Thesis No. :

CSE 4000: Thesis

Bangla Text Summarization from Video Content using Sequential Method of Abstractive and Extractive Summarization

By

Farhatun Shama

Roll: 1907033

&

Lamisa Bintee Mizan Deya

Roll: 1907049

Department of Computer Science and Engineering

Khulna University of Engineering & Technology

Khulna 9203, Bangladesh

October, 2024

Bangla Text Summarization from Video Content using Sequential Method of Abstractive and Extractive Summarization

By

Farhatun Shama

Roll: 1907033

&

Lamisa Bintee Mizan Deya

Roll: 1907049

A thesis submitted in partial fulfillment of the requirements for the degree of

"Bachelor of Science in Computer Science & Engineering"

Supervisor:

Abdul Aziz
Assistant Professor
Department of Computer Science and Engineering
Khulna University of Engineering & Technology (KUET)

Signature

Department of Computer Science and Engineering

Khulna University of Engineering & Technology

Khulna 9203, Bangladesh

October, 2024

Acknowledgment

We express our deepest gratitude to Almighty Allah for His blessings. We extend sincere thanks to our supervisor, Abdul Aziz, Assistant Professor of Computer Science and Engineering, for his invaluable guidance, encouragement, and constructive feedback, which helped us overcome challenges throughout this thesis. We also thank Khulna University of Engineering & Technology for providing a supportive environment. Finally, we are grateful to our family, friends, classmates, and seniors for their continuous support and motivation. This thesis is dedicated to everyone who contributed to its success, with the hope it will positively impact our field and society.

Farhatun Shama

&

Lamisa Bintee Mizan Deya

Abstract

The rapid growth of video content in Bangla presents challenges, especially given the limited resources available for automated Bangla language processing. Watching long videos can be tedious, and the lack of robust summarization systems makes it difficult to extract meaningful information quickly. This research aims to address these issues by developing an efficient Bangla video summarization technique that retrieves text from videos and generates concise summaries. To achieve this, we propose a two-stage methodology that applies sequential abstractive summarization using BanglaT5, followed by extractive summarization with *Bangla-BERTSum*. This sequential approach leverages the strengths of both models—BanglaT5 restructures and summarizes informal spoken content, while *Bangla-BERTSum* ensures precision by selecting the most relevant sentences. This approach ensures context preservation and consistency by capturing the video's core messages while retaining relevant details from the text extraction stage. The result is a well-structured and coherent summary that effectively captures the essential content of Bangla videos, significantly reducing the viewing time required to obtain key information. This research contributes to society by providing an automated solution that helps users identify which videos are relevant, saving time and effort. It is particularly beneficial for time-constrained users, such as students, journalists, and researchers, who need quick access to key information from long videos. The research also promotes the development of Bangla language processing technologies, filling a critical gap in low-resource language automation. Ultimately, this work can enhance accessibility to Bangla video content by streamlining information retrieval and consumption.

Contents

List of Tables

List of Figures

CHAPTER I

Introduction

1.1 Introduction

In recent years, the explosion of multimedia content on the internet has led to an overwhelming amount of video data, particularly in regional languages like Bangla. Summarization is crucial for processing large datasets efficiently, providing concise information to help viewers, researchers, and educators quickly grasp key insights. While most summarization research focuses on textual data, summarizing spoken language or transcriptions from video content presents unique challenges. Unlike standard text summarization, this task involves processing spoken language, which can exhibit irregularities such as incomplete sentences, repetitions, and fillers.

Bangla, being a morphologically rich and syntactically complex language, presents additional challenges not adequately addressed by existing summarization tools. This language lacks resources as well. Directly applying extractive methods to spoken language can result in disjointed or incomplete summaries. This thesis, therefore, focuses on developing a Bangla text summarization system that processes audio-derived transcripts and subtitles. By combining abstractive and extractive techniques in a sequential manner, the model aims to generate more coherent, concise, and meaningful summaries, addressing the nuances specific to the Bangla language.

1.2 Background

The rise of digital video content has presented users with an overwhelming amount of information to sift through. Without proper summarization, viewers often struggle to find relevant content, wasting time watching irrelevant or redundant parts of videos. For example, in educational or informational videos, users may only need a specific portion of the content, but without a summary, identifying those segments becomes laborious.

Similarly, meetings or important discussions need to be summarized to keep track of key points and decisions, ensuring efficient follow-ups and better accountability. Video summarization helps tackle these issues by generating concise overviews, allowing viewers to quickly grasp the key topics discussed and locate relevant information efficiently.

However, summarizing video content, especially in Bangla, introduces additional challenges. Bangla is a low-resource language, meaning it lacks large-scale annotated datasets and advanced NLP tools that are available for languages like English [1]. This poses significant obstacles in developing effective summarization models, as existing state-of-the-art tools are not designed to handle the complexity of Bangla, leaving the language underserved in video summarization technologies. While unsupervised learning can be used for summarization, it lacks the accuracy and robustness of supervised approaches. Traditional extractive summarization methods work by selecting key sentences [2]. They often fall short when applied to spoken language, as they tend to produce disjointed or incomplete summaries. Speech contains informal elements, such as redundancy and repetitions, which extractive methods cannot handle effectively [3]. On the other hand, abstractive summarization generates summaries by paraphrasing the context [2]. So, it may result in inaccurate or irrelevant content that wasn't in the original video. A combined approach that leverages both extractive and abstractive techniques is necessary to handle spoken Bangla content, ensuring the summary is both accurate and coherent. These challenges highlight the need for specialized Bangla video summarization models that can address the unique problems faced by low-resource languages and spoken content.

1.3 Objectives

To address the challenges of low-resource Bangla language processing and the lack of effective video summarization systems, this thesis has the following objectives:

- Develop an efficient Bangla video summarization system by combining abstractive and extractive techniques to generate concise and meaningful summaries.
- Summarize video subtitles to help users quickly access essential information, reducing the time required to watch long videos.
- Handle spoken language data by extracting text from video content and effectively processing it for summarization.

- Preserve context and key messages in the summarization process using a two-stage methodology with non-overlapping abstractive chunk summaries and extractive refinement.
- Address challenges associated with low-resource Bangla language processing by proposing a novel automated summarization framework.
- Improve accessibility to Bangla video content, enabling users to identify relevant videos more efficiently.
- Contribute to the advancement of Bangla Natural Language Processing (NLP) technologies by implementing and finetuning summarization models, such as T5 and BERT.

1.4 Scope

This research leverages state-of-the-art tools and technologies to address the challenges of Bangla video summarization, including extracting spoken data from videos and generating concise summaries. The tools and libraries used span a wide range of domains, from audio processing to machine learning and natural language processing (NLP). Below is an overview of the tools used in this research:

A. Python and NLP Libraries: Core implementation was done using Python, with libraries such as Transformers from Hugging Face and pandas to facilitate text processing, model training, and data manipulation.
B. BanglaT5 for Abstractive Summarization: A transformer-based model fine-tuned for generating abstractive summaries in Bangla, ensuring context is preserved and long texts are concisely summarized.
C. Bangla-BERTSum for Extractive Summarization: A Bangla-specific BERT-based model used for selecting the most relevant sentences from abstractive summaries to generate a concise and meaningful final summary.
D. Speech Recognition Tools for Extracting Audio and Text:
    - SpeechRecognition (sr): Used for converting speech to text with the Google Speech Recognition API to process the spoken content in Bangla.
    - pydub: Used to handle audio files, including splitting audio into smaller chunks for easier processing.
    - MoviePy: Used to extract audio from video files efficiently.

E. Dataset Handling and Preprocessing: pandas and Hugging Face's Dataset library were used to preprocess data and split it into training and test datasets.
F. Google Colab with GPU Support: Google Colab was used for running the experiments, providing GPU resources for faster training and inference of the summarization models.

A combination of audio processing tools, transformer models, and machine learning frameworks can be effectively utilized to develop a Bangla video summarization system. These tools provide a comprehensive framework for processing spoken content, summarizing it accurately, and addressing the challenges of low-resource Bangla language processing.

1.5 Unfamiliarity of the Solution

The domain of Bangla video summarization is largely unexplored, with no existing solutions addressing this issue. Current Bangla text summarization focuses on formal, text-based materials like news articles, ignoring the growing amount of informal content from media such as video.

Our research addresses this gap by developing a system that summarizes video content based on text derived from speech. Most existing systems are limited to short-form, structured content and cannot effectively summarize long, unstructured texts or handle video data. Our system bridges this gap by summarizing both informal spoken language from academic videos and formal content from news videos, advancing multi-domain summarization where text varies in formality and complexity.

Summarizing long video transcripts, particularly in academic contexts, presents additional challenges because large volumes of text have to be considered while preserving core information.

To overcome these limitations, we employ a sequential method combining extractive (*Bangla-BERTSum*) and abstractive (BanglaT5) summarization, ensuring better handling of large video content. Our focus on Bangla video summarization introduces a novel approach, contributing to a largely unexplored research domain, and making it easier for users to quickly assess whether the content is useful for them. This approach saves time by providing

concise summaries, allowing users to determine the relevance of the data without going through lengthy videos.

1.6 Contribution

This thesis addresses the challenges of summarizing spoken and video-based content. By developing tailored datasets and fine-tuning state-of-the-art models, the aim is to fill existing gaps in Bangla language automation.

- A Bangla data-summary dataset has been developed for both abstractive and extractive summarization, providing a valuable resource for future research.
- The dataset and summarization procedure accounts for the complexity of spoken language, which may include informal, unstructured, and grammatically inconsistent content.
- A hybrid summarization approach is employed, combining abstractive and extractive techniques to generate concise and accurate video summaries.
- A Bangla extractive summarizer is proposed by adapting Bangla-BERT within the BERTSUM architecture to improve precision in sentence selection.
- The BanglaT5 model is fine-tuned with a large abstractive dataset, optimizing it for handling long, unstructured video content, ensuring high-quality summaries aligned with our specific task.
- The video-to-audio and audio-to-text conversion processes are integrated, ensuring smooth summarization and producing concise summaries of video content to save users' time.

1.7 Project Planning

The research has been developed gradually over a span of 13 weeks, with key milestones being achieved during this research. The progress made during these weeks is illustrated in the Gantt Chart in Figure 1.1, which highlights the time taken by different parts of the research. This has allowed us to monitor progress and make adjustments as necessary.

