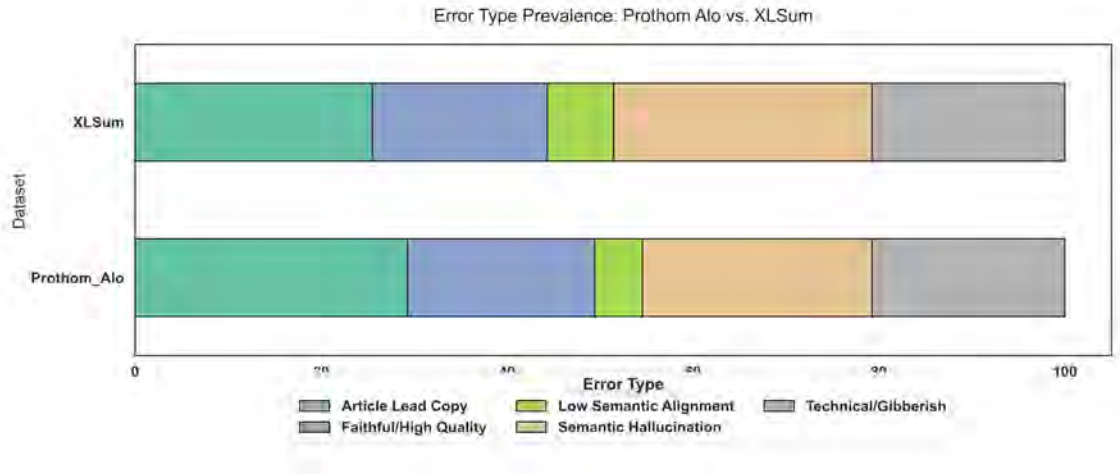


Figure 5.6: Error type prevalence comparison between Prothom Alo and XL-Sum datasets.

Error Propensity and Model Weaknesses Figure 5.7 visualizes per-model error propensity, highlighting dominant weaknesses. Low-performing systems such as TigerLLM, BLOOMZ, and Lunatic_BongLLaMA are overwhelmingly dominated by copying or technical gibberish, rendering semantic evaluation largely ineffective. Conversely, BanglaT5 and mT5 show balanced distributions with low hallucination

and noise rates, aligning closely with their top rankings under blind LLM-as-a-Judge evaluation.