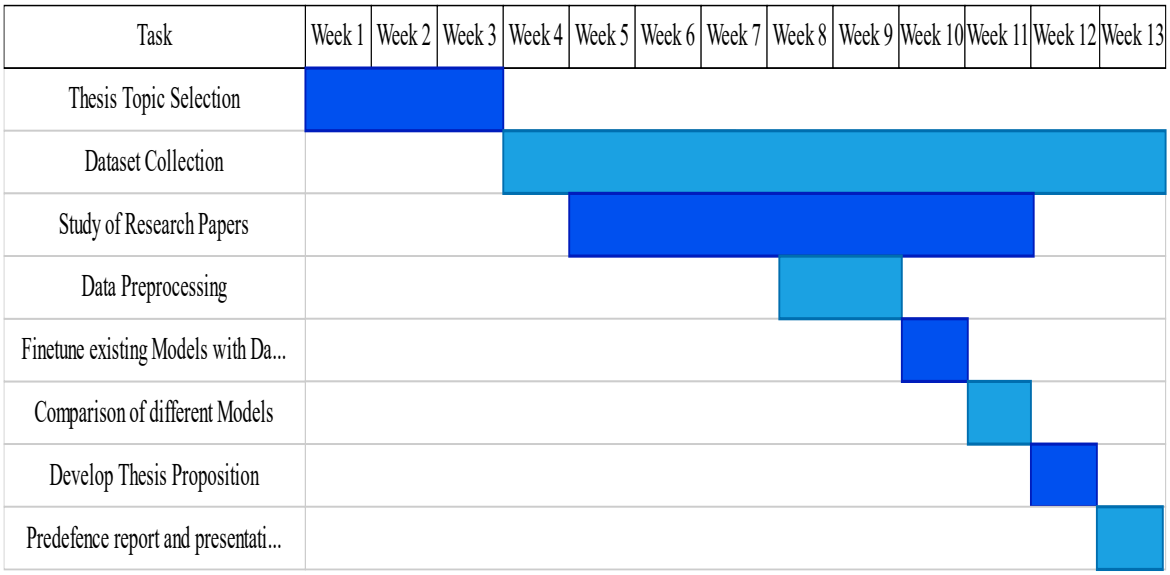

Figure 1. 1: Thesis progress in 4-1

Additionally, the Gantt Chart in Figure 1.2 outlines the next 13 weeks, detailing the upcoming tasks and milestones. This future plan helps in resource allocation, ensures that we stay on schedule, and provides a clear roadmap for completing the remaining phases of the research within the allotted timeframe.

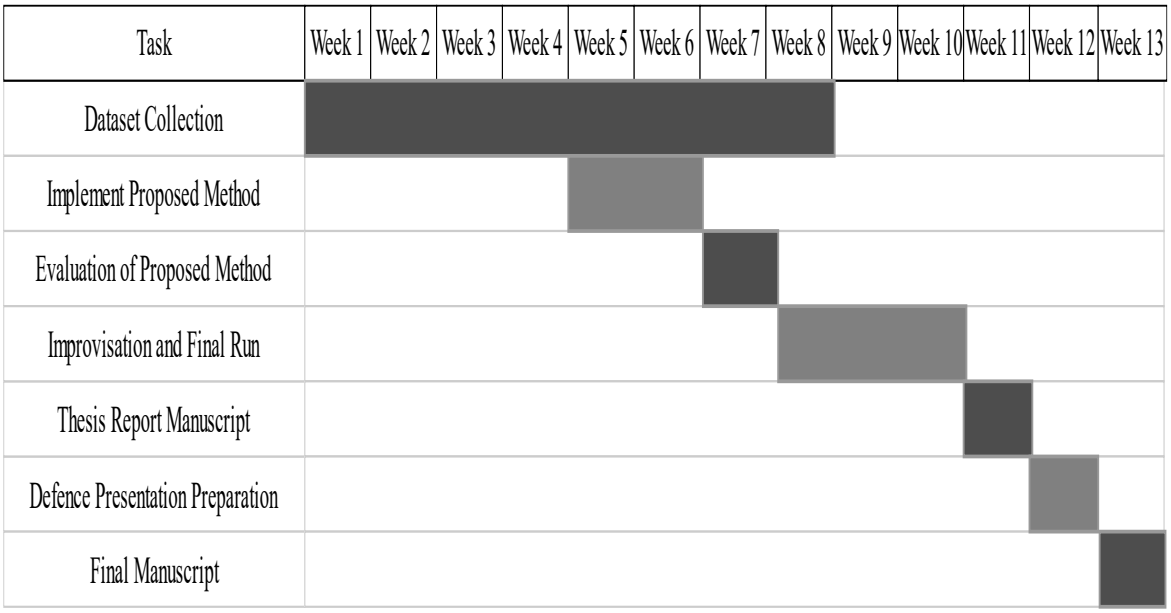

Figure 1. 2: Thesis planning in 4-2

From a societal perspective, the research has the potential to improve access to Bangla video content by providing concise summaries, which can help people decide quickly if the content is useful to them, saving time and enhancing necessary information consumption. In terms of health and safety, this system is primarily digital, so it does not directly pose

significant risks. However, ethical considerations must be made when summarizing sensitive or misleading information to avoid misrepresentation or misinformation.

Legally, the research must follow the copyright laws and ensure that summarized content from videos is appropriately handled, respecting the intellectual property rights of the original content creators. To adhere to this, we have strictly used copyright-free content in the research, ensuring that no legal boundaries are crossed.

1.8 Application of the Work

The summarization techniques developed in this research have several practical applications across different domains. By providing concise summaries of lengthy content, these methods offer significant time savings and improve the accessibility of information. Below are some key areas where our summarization model can be applied:

- News Summarization: Provides concise summaries of daily news, allowing users to quickly grasp key updates without going through lengthy articles or reports, thus saving valuable time.
- Educational Content Summarization: Helps users determine whether the content is relevant to their topics of interest, facilitating easier filtering of large volumes of academic or educational videos.
- Contextual Understanding: By accurately capturing the context of videos, users can get a complete overview of the content without watching the entire video, improving efficiency in content consumption.
- Missed Online Meetings: In the academic domain, summarization allows individuals to easily catch up on missed meetings or lectures by reviewing summaries, eliminating the need to watch the full content.
- Research and Knowledge Sharing: Summarized content can be shared among researchers or professionals, enhancing collaboration and facilitating the sharing of key insights.
- Multitasking and Accessibility: Summarized videos enable users to quickly absorb information, making it ideal for individuals who are multitasking or have limited time, thus improving accessibility.

These are the novel applications of the thesis.

1.9 Organization of Report

The structure of this report is organized as follows: Chapter II reviews relevant prior studies related to this research. Chapter III details the comprehensive methodology employed throughout the study. Chapter IV presents the results and provides an in-depth discussion of the findings. Lastly, Chapter V concludes the report with a clear and concise summary.

1.10 Conclusion

In conclusion, this research addresses the gap in Bangla video summarization by developing a novel system that combines extractive and abstractive methods to efficiently summarize both formal and informal video content. By leveraging Bangla-BERTSum and BanglaT5, the proposed approach ensures accurate, context-aware summaries that reduce the time and effort needed to process large video data. The following chapters will explore the related work, methodology, implementation, and results, providing a comprehensive evaluation of the system's effectiveness and its broader applications.Education: Helps students learn by asking questions about images.

CHAPTER II

Literature Review

2.1 Introduction

The field of text summarization has seen significant advancements over the years, driven by the increasing demand for efficient information processing in various domains. Early work in summarization largely focused on extractive methods, which involve selecting key sentences from the original text. However, with the development of more sophisticated machine learning techniques, abstractive summarization has emerged as a more advanced approach that generates new sentences based on the content's meaning. In the context of Bangla language processing, research is still in its early stages, with most studies focusing on summarizing formal written text such as news articles. Few works have explored the integration of video content and the challenges associated with informal data, such as educational or academic videos. This literature review explores the key developments in both extractive and abstractive summarization methods, particularly focusing on their application in Bangla text processing. It highlights the current gaps in video content summarization and identifies the need for a system that integrates informal and formal text, which this research aims to address.

2.2 Literature Review

Text summarization, especially for low-resource languages like Bengali, has garnered increasing attention in recent years. While most research focuses on extractive summarization, abstractive methods are emerging as more effective for generating summaries that resemble human-written text. Talukder et al. (2019) [4] introduced an abstractive text summarization model for Bengali, using a sequence-to-sequence architecture with bi-directional RNNs and LSTM units. The model applies an attention mechanism in the decoding phase to generate fluent summaries from sources such as news articles and social media posts. Preprocessing techniques like vocabulary counting and text

cleaning were implemented to handle missing and unknown words, essential for improving the model's performance. The authors collected their own dataset due to the lack of available resources for Bengali, reflecting a common challenge in Bengali NLP tasks. Their model demonstrated significant success, reducing training loss to 0.008. While the work makes a substantial contribution to Bengali abstractive summarization, especially with its use of sequence-to-sequence with attention, the authors note the need for larger datasets and further optimization for longer texts.

Bhattacharjee et al. (2019) [5] presented the Bengali Abstractive News Summarization (BANS) model using an LSTM-based encoder-decoder architecture with an attention mechanism for summarizing Bengali news articles. The model generates human-like summaries by capturing key information from the original document and producing fluent sentences. A key contribution of their work is the development of the largest publicly available Bengali news summarization dataset, consisting of 19,096 news articles and corresponding summaries, collected from bangla.bdnews24.com. The dataset was made publicly accessible on Kaggle. The model was evaluated both qualitatively and quantitatively, achieving notable results in terms of BLEU and ROUGE scores. This work significantly advances Bengali abstractive summarization by providing a large-scale dataset and presenting a well-performing attention-based model, but it highlights challenges such as repetition in generated summaries and inaccurate reproduction of details, suggesting future work in improving these limitations.

Chowdhury et al. (2021) [1] proposed *BenSumm*, an unsupervised abstractive summarization model for Bengali text documents. Unlike previous works, this model operates without any parallel document-summary pairs, relying instead on POS tagging and a pre-trained language model. The model constructs word graphs and uses sentence fusion to generate abstractive summaries. It was evaluated on a human-annotated dataset consisting of 139 samples from the National Curriculum and Textbook Board (NCTB), and demonstrated superior performance compared to extractive baselines, especially on ROUGE-1 and ROUGE-L scores. This work makes a significant contribution by eliminating the dependency on large annotated datasets, making it more suitable for low-resource languages like Bengali. The authors also include difficulty in creating entirely new words or phrases, leading to less varied summaries. Without large annotated datasets, it may struggle with handling complex texts effectively.