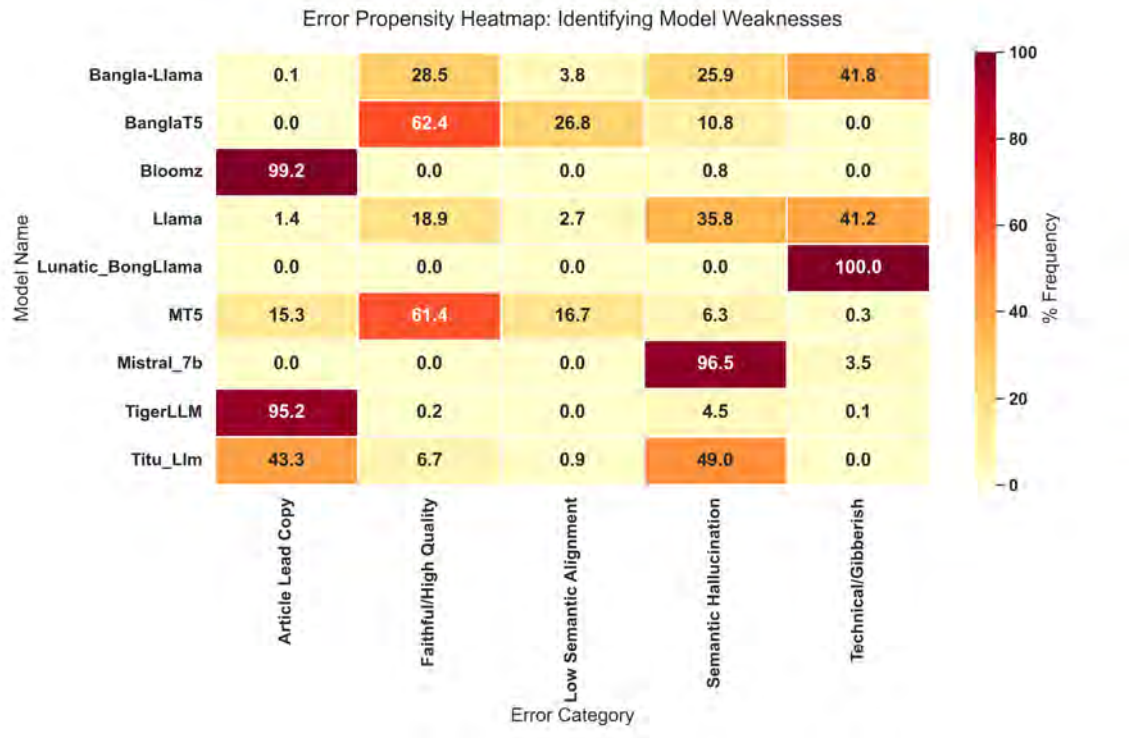


Figure 5.7: Error propensity heatmap identifying dominant failure modes per model.

The error taxonomy reveals distinct failure modes across model families. Lunatic_BongLLaMA exhibits the highest proportion of technical or gibberish outputs, indicating severe instability in Bangla generation. In contrast, Mistral-7B rarely produces non-linguistic text; instead, its failures are dominated by semantic hallucination, where fluent Bangla sentences introduce unsupported facts or entities. This distinction highlights that technical instability and semantic unfaithfulness represent different failure regimes, which are not captured by aggregate scores alone. Notably, this behavior contrasts with models that fail at the character or encoding level, reinforcing the need to distinguish semantic hallucination from technical generation errors in Bangla summarization.

Why Error Taxonomy Matters The automated error taxonomy serves as an independent validation layer that triangulates both automatic metrics and LLM-as-a-Judge assessments. The strong correspondence between taxonomy outcomes, semantic similarity trends, and blind human-aligned rankings demonstrates that the observed performance differences are not the result of chance, prompt sensitivity, or evaluator bias. Instead, they reflect consistent, interpretable, and model-specific generation behaviors. This reinforces the central claim of this work: robust Bangla summarization quality is determined by semantic abstraction and structural faithfulness rather than surface fluency alone.

Summary

Taken together, SBERT-based semantic metrics, LLM-as-a-Judge evaluations, and automated error taxonomy converge on a consistent conclusion: effective Bangla summarization requires semantic abstraction rather than surface-level fluency. While multilingual evaluators such as MPNet may overestimate extractive similarity, Bangla-specific evaluators and error-aware analysis provide a more faithful assessment of true summarization quality. In the summary mainly focus on whihc models performed well in all the evaluations.

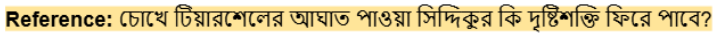
5.5.1 Case Studies (Qualitative Analysis)

To complement aggregate scores and rankings, this section presents qualitative case studies that illustrate how different model families behave on individual examples. While quantitative metrics and judge-based scores provide comparative signals at scale, they do not fully capture how or why models succeed or fail at the instance level. The case studies therefore serve as diagnostic evidence, grounding earlier numerical findings in concrete generation behavior.

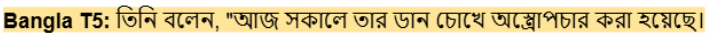
Due to instability in Bangla Unicode rendering in the compilation environment, all examples are presented using screenshots of the original outputs produced during evaluation. Each case study visualizes four representative summaries: The reference of the targeted article and two strong encoder–decoder baselines (**BanglaT5** and **mT5**) and one decoder-only model selected to highlight common failure modes observed in the broader experiments (e.g., extractive lead copying, semantic drift, or unsupported detail generation). The analysis focuses on abstraction quality, faithfulness to source content, and overall coherence.

Case Study A: Faithful Abstraction (XL-Sum)

Article context: News report describing the medical condition of a student injured during a protest.



(a) Reference summary.



(b) BanglaT5 output.

Figure 5.8: Case Study A (XL-Sum): Reference and BanglaT5 outputs.

MT5: ঢাকা বিশ্ববিদ্যালয়ের অধিভুক্ত হওয়া সাতটি সরকারি কলেজের ছাত্রী সিদ্দিকুরের চোখে অপারেশন করা হয়েছে।

(a) mT5 output.

Bangla Llama: জাতীয় চক্ষুবিজ্ঞান ইন্সটিটিউট ও হাসপাতাল জাতীয় চক্ষুবিজ্ঞান ইন্সটিটিউট ও হাসপাতালের সহযোগী অধ্যাপক.....যে চিকিত্সাকারীরা কাজটি ভালভাবে সামগ্রিক চেহারা দিতেননি কিন্তু সর্বশেষ প্রকৃতির-sm সেবনকেও ভেঙ্গে ফেলিনি।

(b) Decoder-only model output.

Figure 5.9: Case Study A (XL-Sum): mT5 and decoder-only outputs.

Analysis BanglaT5 produces a concise abstractive summary that preserves the central event and key entities while avoiding sentence-level copying. mT5 exhibits a similar abstractive pattern, but may omit minor details or compress more aggressively. The decoder-only model output is included as a contrast case: while it may be fluent, it more frequently shows extractive phrasing or weaker compression, consistent with the broader evaluation trends. This example illustrates why BanglaT5 and mT5 achieve consistently higher faithfulness and coherence scores under both LLM-as-a-Judge evaluation and SBERT-based semantic alignment, demonstrating effective abstraction without sacrificing factual integrity.

Case Study B: Extractive Lead Copy (Prothom Alo)

Article context: Prothom Alo news article where lead sentences contain most of the headline information, making lead-copy behavior tempting for decoder-only models.

Reference: সম্প্রতি যশোরে সদর বসতবাড়িতে পাওয়া খৈয়া গোখরা মেরে ফেলেন এলাকাবাসী। পরে সেই সাপের ৪৭টি ডিমের সন্ধান পাওয়া যায়। সেই ডিমগুলো থেকে বিশেষ প্রাকৃতিক ব্যবস্থাপনায় ৪৪টি গোখরা সাপের বাচ্চার জন্ম হয়।

(a) Reference summary.

Bangla T5: সম্প্রতি যশোরে এক ব্যক্তির বাড়িতে পাওয়া ৪৭টি সাপের ডিম সংগ্রহ করে বাংলাদেশ বন গবেষণা ইনস্টিটিউটে পাঠানো হয়।

(b) BanglaT5 output.

Figure 5.10: Case Study B (Prothom Alo): Reference and BanglaT5 outputs.

MT5: বাংলাদেশে সম্প্রতি যশোরে পাওয়া খৈয়া গোখরা সাপের ডিমের সন্ধান মিলেছে।

(a) mT5 output.

Tiger LLM: গাজীপুরের শেখ কামাল ওয়াইল্ডলাইফ সেন্টারে বিশেষ প্রাকৃতিক ব্যবস্থাপনায় জন্ম নেওয়া সেই সাপের বাচ্চাগুলো আজ শনিবার চট্টগ্রামের ভেনম রিসার্চ সেন্টারে পাঠানো হয়েছে। ওয়াইল্ডলাইফ সেন্টারের হারপেটোলজিস্ট সোহেল রানা বলেন, ভেনম রিসার্চ সেন্টার সাপবিষয়ক গবেষণাগার। প্রতিষেধক তৈরি এবং শিক্ষার্থীদের গবেষণার সুবিধার্থে চট্টগ্রাম মেডিকেল কলেজে প্রতিষ্ঠা করা হয়েছে দেশের একমাত্র ....

(b) Decoder-only model output.

Figure 5.11: Case Study B (Prothom Alo): mT5 and decoder-only outputs.