The paper [6] presents BART, a versatile transformer-based model that merges the bidirectional encoder of BERT and the autoregressive decoder of GPT. BART is pretrained using a denoising autoencoder approach, where the input text is corrupted through methods like text infilling and sentence shuffling, and the model learns to reconstruct the original text. This pretraining allows BART to excel in various NLP tasks, such as abstractive summarization, machine translation, and dialogue generation, achieving state-of-the-art results on benchmarks like CNN/DailyMail and *XSum*.

Bhattacharjee et al. (2023) [7] introduced BanglaT5, a pre-trained sequence-to-sequence model designed specifically for Bangla, achieving state-of-the-art performance in various Natural Language Generation (NLG) tasks, including text summarization, question answering, and translation. Fine-tuned on a range of formal and informal datasets, BanglaT5 outperformed popular multilingual models such as mT5 and *mBART*. The model's architecture is based on Google's T5 framework, adapted to Bengali language-specific tasks, allowing it to handle both structured news content and more informal text like social media or academic videos. Its strong performance across these domains highlights its robustness and adaptability, setting a new benchmark for Bengali language processing and opening new opportunities for future research in low-resource languages.

The authors [8] propose a transformer-based approach for generating abstractive summaries in the Bengali language. Recognizing the need for robust automated summarization techniques, especially given the increasing amount of digital textual data, the authors aim to bridge the gap between high-resource and low-resource languages like Bengali, where most prior summarization work has been extractive. The study evaluates five different transformer models, including B-T5, B-T5-base, mT5-small, mT5-base, and mBART-50, using publicly available datasets like *XLSum* and BANS. The authors fine-tune these models and observe their performance across various evaluation metrics. The B-T5 model, fine-tuned on the *XLSum* and BANS datasets, consistently outperforms other models in terms of ROUGE scores, particularly achieving a ROUGE-2 score of 13.83 on the merged dataset, demonstrating a 28.29% improvement from *XLSum* data alone. The models were trained using standard transformer training techniques, leveraging Adam and *Adafactor* optimizers with appropriate hyperparameter settings to achieve optimal performance. The paper also discusses error analysis, noting issues such as omission, redundancy, and factual inaccuracies in the generated summaries. For my project, I have used B-T5 as the base

model for abstractive summarization in Bengali, given its superior performance in both monolingual settings and its ability to handle lengthy text sequences with enhanced accuracy and reduced redundancy. Still, this paper introduces some limitations as it is mostly focused on formal text.

Extractive summarization is needed for its accuracy, simplicity, and reliability in preserving original content without introducing inaccuracies. An LSTM-based encoder-decoder model [9] was introduced for extractive text summarization, utilizing the CNN news article dataset to extract key sentences from lengthy texts. The model scores sentences based on their importance, enabling the selection of significant sentences for the summary. Evaluated using metrics such as ROUGE-1 and ROUGE-2, the proposed approach achieved an average F1-Score of 0.8353, thus demonstrating its effectiveness in generating concise and meaningful summaries from extensive content.

The research in [10] introduces a method for extractive text summarization of lecture transcripts using BERT (Bidirectional Encoder Representations from Transformers). The methodology leverages BERT's capability to generate contextual embeddings for sentences within the transcript and employs K-Means clustering to group similar sentences, enabling the extraction of significant sentences that best represent the content. The approach is evaluated against traditional extractive summarization methods using standard metrics, demonstrating improved performance in summary accuracy and coherence. However, it may not surpass supervised techniques that fine-tune BERT for specific summarization tasks, as supervised methods benefit from labeled training data that allow for tailored learning of essential features and patterns, which the unsupervised nature of Miller's method may not effectively capture, potentially leading to less precise summaries compared to fully supervised approaches.

The paper [11] introduces BERT (Bidirectional Encoder Representations from Transformers), a novel language representation model that pre-trains deep bidirectional representations by jointly conditioning on both left and right contexts in all layers. This allows for fine-tuning with minimal additional parameters, resulting in state-of-the-art performance across various natural language processing tasks, including question answering and language inference. BERT achieves impressive scores, such as 80.5\% on GLUE and 93.2 F1 on *SQuAD* v1.1, highlighting its effectiveness over unidirectional models. The study

emphasizes that BERT's bidirectional approach enhances contextual understanding, making it a powerful foundation for numerous NLP applications.

The authors in [12] present a method for extractive text summarization utilizing BERT (Bidirectional Encoder Representations from Transformers), which includes a document-level encoder designed to capture the semantics of the text and generate sentence representations. Their approach enhances sentence selection by stacking inter-sentence Transformer layers, allowing the model to better understand relationships within the text. Evaluated on the CNN/DailyMail and NYT datasets, their method achieves state-of-the-art results, surpassing pure BERT summarization techniques by effectively leveraging BERT's bidirectional representations to improve the scoring and selection of important sentences, leading to higher-quality summaries.

The paper [13] introduces *BERTSum*, an extractive summarization model that leverages the capabilities of BERT (Bidirectional Encoder Representations from Transformers) to enhance sentence selection for summarization tasks. *BERTSum* achieves state-of-the-art results on benchmark datasets, including CNN/DailyMail, by significantly improving ROUGE scores compared to previous extractive summarization models. Its architecture not only enhances the extraction process but also outperforms existing approaches by utilizing BERT's rich contextual embeddings, thus demonstrating superior performance in producing coherent and high-quality summaries.

The author [14] evaluate ChatGPT's performance in extractive summarization, focusing on its ability to condense lengthy documents into concise summaries by directly selecting key sentences. The study compares ChatGPT against traditional fine-tuning methods across various benchmark datasets, revealing that while ChatGPT falls short in ROUGE scores compared to existing supervised systems, it excels in LLM-based evaluation metrics. Despite these advancements, the results indicate that ChatGPT does not surpass *BERTSum* in extractive summarization, as *BERTSum* demonstrates superior performance metrics and a more structured approach to sentence selection.

In the paper [15] on generating extended summaries of long documents, the authors present a multi-task learning model that integrates the hierarchical structure of documents into an extractive summarization framework. This model leverages section prediction alongside sentence selection to enhance summary quality over traditional methods, particularly for

complex and detailed documents like scientific papers. They demonstrate that while traditional extractive methods may excel in capturing key points concisely, they often fail to provide detailed insights for longer documents or those with rich structures. This inadequacy stems from the extractive approach's limitation to surface-level content, which can overlook nuanced information distributed across different sections of a document. By incorporating multi-task learning, their model aims to address these gaps by not only selecting salient sentences but also understanding the structural context within the document, which is crucial for summarizing formal and extensive texts effectively. But extractive summarization methods may struggle with informal data due to their reliance on structured content to identify key points, often failing to interpret colloquial language and implicit meanings effectively. In contrast, abstractive summarization can generate new text, better handling the nuances and scattered information typical in informal communications.

The paper [16] introduced Bangla-*ExtraSum*, a study focusing on extractive text summarization for the Bengali language. The paper highlights the scarcity of extractive summarization models and datasets for Bengali, despite the increasing demand for summarization in areas like news portals. The authors propose various methodologies that leverage semantic and contextual relationships between sentences to enhance summarization quality. They conducted experiments using five different models on two datasets, including 500 articles with human-generated summaries. The proposed model achieved an F-score of 0.68 on an existing dataset and 0.63 on their newly introduced dataset. This work emphasizes the importance of semantic relations in summarization and provides a comprehensive analysis of the methodologies, showcasing improvements over existing approaches in Bengali text summarization.

The authors in [17] introduce *Bangla-BERT*, a ELECTRA-based model specifically pretrained for Natural Language Understanding (NLU) tasks in the Bangla language, which is often considered low-resource in the NLP domain. To create BanglaBERT, the authors collected 27.5 GB of training data, referred to as ‘Bangla2B+’, by crawling 110 popular Bangla websites. They also developed *BanglishBERT*, which is pretrained on both Bangla and English data to facilitate zero-shot transfer learning between the two languages. However since this model is based on ELECTRA, It also cannot inherently summarize - like BERT. Also ELECTRA lacks the next sentence prediction mechanism of BERT, overall,

the summarization capability of this model is untested. Also the *Bangla-BERTSum* of same name developed in paper [18] is developed on a much larger dataset.

Bangla-BERT [18] is a transformative model in Bengali natural language processing, utilizing a monolingual BERT architecture specifically tailored to grasp the intricate linguistic characteristics of Bengali. Trained on 'BanglaLM', an expansive 40 GB dataset—the largest dataset ever used for a Bengali language model—Bangla-BERT harnesses the robust capabilities of the BERT architecture to provide superior language understanding. This extensive training allows it to excel in a variety of NLP tasks, establishing new performance benchmarks and significantly advancing the digital processing capabilities for the Bengali language. It cannot inherently summarize. This surpasses all existing Bangla language models in terms of pretraining data and performance across several NLU tasks.

The research [19] investigates the performance of Multilingual BERT (mBERT) across a broader spectrum of languages, particularly focusing on low-resource languages. While *mBERT*, trained on 104 languages, demonstrates strong cross-lingual capabilities and performs comparably to monolingual models on high-resource languages, its performance significantly declines for low-resource languages in tasks such as Named Entity Recognition, Part-of-Speech Tagging, and Dependency Parsing. The authors highlight that although *mBERT* can leverage zero-shot learning for these tasks, it still struggles with low-resource languages due to insufficient pretraining data and representation quality. They suggest that enhancing model performance for low-resource languages necessitates more effective pretraining techniques or larger datasets. Consequently, while *mBERT* shows potential, particularly with more data, it remains uncertain if it can outperform specialized models like Bangla-BERT without substantial improvements in training methodologies. *mBERT* cannot inherently do summarization.

The paper [20] presented a hybrid approach for Bengali news summarization using the T5 Transformer combined with extractive techniques. The model was fine-tuned to generate coherent and balanced summaries, integrating extractive elements to preserve key sentences while allowing for natural language generation. They proposed to generate extractive summaries first and then abstractive summaries. While the hybrid approach improved readability and contextual accuracy, challenges such as repetition and handling longer texts were identified, suggesting further fine-tuning and dataset expansion for better results.