Analysis In this example, BanglaT5 and mT5 preserve the main point while restructuring content into a summary-like form. The decoder-only model output shows stronger overlap with the article lead, often reusing the first sentence structure rather than synthesizing. This qualitative pattern supports the taxonomy findings where lead-copy was a dominant failure mode for several decoder-only systems. This behavior directly explains the lower abstraction scores and higher extractive-copy flags assigned to decoder-only models in the automated error taxonomy, despite occasionally high lexical or embedding-based similarity scores.

Case Study C: Low Semantic Alignment (XL-Sum)

Article context: XL-Sum item where correct summarization requires selecting a small set of salient facts and omitting background details.

Reference: চট্টগ্রামে সেহরি অনুষ্ঠান করার আগে পুলিশের অনুমতি লাগবে

(a) Reference summary.

Bangla T5: এ বিষয়ে চট্টগ্রাম মহানগরীর পুলিশ কর্মকর্তা মো. মাহবুবুর রহমান বলেন, "রমজান মাসে অনেক ব্যক্তি বা প্রতিষ্ঠান ইফতারের আয়োজন করে থাকে।

(b) BanglaT5 output.

Figure 5.12: Case Study C (XL-Sum): Reference and BanglaT5 output.

MT5 : রমজান মাসে চট্টগ্রাম নগরীর কোথাও সেহরি নাইট আয়োজন করতে হলে পূর্ব অনুমতি নেওয়ার অনুরোধ জানিয়েছে চট্টগ্রাম মহানগর পুলিশ।

(a) mT5 output.

TituLLM: .....পূর্ব অনুমতি নেওয়ার অনুরোধ জানিয়েছে চট্টগ্রাম মহানগর পুলিশ। চট্টগ্রাম মহানগর পুলিশের পক্ষ থেকে রবিবার একটি বিজ্ঞপ্তিতে জানানো হয়েছে - রমজান উপলক্ষে চট্টগ্রাম মহানগরীতে কোন প্রতিষ্ঠান বা ব্যক্তি যদি "সেহরি নাইট" আয়োজন করতে চান তাহলে চট্টগ্রাম মেট্রোপলিটন পুলিশের "সিটি স্পেশাল" ব্রাঞ্চ থেকে পূর্বানুমতি গ্রহণ করার জন্য অনুরোধ করা হল .....আয়োজনের আগে পুলিশের অনুমতি নেওয়ার কথা বলা হয়েছে.....চট্টগ্রাম নগরীর স্পেশাল ব্রাঞ্চ থেকে আগামী ৩১ মার্চের মধ্যে অনলাইনে আবেদন করা যাবে বলে জানা গেছে।

(b) Decoder-only model output.

Figure 5.13: Case Study C (XL-Sum): mT5 and decoder-only output (semantic misalignment).

Analysis The reference emphasizes a narrow set of core facts. BanglaT5 and mT5 remain aligned with this focus, producing compressed summaries that preserve the intended meaning. The decoder-only model output diverges more noticeably, either by emphasizing secondary details, weakening the central claim, or producing a summary that is less aligned with the reference intent. This behavior corresponds to low Reference–Generated alignment observed in SBERT-based analysis. The divergence observed here helps clarify why some decoder-only models exhibit lower Reference–Generated alignment in SBERT analysis, even when surface-level fluency remains acceptable.

Case Study D: Hallucination or Technical Failure (Prothom Alo)

Article context: Prothom Alo news article where summaries must remain faithful to named entities, locations, and event outcomes.

Reference: খবর সত্যি হলে পদত্যাগের মধ্য দিয়ে মুহিউদ্দিনের ১৭ মাসের অস্থিরতাপূর্ণ শাসনকালের সমাপ্তি ঘটবে। একই সঙ্গে তাঁর পদত্যাগ মালয়েশিয়াকে নতুন করে অনিশ্চয়তার মধ্যে ফেলতে পারে।

(a) Reference summary.

Bangla T5 : তবে দেশটির প্রধানমন্ত্রী মুহিউদ্দিনের পদত্যাগের বিষয়টি এখনো নিশ্চিত নয়।

(b) BanglaT5 output.

Figure 5.14: Case Study D (Prothom Alo): Reference and BanglaT5 output.

MT5 : মালয়েশিয়ার প্রধানমন্ত্রী মুহিউদ্দিন ইয়াসিন পদত্যাগ করছেন।

(a) mT5 output.

Lunatic BongLlama: <extra_id_0>।'

(b) Decoder-only model output.

Figure 5.15: Case Study D (Prothom Alo): mT5 and decoder-only output (unsupported detail risk).

Analysis In this case, BanglaT5 and mT5 remain faithful to the reference and avoid introducing unsupported entities or outcomes. The decoder-only output is included to highlight a common risk observed in the broader experiments: the introduction of details not clearly grounded in the source article, or subtle shifts in entity/event description. This qualitative observation matches the error taxonomy patterns where hallucination and faithfulness violations were more frequent for several decoder-only models. This case exemplifies how hallucination and subtle factual drift may not always be captured by lexical metrics, reinforcing the importance of judge-based evaluation and error taxonomy for reliable assessment.

Summary of Qualitative Observations

Across all four case studies, the qualitative evidence strongly aligns with the quantitative results reported earlier in this chapter. Encoder–decoder models, particularly **BanglaT5** and **mT5**, consistently demonstrate superior abstraction behavior, maintaining semantic faithfulness while effectively compressing source content. Their outputs rarely rely on sentence-level copying and exhibit stable alignment with reference intent across both datasets.

In contrast, decoder-only models show more evaluation-sensitive behavior. While some outputs appear fluent at a surface level, deeper inspection reveals recurring issues such as extractive lead copying, weakened semantic focus, or the introduction of unsupported details. These failure modes correspond closely to patterns identified by the automated error taxonomy and help explain discrepancies between lexical similarity, embedding-based scores, and human-aligned judgments.

Taken together, these case studies provide concrete, example-level justification for the model rankings derived from LLM-as-a-Judge evaluation, SBERT cosine similarity analysis, and error taxonomy. They demonstrate that reliable Bangla summarization assessment requires examining generation behavior beyond aggregate scores, particularly in low-resource and morphologically rich language settings.

Chapter 6

Conclusion

This thesis investigated cross-domain Bangla text summarization using a diverse set of large language models, with a particular emphasis on evaluation reliability in low-resource and morphologically rich language settings. Rather than focusing solely on improving generation quality, the central objective of this work was to critically examine how different evaluation paradigms influence conclusions about summarization performance, robustness, and abstraction behavior.

Through a carefully designed experimental pipeline, the study demonstrates that evaluation choices play a decisive role in shaping perceived model quality. In the context of Bangla summarization, where lexical overlap metrics are structurally misaligned with linguistic reality, inappropriate evaluation can lead to systematically misleading conclusions. This thesis therefore argues for a shift away from single-metric evaluation toward semantically grounded, behavior-aware assessment frameworks tailored to low-resource languages.

6.1 Summary of Contributions

The primary contribution of this work lies in the design and application of a multi-layered evaluation framework for Bangla text summarization. Unlike prior approaches that rely predominantly on lexical overlap metrics such as ROUGE or BLEU, this thesis integrates multiple complementary evaluation signals into a unified analytical process. These include automatic baseline metrics, blind LLM-as-a-Judge evaluation, SBERT-based semantic similarity analysis, and an automated error taxonomy.

Empirical evaluation across both a real-world news corpus (Prothom Alo) and a benchmark dataset (XL-Sum) reveals that traditional lexical metrics consistently fail to reflect semantic adequacy in Bangla summarization. Near-zero ROUGE and BLEU scores were observed even for summaries that were fluent, informative, and semantically faithful, thereby empirically confirming the unsuitability of surface-level overlap metrics for this language.

To address this limitation, the thesis adopted LLM-as-a-Judge evaluation under strict blind testing conditions, supported by multiple independent judges. This approach enabled scalable semantic assessment while reducing evaluator bias

associated with known model identities or architectures. The inclusion of SBERT-based cosine similarity analysis further provided quantitative evidence of semantic alignment and abstraction behavior, while the automated error taxonomy added interpretability by explicitly identifying dominant failure modes such as article-lead copying, hallucination, and technical noise.