2.3 Discussion

Table 2.1: Summary of The Existing Studies on Abstractive Summarization

Models	Key Approach	Dataset	Limitations	Research gaps
Seq2Seq LSTM with Attention [4]	Bi-directional LSTM with attention mechanism	Manually collected Bengali texts	Struggles with long texts; limited dataset	Lacks handling of informal content and video-based data
Seq2Seq RNNs [5]	Sequence-to-sequence RNNs with attention	Manually collected texts	Requires more optimization for longer texts	Focuses on formal text, does not handle informal video data
Unsupervised Learning (*BenSumm*) [1]	POS tagging, sentence fusion using word graphs	NCTB Bengali Text Dataset	Cannot generate new words/phrases; issues with long texts	No parallel dataset, works only with unsupervised methods
BART [6]	Pretrained transformer model for text summarization	General-purpose multilingual data	Trained on multilingual data, not optimized for Bangla	Lacks fine-tuning for Bangla; struggles with informal data
BanglaT5 [7]	Pretrained sequence-to-sequence model for Bangla	Diverse Bengali datasets	Limited data availability for informal content	Best model for Bengali tasks but needs video content-specific tuning

Table 2.2: Summary of The Existing Studies on Extractive Summarization

Model	Key Approach	Dataset	Limitations	Research Gaps
Unsupervised (BERT based) Summarization [11]	Extractive summarization using BERT with K-Means clustering	Lecture transcripts	May not surpass supervised techniques, less precise summaries possible.	Exploration of supervised fine-tuning techniques for better feature learning.
BERTSUM [13]	Modified BERT based model for summarization	CNN/DailyMail datasets	Requires substantial training data	Investigation of BERTSUM's adaptability in less common languages.

For abstractive summarization, BanglaT5 has the best performance on Bangla. For extractive summarization *BERTSum* performs great. But it is not available in Bangla. Some Bangla pretrained models are Bangla-BERT, Banglish-BERT and *mBERT*. So for supervised Bangla extractive summarization, a hybrid approach can be used. Table 2.1 represents different abstractive summarization approaches while Table 2.2 shows different extractive summary approaches. Table 2.3 shows Bangla pre-trained models.

Table 2.3: Summary of The Existing Studies on Extractive Summarization (Bangla)

Model	Key Approach	Dataset	Limitations	Research Gaps
Bangla-BERT [19]	BERT-based model pretrained for NLU tasks in Bangla	40GB data of BanglaLM	Not specifically trained for summarization tasks.	Exploration of more NLU tasks and improvements in low-resource scenarios.

Table 2.3: Summary of The Existing Studies on Extractive Summarization (Bangla) (Cont.)

Model	Key Approach	Dataset	Limitations	Research Gaps
Banglish-BERT [18]	Pretrained on both Bangla and English data for zero-shot learning	Mixed Bangla and English datasets	May not perform optimally for Bangla tasks.	Improvement of cross-lingual transfer learning techniques.
mBERT [20]	Performance analysis of *mBERT* on low-resource languages	Various NLP tasks on 104 languages	Significant performance decline for low-resource languages	Enhancement of pretraining techniques or larger datasets for low-resource languages.

2.5 Conclusion

The review of existing models highlights the advancements made in Bengali text summarization, with notable contributions in both extractive and abstractive techniques. Despite these developments, several gaps remain, particularly in handling informal text such as video content and addressing challenges in longer text summarization. Models like Seq2Seq and T5 Transformer have shown promise, but they often struggle with dataset limitations and generating diverse summaries for more complex content. Our research aims to bridge these gaps by integrating both abstractive (BanglaT5) and extractive (*Bangla-BERTSum*) models in a sequential manner, specifically targeting informal video-based text summarization. By leveraging custom datasets and combining the strengths of existing models, we propose a solution that addresses both contextual accuracy and summary coherence, filling a critical void in Bengali language processing.

CHAPTER III

Methodology

3.1 Introduction

The methodology chapter outlines the approach used to develop a novel Bangla video summarization system by combining abstractive and extractive summarization techniques. The goal of this research is to produce accurate, concise, and coherent summaries for both formal and informal video content by leveraging the strengths of the Bangla-T5 and Bangla-*BERTSum* models. This chapter will detail the steps taken to fine-tune the pre-trained models for Bangla, the process of data collection for both formal and informal domains, and the preprocessing techniques applied to ensure the quality of the input data. Additionally, the sequential combination of abstractive and extractive methods will be explained, followed by a discussion of the tools and libraries used for audio extraction and speech-to-text conversion from video content.

3.2 Detailed Methodology

3.2.1 Video to Audio

The first step in the summarization process involves extracting the audio from video files. Audio extraction from video involves separating the audio track embedded within a video file format. To achieve this, we used Python libraries such as *MoviePy* and *Pydub*, which allow for easy manipulation of video and audio files. The *MoviePy* library is used to load the video and to access the video file's internal structure, where the audio stream is stored. They parse the video container format (such as MP4, AVI, etc.) to locate the audio data, which is often compressed. Once the audio stream is identified, these libraries can decode

the audio data and save it as a separate audio file in formats such as MP3 or WAV. The extracted audio will later be processed to convert speech into text for summarization.

3.2.2 Audio to Text

The audio-to-text conversion process begins by loading the extracted audio file using *Pydub* and splitting it into smaller, manageable chunks (3 minutes each) to ensure efficient processing. Each chunk is then transcribed into Bengali text using Google's Speech Recognition API via the *SpeechRecognition* library. The API converts spoken words in the audio into text, with any unrecognized portions (due to unclear audio) being skipped to maintain flow. Once all chunks are processed, the complete transcription is saved to a text file for further use, such as summarization. This method ensures the efficient handling of long audio files and accurate transcription.

Inside the API, the audio data undergoes a systematic series of processing steps to convert spoken language into text. Initially, the audio signal is subjected to feature extraction, where distinctive characteristics are analyzed to represent the spoken content. These extracted features are then matched against an acoustic model specifically trained on a variety of languages and accents, allowing the model to interpret sound patterns and map them to phonetic transcriptions. Following this, a language model analyzes the context of the recognized words, predicting the likelihood of word sequences and enhancing transcription accuracy by recognizing common phrases and grammatical structures. The processed audio features are subsequently decoded into text, yielding the most probable transcription of the spoken content. After completing the transcription, the API sends the results back as a response to the *SpeechRecognition* library, which then compiles and saves the entire transcription into a text file for further use.

3.2.3 Preprocessing of Bangla Transcribed Text

- **Adding Punctuation to Transcribed Text**

After the audio-to-text conversion, the generated text lacks proper punctuation (such as periods, commas, and question marks) since speech recognition models typically do not include these in the transcription. To enhance the readability and coherence of the text, punctuation marks must be added. This step is essential for improving the quality of the text input before it can be used for summarization.

For this, we utilize the *DeepPunct* model, a punctuation restoration model trained on large datasets to predict where punctuation marks should be inserted based on the structure and flow of the text. *DeepPunct* relies on machine learning techniques to analyze sentence boundaries, pauses, and other linguistic cues from the transcription, automatically adding punctuation in appropriate places. This model has been shown to perform well in accurately restoring punctuation in long, unstructured text, making it ideal for use in this context.

By ensuring the transcribed text is properly punctuated, the overall semantic structure is preserved, which is crucial for downstream tasks such as summarization. Proper punctuation helps the summarization models, like Bangla-T5 and *Bangla-BERTSum*, to better understand the sentence boundaries and context within the text, leading to more coherent and meaningful summaries. Models like BERT will perform poorly without punctuation as the punctuation is a crucial step for segment embedding. Without punctuation, the summarization process may result in incomplete or unclear summaries, as the model could misinterpret where one thought ends and another begins.

- **Normalization of Text**

To ensure consistency and improve the quality of the transcribed Bangla text, we employed the BNLTK (Bengali Natural Language Toolkit) for text normalization. This process involved standardizing spelling variations, removing unnecessary diacritics, and handling colloquial expressions by converting them to their formal equivalents. BNLTK also allowed us to tokenize the text and eliminate redundant whitespace and special characters. By automating these steps, we ensured that the input text was clean, uniform, and ready for the summarization model, significantly enhancing the accuracy and coherence of the generated summaries.

- **Stop words Removal**

To optimize the transcribed Bangla text for summarization, we performed stopword removal using Python. Common Bangla stopwords, such as ও, এবং, কিন্তু, এই, সে, তারা, তুমি, আমি, আমরা, তোমরা, যার, যে, যা, যদি, তখন, কেন, কোথায়, এখানে, সেখানে, যেমন, দ্বারা, উপর, থেকে, নিচে, সব, জন্য, মধ্যে, সাথে, করতে, করলো, ছিল, হলো, করতে হবে, হয়, ছিলাম, থাকবে, হয়েছে, ছিলো, বুঝছো, বুঝছিস, and কিভাবে were filtered out as they do not add significant meaning to the sentences. Removing these frequent but uninformative words allowed the

model to focus on the more critical content, such as specific subject terms and key information relevant to the summarization process.

❖ Handling Non-Bengali Words and Noise

When using Google's Speech Recognition API for transcribing Bangla speech, the output is typically in Bangla script, but any numbers in the transcription are generated in English numerals. To ensure full consistency in the transcribed text, it's important to convert these English numerals into their Bangla equivalents. This can be achieved by implementing a number mapping dictionary in Python that automatically replaces English numerals (e.g., 0, 1, 2, etc.) with their corresponding Bangla numerals (e.g., ০, ১, ২, etc.). This step is critical for maintaining uniformity throughout the text and ensuring that all elements are represented in the Bangla script, which is especially important for improving the readability and coherence of the data, particularly in formal and academic contexts. Additionally, this approach ensures that the transcribed dataset is clean and ready for subsequent processing steps, such as summarization, without unnecessary distractions caused by mixed-language elements.

❖ Splitting in Chunks

To effectively manage the large volume of transcribed text from the video, the text is split into chunks, with each chunk containing a maximum of 6 sentences. Sentence boundaries are recognized using the punctuation marks ।, !, and ? as separators. After the summarization process, the generated summaries are concatenated into a single text file, and the summarized text is then further divided into smaller chunks, with each chunk containing a maximum of 4 sentences. The length of the summary for each chunk varies depending on the model used: for Bangla-T5, the maximum summary length is around 100 words, for *Bangla-BERTSum* (extractive), it's approximately 150 words, and for models like Seq2Seq LSTM or RNNs, the summary length typically ranges between 120-140 words per chunk. This structured chunking and summarization process ensures better organization of the text and improves the model’s performance, especially when dealing with longer video content, making the text more manageable and easier to process. This approach also enhances the organization and clarity of the summarized text, making it more suitable for further analysis or direct consumption by users.

3.2.4 Dataset Collection

Since there is no large-scale standard public dataset available for Bengali abstractive summarization, we referenced the dataset collection method used by Bhattacharjee et al. ref They collected a significant dataset from the online news portal bangla.bdnews24.com, which includes both news articles and their summaries. Using a web crawler, they scraped 19,352 news articles from various categories such as sports, politics, and economics. The raw data initially contained advertisements, non-Bengali words, and irrelevant links, which were cleaned through a data cleaning program to ensure the dataset's quality. This processed dataset was later made publicly available on Kaggle [5]. While we did not replicate this exact data collection process, we have used this dataset as a reference for our work, supplementing it with additional data manually collected from informal sources such as academic video transcripts. This combined dataset allows us to fine-tune our models for Bengali abstractive summarization, incorporating both formal and informal content to enrich the diversity of our input data. We applied a similar data-cleaning process.

In addition to utilizing this dataset as a reference, for extractive summarization we collected 200 data points, each with three summaries from a public dataset named BNLPC, which was created specifically for Bengali summarization. Furthermore, for extractive summarization, we collected 100 data points, each with one corresponding summary. All of the collected data underwent a thorough cleaning process to remove non-textual elements, irrelevant content, and other inconsistencies, ensuring the dataset's suitability for model training.

That's how we have collected two datasets.

3.2.5 Model Design

This proposed methodology section covers three portions.

- **Bangla-T5 for Abstractive Summarization**

The proposed model uses the Text-to-Text Transfer Transformer (T5) architecture as the foundation for Bangla abstractive summarization. The T5 model, originally designed by Google, is based on a sequence-to-sequence (seq2seq) architecture that converts text input

to text output. The primary phases of the architecture include the encoder, decoder, and output generation, as illustrated in Figure 3.1.

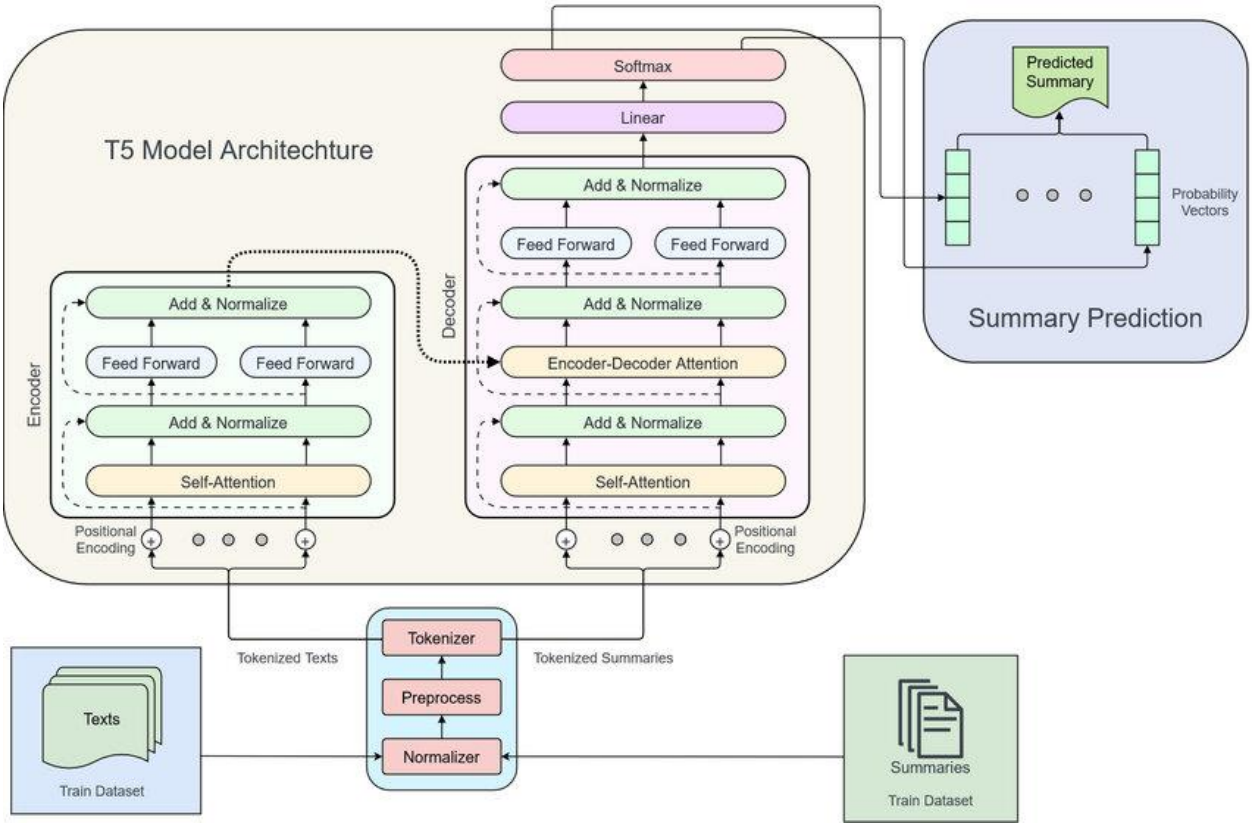

Figure 3. 1: Overview of abstractive text summarization model [8]

i) Encoder

The encoder consists of a stack of identical layers, each comprising two sub-layers:

- **Self-Attention Layer:** This layer enables the model to focus on different parts of the input text when generating the summary. It processes the relationships between words to capture context more effectively.
- **Feed-Forward Neural Network:** After the self-attention layer, the output is passed through a feed-forward neural network that enhances the model's ability to process nonlinear transformations.
- **Layer Normalization:** The encoder applies layer normalization at each step, ensuring stable learning and improving the efficiency of the model. This step ensures the data is standardized across layers, making it easier for the model to understand the text.

ii) Decoder

The decoder mirrors the encoder but with an additional layer to attend to the encoder's output:

- **Encoder-Decoder Attention:** This attention mechanism allows the decoder to focus on relevant parts of the encoded input sequence during the summary generation process.
- **Feed-Forward Neural Network:** As with the encoder, the feed-forward network processes the data further in the decoder to ensure the output sequence is coherent.
- **Repetition of Process:** This process continues until the decoder generates the entire summary, repeating until a stop signal is reached.

iii) Linear and Softmax Layer

- **Linear Layer:** The output from the decoder is passed through a linear layer that converts the decoder output into a logits vector.
- **Softmax Layer:** The logits vector is then passed through a softmax function, which converts the scores into probabilities, indicating the most likely words for the summary.

iv) Text Normalization

To improve the quality of the input data, the text is normalized using the Bangla normalization module (csebuetnlp/normalizer). This process ensures consistency in the format of the input text, standardizing it so that the model can better understand it. This step is particularly important in low-resource languages like Bangla, where text variations may affect the model's performance.

v) Tokenization

Before passing the text into the model, it undergoes tokenization using the *SentencePiece* algorithm. Tokenization breaks down the text into smaller, manageable units (tokens), making it easier for the model to process. The *SentencePiece* tokenizer is effective in handling large vocabularies, especially in multilingual and low-resource language settings.

❖ ***Bangla-BERTSum***

Our proposed model for extractive summarization is based on two models – Bangla-BERT and *BERTSum*. The detailed architecture of these models are given below:

i) Bangla-BERT

Bangla BERT is built by pretraining BERT on Bangla. BERT's architecture is based on the Transformer model, which relies on an attention mechanism to draw global dependencies between input and output. The Transformer model in BERT consists solely of attention layers and feed-forward layers—specifically, the encoder part of the Transformer. The architecture is given in Figure 3.2.

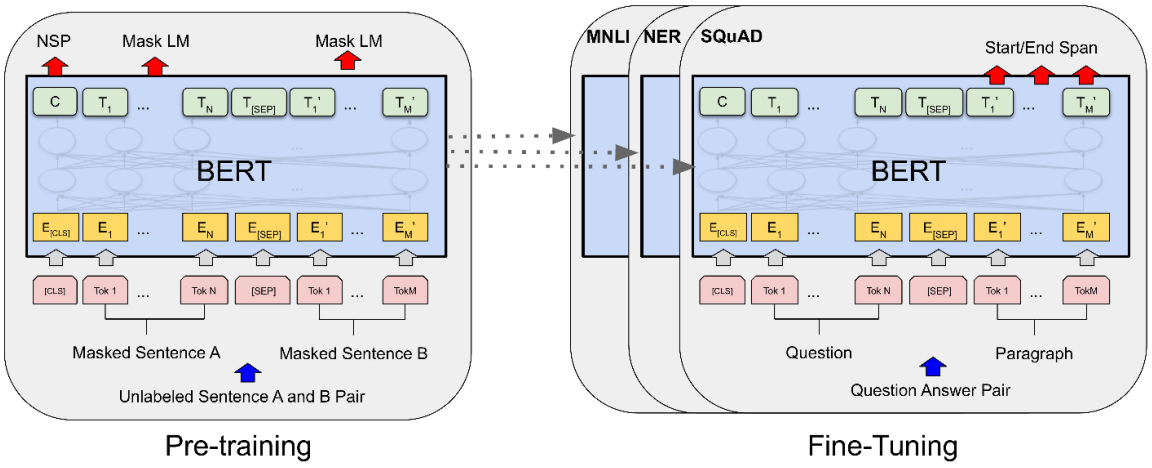

Figure 3. 2: Architecture of BERT model [12]

Components of BERT:

- **Transformers (Encoders only):** BERT utilizes the encoder stack of the Transformer model. The encoder reads the entire sequence of words at once. This mechanism allows BERT to capture the context of a word based on all of its surroundings (left and right of the word).
- **Self-Attention Mechanism:** The key component that allows BERT to understand the context of a word in relation to every other word in the sentence. Unlike directional models, which read the text input sequentially (left-to-right or right-to-left), BERT reads the entire sequence of words at once. Therefore, it is considered bidirectional, though it's more accurate to say it's non-directional.

- **Positional Embeddings:** BERT uses positional embeddings to express the position of words in a sentence, adding this to the input embeddings to preserve the order of words.

BERT's training process involves two main strategies:

1. Pre-training:

BERT is pre-trained on a large corpus of text on two unsupervised tasks:

Masked Language Model (MLM): Random words in each sentence are replaced with a [MASK] token, and the model must predict the original word based only on its context. Unlike traditional sequential models, this allows BERT to freely encode the context of a word based on both its left and right surroundings (i.e., bidirectionally).