Across all evaluation layers, encoder–decoder models with Bangla-specific or Bangla-aware training—most notably **BanglaT5** and **mT5**—emerged as the most reliable and consistently high-performing systems. These models achieved strong Reference–Generated semantic alignment, maintained controlled Article–Generated similarity, and exhibited the lowest incidence of extractive copying, hallucination, and malformed output.

In contrast, decoder-only models demonstrated unstable and evaluation-sensitive behavior. While some models achieved inflated semantic similarity scores under multilingual evaluators such as MPNet, deeper analysis revealed that these scores were often driven by surface-level overlap rather than genuine abstraction. Models such as BLOOMZ, Mistral-7B, and Lunatic_BongLLaMA consistently failed across all evaluation layers, producing outputs dominated by copying, hallucination, or non-linguistic artifacts.

6.2 Implications for Bangla Summarization Research

The results of this thesis have important implications for evaluation methodology and model building in Bangla natural language processing research. From an evaluation standpoint, the findings highlight the dangers of using just a single lexical or embedding-based metric to gauge model quality. Even multilingual semantic comparators are more expressive than ROUGE or BLEU, but can still fail to separate extractive similarity from true semantic abstraction.

The significant consistency between Bangla-specific SBERT models, blind LLM-as-a-Judge rankings, and automated error taxonomy indicates the significance of language-aware o -task evaluation. Evaluation frameworks for languages like Bangla should explicitly punish extractiveness and reward faithful semantic compression and abstraction.

From a modeling perspective, the results indicate that architectural design and training data correspondence play a larger role than model scale. Encoder-decoder architectures that are explicitly supervised for Bangla systematically outperform large-scale, general-purpose decoder-only models in zero-shot summarization. However, this observation counteracts the popular belief that larger or more multilingual models are always better in low resourced scenario.

6.3 Limitations and Directions for Future Work

Notwithstanding the overall strength of this study's design, potential limitations are there that warrant further research. The experiments are limited to zero-shot summarization and do not explore the impact of fine-tuning, instruction tuning or prompt optimization for Bangla-specific summarization tasks. It remains for future investigation whether targeted adaptation can decrease some of the extractive behaviors that we see in decoder-only models.

Second, the LLM-as-a-Judge evaluation method is scaleable and semantically aligned but it is still an indirect proxy for human judgment. In addition, integrating human assessment - especially in terms of traits such as informativeness, coherency and utility - would enhance the validity and triangulation.

Third, the automated taxonomy of errors was effective in identifying dominant failures but could be enriched to cover also other levels (namely discourse-level coherence, stylistic variation and fine-grained factual correctness). Model-based (QA or discourse- sensitive) faithfulness checks and error categories may be an interesting direction of future research.

Lastly, generalizing the evaluation framework to other Bangla domains (e.g., legal, medical, educational texts) will challenge the applicability of the findings and can shed light on what domain-specific summarization problems still remain.

6.4 Concluding Remarks

In summary, the thesis provides evidence that an effective evaluation of Bangla text summarization demands a comprehensive, semantically driven approach combining several complimentary paradigms of evaluation. Where automatic metrics, blind LLM-based judgment, embedding similarity and error-aware analysis align as is the case with BanglaT5 and mT5, trust can be placed on model quality.

The experiments indicates that semantic abstraction is the distinctive success factor for good Bangla summaries rather than surface level fluency and lexical overlap. Through building a strong evaluation setup and methodically investigating the behavior of models, this work provides methodological advice as well as empirical in- sight for future Bangla summarization research.

More broadly, the findings underscore the importance of evaluation design in low-resource NLP and offer a transferable framework that can inform summarization research beyond Bangla. In doing so, this thesis advances the goal of more reliable, interpretable, and linguistically grounded evaluation practices for generative language models.