Next Sentence Prediction (NSP): Given pairs of sentences as input, BERT must predict whether the second sentence in the pair is the subsequent sentence in the original document. This task trains BERT to understand sentence relationships, which is beneficial for tasks that require understanding the relationship between sentences, such as question answering and natural language inference.

Bangla BERT model is pretrained specifically for the Bangla language on a dataset named BanglaLM. This dataset comprises 40 GB of text data, making it the largest Bangla language model dataset used for training Bangla-BERT to date.

2. Fine-tuning:

After pre-training, BERT can be fine-tuned with just one additional output layer for a wide range of tasks, without substantial task-specific modifications. During fine-tuning, BERT is trained on a smaller dataset specific to a given task, such as sentiment analysis or question answering. Here, all parameters of BERT are fine-tuned to optimize performance on this downstream task.

ii) *BERTSum*

BERTSum builds on the original BERT architecture, which is based on a multi-layer bidirectional Transformer encoder. Each layer of the Transformer uses self-attention

mechanisms that allow the model to weigh the importance of different words within the same sentence or across different sentences, based on the task requirements. The architecture is illustrated in Figure 3.3.

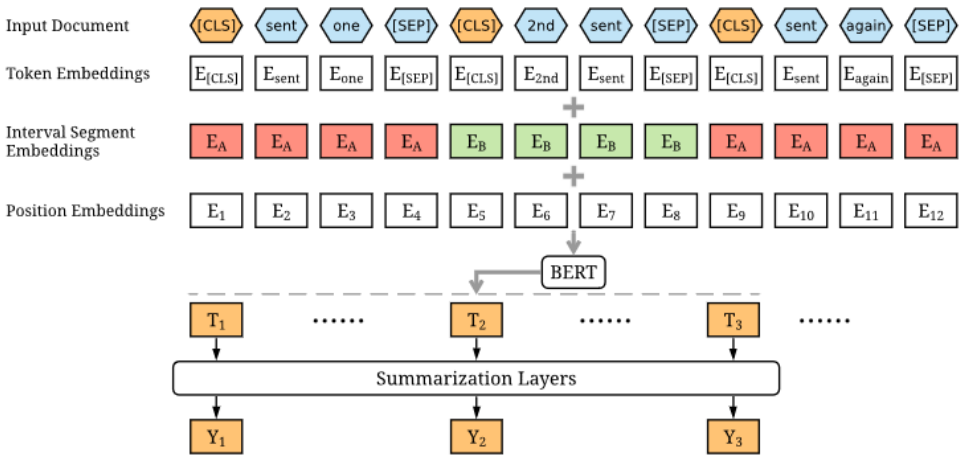

Figure 3. 3: Architecture of *BERTSum* model [13]

Document Encoding:

BERT as Encoder: *BERTSum* uses the BERT model as the backbone for encoding text. The deep bidirectional nature of BERT, built on the Transformer architecture, makes it exceptionally good at understanding context and nuances in the text.

Modifications on BERT for Summarization:

Segment Embeddings: *BERTSum* introduces segment embeddings that are different from the original BERT. These are used to encode sentences in a document distinctly, aiding the model in recognizing sentence boundaries within longer texts.

[CLS] Tokens for Sentences: In extractive summarization, *BERTSum* modifies the input by inserting [CLS] tokens at the beginning of every sentence instead of just at the beginning of the document. This approach allows the model to use the output from each [CLS] token as a representation of the corresponding sentence, making it particularly useful for sentence-level classification tasks.

These are significant for handling Bangla as well as summarizing properly.

Extractive Summarization Specifics:

- **Classification Layer:** For extractive summarization, a classification layer is added on top of the output of the [CLS] tokens. This layer computes a score for each sentence indicating its likelihood of being part of the summary. The scores are typically passed through a sigmoid function to map them to a probability space.
- **Training and Objective:** The model is trained using a binary cross-entropy loss, where each sentence is labeled as 1 if it is part of the reference summary and 0 otherwise. The model's task during training is to predict these labels as accurately as possible, based on the sentence embeddings derived from the [CLS] token.

❖ **Our Proposed Model: *BanglaSum***

For abstractive Summarization:

In this work, we propose a modified version of the Bangla-T5 model specifically tailored for abstractive summarization of video content. The proposed model is designed to handle a broader range of content, including both formal (e.g., news) and informal (e.g., academic lectures) language, making it suitable for domain-specific summarization.

1. Domain-Specific Summarization

While Bangla-T5 excels in abstractive summarization for structured and formal content, its performance is limited when processing informal, unstructured text. To address this limitation, we expanded the domain coverage of our model by incorporating both news and academic content. This approach allows the model to handle formal and informal language, making it adaptable across different content types, ensuring that the summarization process is equally effective in varied contexts.

2. Fine-tuning with Custom Dataset

We further enhanced the Bangla-T5 model by fine-tuning it with a custom dataset specifically created for this research. The dataset includes manually collected text from both news articles and academic videos, which helped the model better understand the nuances of different types of Bangla text. By training on this domain-specific dataset, our model is

capable of generating abstractive summaries that preserve key information while handling the variations in formality and content structure.

For Extractive Summarization:

Model Architecture

Bangla-BERTSum is built upon the original *BERTSum* architecture, which is designed for extractive summarization. However, in our approach, we replace the original BERT model with Bangla-BERT to tailor the system for Bangla language processing. The choice of Bangla-BERT ensures that the model leverages Bangla-specific semantic and syntactic nuances by using Bangla pretraining data, yielding more accurate and meaningful summaries for Bangla content.

Base Model Substitution

BERTSum uses BERT to encode input sequences and classify sentence importance for extractive summarization. In our adaptation, we replace the BERT base with *Bangla-BERT*, which has been pretrained on a large corpus of Bangla text. This adaptation ensures that the model understands Bangla-specific token patterns, morphology, and idiomatic expressions better than a generic BERT model. This is the most effective model for extractive text summarization

Dataset Preparation for Supervised Fine-Tuning

To fine-tune *Bangla-BERTSum*, we adopt a supervised learning approach. This involves the creation of a large dataset of extractive summary pairs, where each instance consists of full text i.e. block of video subtitles or content to be summarized and extractive summary that is a subset of sentences selected as the most important, serving as the ground truth for the extractive task. The model learns to predict which sentences should appear in the summary based on this labeled data, improving sentence selection accuracy.

Sequence of Abstractive and Extractive Summarization:

Since video content often consists of informal and unstructured language, our approach adopts a hybrid abstractive-extractive summarization technique to ensure both coherence and conciseness. We initiate the process with abstractive summarization using the Bangla-

T5 model, as abstractive methods are better suited for handling informal data. Abstractive models generate summaries by paraphrasing and restructuring content, which helps transform unorganized speech or subtitles into more coherent and structured text. However, videos are usually long, making it necessary to divide the input subtitles into smaller chunks. Each chunk is fed into Bangla-T5, which produces structured summaries for individual sections. These individual summaries are then concatenated to form a comprehensive abstractive summary of the entire video.

While the abstractive step provides a structured overview, it can result in verbose summaries, as the model attempts to capture all relevant information. To address this, we further process the concatenated abstractive summary using the *Bangla-BERTSum* model. This model, an adaptation of *BERTSum* with Bangla-BERT as its base, performs extractive summarization to refine the output, ensuring conciseness without sacrificing critical information. To maintain coherence and consistency across the final output, we divide the concatenated abstractive summary into smaller chunks before feeding them into *Bangla-BERTSum*. This ensures that overlapping content between the chunks helps avoid any loss of essential information, improving the precision of the extractive summary.

The use of this sequential approach—abstractive summarization followed by extractive summarization—is crucial for handling informal and complex video data. Starting with an abstractive model allows the unstructured and verbose content to be organized into more readable forms, while the subsequent extractive summarization ensures that the final summary is concise and retains the most relevant information. The flexibility of the extractive stage also ensures that all sentences meeting the threshold probability are included, without imposing a strict limit on summary length, thus preventing the loss of significant content. Both the Bangla-T5 and *Bangla-BERTSum* models are trained independently to optimize their individual tasks, and they are integrated sequentially to achieve a smooth and effective summarization process for Bangla video subtitles. The overall system architecture is given below in Figure 3.4.

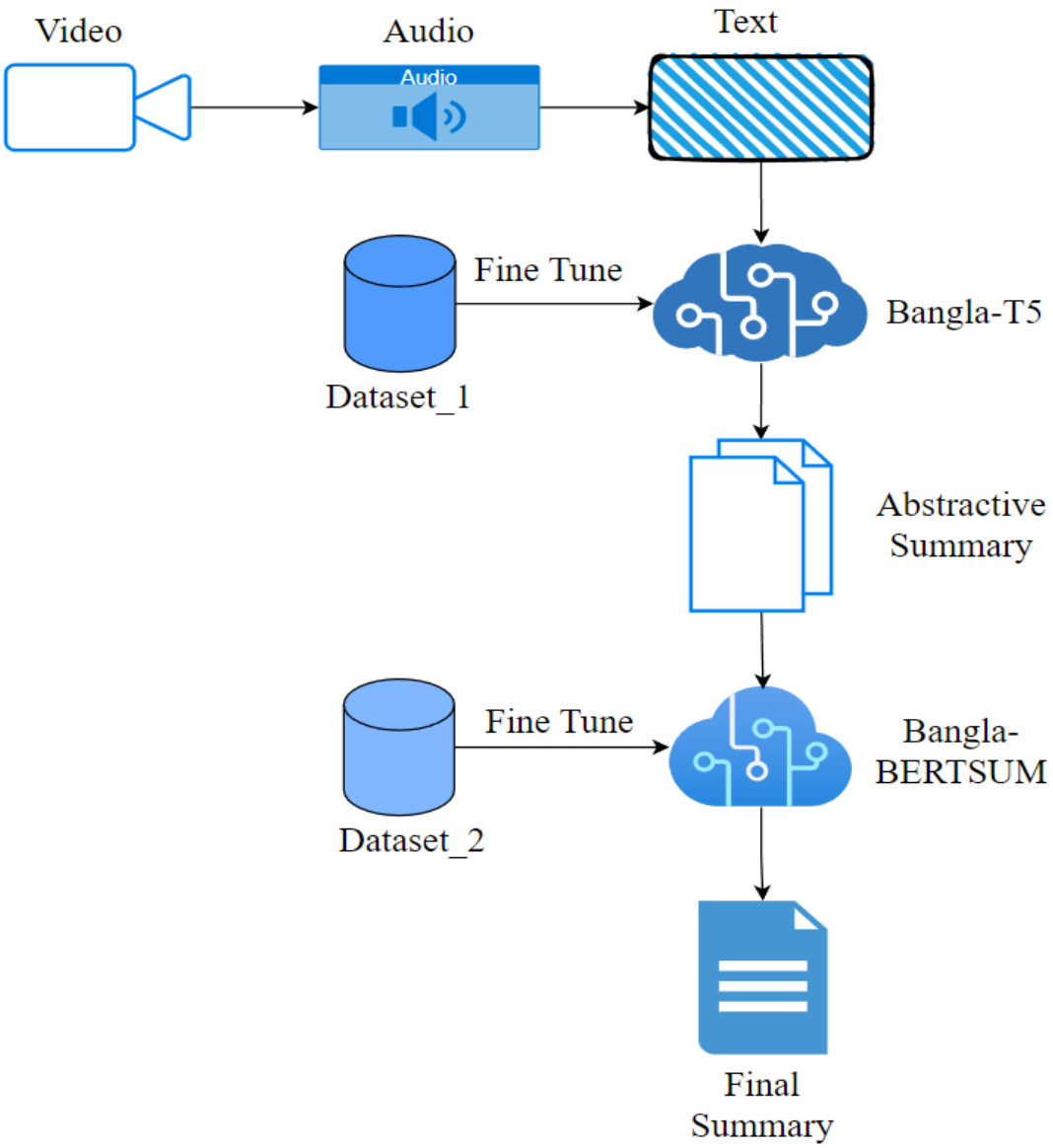

Figure 3.4: Overview of our proposed model (*BanglaSum*)

3.3 Conclusion

In conclusion, the methodology chapter outlines the systematic approach taken to achieve the research objectives. The chapter presents the data collection process, which was carefully designed to ensure the accuracy and relevance of the information gathered. The model selection was based on its suitability for the specific tasks and was fine-tuned for better performance. The overall methodology ensures that the research is replicable, data-driven, and aligned with the study's goals.

CHAPTER IV

Implementation and Results

4.1 Introduction

This section presents the results of the Bangla video summarization system, utilizing Bangla-T5 for abstractive summarization and *BERTSum* for extractive summarization. The system's performance is evaluated across formal (news) and informal (academic) content, focusing on the quality of the summaries generated. Key evaluation metrics such as ROUGE scores are used to assess coherence, fluency, and content retention. Comparisons between extractive and abstractive summarization methods highlight the strengths and limitations of each approach. The analysis also considers the impact of fine-tuning on the system's effectiveness across different text lengths and domains.

4.2 Experimental Setup

The experiments were conducted on a high-performance workstation equipped with an Intel i7 processor, 16 GB RAM, and an NVIDIA GPU to accelerate model training. We employed both the Bangla-T5 model for abstractive summarization and the *BERTSum* model for extractive summarization. The audio content from the videos was first extracted using the *MoviePy* library and then transcribed into text using Google Cloud's Speech-to-Text API.

The transcribed text was preprocessed using BNLTK, which handled tasks like stop word removal, punctuation restoration, and text normalization. The dataset, consisting of news articles and academic video transcripts, was divided into training, validation, and test sets to ensure fair and accurate evaluation.

For abstractive summarization, the Bangla-T5 model was fine-tuned with a learning rate of 2e-5, batch size of 16, and trained for 10 epochs. For extractive summarization, we used the *BERTSum* model, which selects the most relevant sentences from the text for a concise and structured summary.

Evaluation metrics such as ROUGE scores were used to assess the performance of both models. The results were validated through cross-validation to ensure generalizability and avoid overfitting.

4.3 Evaluation Metrics

Evaluating summaries generated through abstractive and extractive methods requires metrics that can measure the quality of the summaries from different perspectives, such as content retention, coherence, conciseness, and readability. Some of the most commonly used evaluation metrics in the field are Rogue and BLEU.

1. ROUGE (Recall-Oriented Understudy for Gisting Evaluation)

ROUGE is extensively utilized for evaluating both abstractive and extractive summarization models. This metric assesses the overlap of n-grams between the machine-generated summary and one or multiple human-written reference summaries. Its effectiveness lies in its ability to measure the preservation of content that is deemed important by the reference. The variants are given below:

- **ROUGE-N:** This variant focuses on the overlap of n-grams between the generated and reference summaries. ROUGE-1 and ROUGE-2 are frequently used to evaluate the overlap of unigrams and bigrams, respectively. This helps in understanding how much of the basic and slightly complex content has been captured by the summary.
- **ROUGE-L:** This involves skip-bigram co-occurrence statistics, effectively capturing more distant relationships within the text. It is useful for evaluating the preservation of more complex relational content.
- **ROUGE-S:** This involves skip-bigram co-occurrence statistics, effectively capturing more distant relationships within the text. It is useful for evaluating the preservation of more complex relational content.

2. BLEU (Bilingual Evaluation Understudy)

Originally designed for evaluating machine translation, BLEU has been effectively adapted for summarization tasks. It quantitatively measures how many n-grams of the generated summaries match the n-grams of any reference summary, adjusting for the length to avoid favoring overly terse summaries through a brevity penalty. BLEU is particularly advantageous in abstractive summarization settings where paraphrasing significantly alters

the phrasing from the source text. It provides a measure of how much essential information is retained despite the linguistic transformations inherent to abstractive methods.

These metrics collectively offer a comprehensive quantitative evaluation of the summarization models, providing clear evidence of their effectiveness and areas for improvement.

4.4 Dataset

Table 4.1 and Table 4.2 represent dataset details for abstractive and extractive summarization respectively.

Table 4.1: Statistics of the dataset for abstractive summarization

Total No. of Articles	19396
Total No. of Summaries	19396
Maximum No. of Words in Articles	401
Maximum No. of Words in Summaries	65
Minimum No. of Words in Articles	312
Minimum No. of Words in Summaries	49

Table 4.2: Statistics of the dataset for extractive summarization

Total No. of Articles	300
Total No. of Summaries	700
Maximum No. of Words in Articles	324
Maximum No. of Words in Summaries	43
Minimum No. of Words in Articles	260
Minimum No. of Words in Summaries	34

4.5 Implementation and Results

We have conducted 3 types of experiments and taken result from them. Firstly we have summarized the input video by using abstractive summarization method only (fine tuning of Bangla-T5). Then we have done the same for a sequential setup of abstractive and extractive summarization methods using fine tuning of Bangla-T5 and our new generated model *Bangla-BERTSum*. Also we have done the same in the reverse order. We have

observed both qualitative and quantitative results from these experiments. Table 4.3 shows the qualitative result that indicates that our proposed sequence generates a better summary. Table 4.4 shows the rogue scores where our proposed method achieves slightly better scores.

4.5.1 Qualitative Results

Table 4.3: Qualitative results

Input	পাইথন বাংলা ভিডিও সিরিজের এই ভিডিওটিতে আপনাদের সবাইকে স্বাগতম জানাচ্ছি। আমি আনিসুল ইসলাম। এই ভিডিওটিতে আমি আপনাদের পাইথন ল্যাঙ্গুয়েজের সাথে পরিচয় করিয়ে দেব। ভিডিওটি দেখার পর আপনার যে সকল টপিক সম্পর্কে জানতে পারবেন, সেগুলো এখানে উল্লেখ করা আছে। আপনারা দেখতে পাচ্ছেন প্রথমেই পাইথন কি, পাইথন আপনার কেন শিখবেন, পাইথন কি কি কাজে ব্যবহৃত হয়, পাইথনের বিভিন্ন ভার্সন সম্পর্কে জানতে পারবেন। এছাড়াও পাইথন শিখার জন্য কি কি সফটওয়্যার প্রয়োজন, সেগুলো সম্পর্কে আপনারা জানতে পারবেন।পাইথন হচ্ছে একটি হাই লেভেল প্রোগ্রামিং ল্যাঙ্গুয়েজ। হাই লেভেল প্রোগ্রামিং ল্যাঙ্গুয়েজ বলতে আমরা বুঝি এমন সকল ল্যাঙ্গুয়েজ, যেগুলো মানুষের ভাষার সাথে মিল আছে। লো লেভেল প্রোগ্রামিং ল্যাঙ্গুয়েজ হচ্ছে মেশিন লেভেল, অর্থাৎ বাইনারি ল্যাঙ্গুয়েজ। তো পাইথন যেহেতু হাই লেভেল প্রোগ্রামিং ল্যাঙ্গুয়েজ, অর্থাৎ মানুষের ভাষার সাথে মিল আছে, সেহেতু পাইথন শিখতে আপনার অনেক কম সময় লাগবে এবং অনেক সহজ হবে। পাইথন হচ্ছে একটি ইন্টারপ্রেটার ল্যাঙ্গুয়েজ; এটি কিন্তু কম্পাইল ল্যাঙ্গুয়েজ না। ইন্টারপ্রেটার এবং কম্পাইলার মধ্যে কি পার্থক্য সেটা জানার জন্য আমার ভিডিও অলরেডি করা আছে। আপনার চাইলে সেই ভিডিওটি দেখে নিতে পারেন। সংক্ষেপে যদি বলি, ইন্টারপ্রেটারের কাজ হচ্ছে আপনি যখন এটা হাই লেভেল প্রোগ্রামিং ল্যাঙ্গুয়েজের প্রোগ্রাম লিখবেন, সেটাকে লো লেভেল ল্যাঙ্গুয়েজে কনভার্ট করার কাজ হচ্ছে ইন্টারপ্রেটারের। ইন্টারপ্রেটার হচ্ছে একটি সফটওয়্যার। তো এই সফটওয়্যার আপনাকে সাহায্য করবে হাই লেভেল ল্যাঙ্গুয়েজের লো লেভেল ল্যাঙ্গুেজে কনভার্ট করতে। ইন্টারপ্রেটার একটা বৈশিষ্ট্য হচ্ছে এটি লাইন বাই লাইন ইন্টারপ্রেট করে, অর্থাৎ একটি লাইন ইন্টারপ্রেট করার পর পরবর্তী লাইন করবে। এই ছিল ইন্টারপ্রেটার সম্পর্কে বেসিক ধারণা। এটি হচ্ছে একটি অবজেক্ট ওরিয়েন্টেড ল্যাঙ্গুয়েজ। অবজেক্ট ওরিয়েন্টেড বলতে এখানে বোঝাচ্ছি রিয়েল লাইফের অবজেক্ট নিয়ে কাজ