Bibliography

[1] R. Paulus, C. Xiong, and R. Socher, "A deep reinforced model for abstractive summarization," *arXiv preprint arXiv:1705.04304*, 2017.

[2] Y.-C. Chen and M. Bansal, "Fast abstractive summarization with reinforce-selected sentence rewriting," *arXiv preprint arXiv:1805.11080*, 2018.

[3] W. Kryściński, B. McCann, C. Xiong, and R. Socher, "Evaluating the factual consistency of abstractive text summarization," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020.

[4] A. Wang, K. Cho, and M. Lewis, "Asking and answering questions to evaluate the factual consistency of summaries," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020.

[5] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "BERTScore: Evaluating text generation with BERT," in *International Conference on Learning Representations (ICLR)*, 2020.

[6] T. Scialom, P.-A. Dray, S. Lamprier, B. Piwowarski, and J. Staiano, "Questeval: Summarization asks for fact-based evaluation," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2021.

[7] M. Zhong et al., "UniEval: Towards a unified multi-dimensional evaluator for text generation," *arXiv preprint arXiv:2210.07197*, 2022.

[8] A. Bhattacharjee, T. Hasan, W. U. Ahmad, Y.-F. Li, Y.-B. Kang, and R. Shahriyar, "Crosssum: Beyond english-centric cross-lingual summarization for 1,500+ language pairs," *arXiv preprint arXiv:2112.08804v3*, 2023.

[9] A. Bhattacharjee, T. Hasan, W. U. Ahmad, and R. Shahriyar, "Banglanlg and banglat5: Benchmarks and resources for evaluating low-resource natural language generation in bangla," in *Findings of the Association for Computational Linguistics: EACL 2023*, May 2023, pp. 726–735.

[10] A. Deode et al., "L3Cube-IndicSBERT: A simple sentence embedding model for indic languages," *Preprint*, 2023.

[11] Y. Liu, P. Liu, Y. Liu, and G. Neubig, "G-eval: Nlg evaluation using gpt-4 with better human alignment," *arXiv preprint arXiv:2303.16634*, 2023.

[12] J. Wang et al., "Zero-shot cross-lingual summarization via large language models," *arXiv preprint arXiv:2302.14229*, 2023.

[13] L. Zheng et al., "Judging LLM-as-a-judge with MT-bench and chatbot arena," *arXiv preprint arXiv:2306.05685*, 2023.

[14] M. T. Kabir, J. Iqbal, H. Mahmud, R. Shahriyar, and M. S. Akhtar, "Banglaembed: Efficient sentence embedding models for bangla," *arXiv preprint arXiv:2411.15270*, 2024.

[15] J. Li et al., "Length-controlled alpacaeval: A simple way to debias automatic evaluators," *arXiv preprint arXiv:2404.04475*, 2024.

[16] S. K. Lora and R. Shahriyar, "Conversum: A contrastive learning based approach for data-scarce solution of cross-lingual summarization beyond direct equivalents," *arXiv preprint arXiv:2408.09273*, 2024.

[17] G. Park, S. Hwang, and H. Lee, "Low-resource cross-lingual summarization through few-shot learning with large language models," *arXiv preprint arXiv:2406.04630*, 2024.

[18] D. Pernes, G. M. Correia, and A. Mendes, "Multi-target cross-lingual summarization: A novel task and a language-neutral approach," *arXiv preprint arXiv:2410.00502*, 2024.

[19] M. R. Zehady, M. A. Hossain, and M. S. Islam, "Banglallama: Llama for bangla language," *arXiv preprint arXiv:2410.21200*, 2024.

[20] Z. Li et al., "Think carefully and check again! meta-generation unlocking llms for low-resource cross-lingual summarization," *arXiv preprint arXiv:2410.20021*, 2025.

[21] M. M. Raihan, A. Ahmed, and T. Hasan, "A family of bangla large language models," *arXiv preprint arXiv:2503.10995*, 2025.

[22] M. A. T. Rony and M. S. Islam, "Evaluating large language models for summarizing bangla texts," *Preprint*, 2025.