	করতে পারে এমন একটি প্রোগ্রামিং ল্যাঙ্গুয়েজ হচ্ছে পাইথন। পাইথন মূলত যিনি তৈরি করেছিলেন, তার নাম এখানে দেয়া আছে: গুটেন রোশন। হয়তোবা আমার প্রোনউন্সিয়েশন ভুল হতে পারে, সেজন্য দুঃখিত। এটি রিলিজ হয়েছিল 1991 সালে। আপনার কেন শিখবেন, সে সম্পর্কে এবার জেনে নিন। বিগিনার ফ্রেন্ডের ল্যাঙ্গুয়েজ এটি। খুব সহজেই আপনি শিখতে পারবেন। আপনি যদি অন্য কোন ল্যাঙ্গুয়েজ যদি নাও শিখেন, কোন প্রবলেম নেই; খুব সহজেই ল্যাঙ্গুয়েজ শিখতে পারবেন, কারণ এটার সিনট্যাক্স, পাইথন প্রোগ্রামিং লেখার পদ্ধতি টা খুবই সহজ, যেটা ইংলিশ ভাষার সাথে মিল আছে। বাইতুল যদি আপনি কম সংখ্যক লাইন লিখেন, সে ক্ষেত্রে অনেক বেশি কাজ করতে পারবেন। এখানে, আপনার অন্যান্য ল্যাঙ্গুয়েজ যদি শিখে থাকেন, তাহলে দেখবেন যে অন্যান্য ল্যাঙ্গুয়েজের তুলনায় এখানে কম সংখ্যক লাইন ইউজ করার প্রয়োজন হয়। দা মোস্ট ওর প্রোগ্রামিং ল্যাঙ্গুয়েজ ইন ইউনিভার্সিটি। পৃথিবীর বিভিন্ন ইউনিভার্সিটিতে সবচেয়ে বেশি যে ল্যাঙ্গুয়েজটা শেখানো হয়, সেটি হচ্ছে পাইথন ল্যাঙ্গুয়েজ। পাইথন বিভিন্ন প্ল্যাটফর্মে কাজ করতে পারে, অর্থাৎ আপনি উইন্ডোজ অপারেটিং সিস্টেম ব্যবহার করেন কিংবা ম্যাক, লিনাক্স, যে কোনো অপারেটিং সিস্টেম ব্যবহার করেন, সমস্যা নেই; পাইথন সেগুলোতে কাজ করবে। ইউজড বাই মেনি পপুলার অর্গানাইজেশন: গুগল, নাসা, সিআইএ এরকম অনেক পপুলার যে অর্গানাইজেশনগুলো আছে, সেখানে পাইথন ব্যবহার করা হয়। পাইথন খুবই ডিমান্ডেবল একটা ল্যাঙ্গুয়েজ, ডেফিনেটলি। এবং এখানে দেখতে পাচ্ছেন পাইথন কি কি কাজে ব্যবহৃত হয়। এক নজর দেখে নিতে পারেন: ওয়েব ডেভেলপমেন্ট, যেটি ব্যবহার করা হয়
Expected Summary	এই ভিডিওতে আনিসুল ইসলাম পাইথন প্রোগ্রামিং ভাষার সঙ্গে পরিচয় করিয়ে দেন এবং এর বিভিন্ন দিক নিয়ে আলোচনা করেন। পাইথন একটি উচ্চ স্তরের প্রোগ্রামিং ভাষা, যা শিখতে খুব সহজ এবং কার্যকর। হাই লেভেল প্রোগ্রামিং ভাষা হিসেবে এটি মানুষের ভাষার সাথে বেশ মিল আছে, ফলে এটি শিখতে তুলনামূলকভাবে কম সময় লাগে। পাইথন একটি ইন্টারপ্রেটার ভিত্তিক ভাষা, যা কোড লাইন বাই লাইন ইন্টারপ্রেট করে। এর ফলে, কোড লেখার সময় সহজেই ডিবাগিং এবং দ্রুত পরীক্ষা করা যায়। ভিডিওতে ইন্টারপ্রেটার এবং কম্পাইলারের মধ্যে পার্থক্য নিয়েও আলোচনা করা হয়েছে। পাইথন একটি অবজেক্ট ওরিয়েন্টেড ভাষা, যা বাস্তব জীবনের অবজেক্টগুলোর সাথে কাজ করতে পারে।

	এটি ১৯৯১ সালে গুইডো ভ্যান রোসাম দ্বারা তৈরি হয়েছিল। পাইথনের সবচেয়ে বড় সুবিধা হলো, এটি বিগিনারদের জন্য অত্যন্ত উপযোগী, এবং অন্যান্য প্রোগ্রামিং ভাষার তুলনায় কম কোড লিখেই অধিক কার্য সম্পাদন করা যায়। যেকোনো প্রোগ্রামিং ব্যাকগ্রাউন্ড ছাড়াই সহজেই এটি শেখা যায়। পাইথনের ব্যবহারিক ক্ষেত্র সম্পর্কে আলোচনা করা হয়, যেখানে উল্লেখ করা হয় যে এটি ওয়েব ডেভেলপমেন্ট, ডেস্কটপ অ্যাপ্লিকেশন, মোবাইল অ্যাপ্লিকেশন, এবং ডাটা সায়েন্সের মতো বিভিন্ন ক্ষেত্রে ব্যবহৃত হয়। বিশেষ করে, ডাটা সায়েন্স এবং মেশিন লার্নিংয়ে এর ব্যবহার দিন দিন বৃদ্ধি পাচ্ছে। পাইথনের আরেকটি বড় সুবিধা হলো, এটি বিভিন্ন প্ল্যাটফর্মে যেমন উইন্ডোজ, ম্যাক এবং লিনাক্সে কাজ করতে সক্ষম। এছাড়া গুগল, নাসা, সিআইএ এর মতো বড় প্রতিষ্ঠানগুলোতে পাইথন ব্যবহার করা হয়, যা এটিকে একটি অত্যন্ত চাহিদাসম্পন্ন এবং শক্তিশালী প্রোগ্রামিং ভাষা হিসেবে প্রতিষ্ঠা করেছে।
Generated Summary (Abstractive only)	পাইথন বাংলা ভিডিও সিরিজের এই ভিডিওটিতে আমি আপনাদেরকে পাইথ্যালার্মের সাথে পরিচয় করিয়ে দেব। এটি বাংলা ভাষায় তৈরিকৃত। ভিডিওটির নাম। PLEASE JOIN ME. THIS VIDEO আপনাদের সবাইকে ধন্যবাদ। ভিডিওটা দেখতে ভুলবেন না যেন। এই ভিডিওতে রয়েছে: পাইথুত। পাইথন হচ্ছে একটি হাই লেভেল প্রোগ্রামিং ল্যাংগুয়েজ। অর্থাৎ বাইনারি প্রোগ্রামিং ল্যাঙ্গুয়েজ। তো পাইথ্নেন হচ্ছে মানুষের ভাষার সাথে মিল আছে এমন সব প্রোগ্রামিং ভাষা। তাই তাই না? YES?? WELL? ই-থন হচ্ছে একটি ইন্টারপ্রেটার ল্যাঙ্গুয়েজ; এটি কিন্তু কম্পাইলার নয়। I HAVE AN ALREADY-DOED VIDEO ITHINK IS AN INTERPRETER LANGUAGE; IT IS A VIDEO ইথনান ইথন ইথান IDO, IDAN. ইন্টারপ্রেটারের কাজ হচ্ছে আপনাকে হাই লেভেল প্রোগ্রামিং ল্যাংগুয়েজ থেকে লো লেভেল ল্যাংগুেজে প্রোগ্রাম করা। IN SHORT, THE JOB OF THE SOFTWARE IS TO DO THIS. SO, SO THE WORK OF SOFTWARE. তো এই বৈশিষ্ট্য হচ্ছে, সফটওয়্যার আপনাকে সাহায্য করবে করতে। তো তোেে (INTERPRET) সুতরাং সুতরাং এটি একটি সফটওয়্যার। I'M SORRY FOR MY PRONUNCIATION ERRORS, THAT MAY BE MY FAULT, FOR THAT TO BE SORRY. এটি রিলিজ হয়েছিল ১৯৯১ সালে। এটি ছিল একটি রিলিজ এটি 1991 সালে মুক্তি পেয়েছিল। খুব সহজে আপনি ল্যাঙ্গুয়েজটি শিখতে পারবেন। এখানে, আপনার অন্য কোন ল্যাঙ্গুয়েজ না জানলেও চলবে। খুব সহজেই আপনি এটা শিখতে পারেন। এটা খুবই সহজ। সহজ, খুব সহজ

	এখানে কোন প্রবলেম নেই নেই এখানেটা জেনে নিন। THIS IS THE LANGUAGE YOU WILL LEARN. এখানে আপনি পাবেন ফ্রেন্ডের ল্যাংগুয়েজ। খুবই সরল।। এটি খুবই সিম্পল এখানে এটা সহজ এখানে পাইথন হচ্ছে সবচেয়ে বেশি ব্যবহৃত প্রোগ্রামিং ল্যাঙ্গুয়েজ ইন ইউনিভার্সিটি। পাইথউইন ল্যাংগুয়েজ হচ্ছে ইউনিভার্সিটিগুলোর শেখানো সবচেয়ে জনপ্রিয় প্রোগ্রামিং ভাষা। এটি একটি উন্মুক্ত বিশ্ববিদ্যালয়। PYTHON IS THE MOST USED LANGUAGE IN THE WORLD.োোোো:ো; ইউনিভার্সিটি: পাইওনিয়ার আপনি যদি ম্যাক, লিনাক্স অথবা ম্যাক অপারেটিং সিস্টেম ব্যবহার করেন, তাহলে আপনি যে কোনো অপারেটিং সিস্টেমে কাজ করতে পারবেন। THE MOST WIDELY USED ENGINEERING LANGUAGE IS PIONEER IN UNIVERSITIES. এখানে দেখতে পাচ্ছেন, গুগল, নাসা, সিআইএ এরকম অনেক কাজে ব্যবহৃত হয়। এখানে দেখুন, অনেক অ্যাপ্লিকেশন ডাটা সায়েন্স ব্যবহার করে। এবং এখানে আমরা পাইথন ব্যবহার করি। হ্যাঁ, এটা খুবই দরকারী একটা সাবজেক্ট। SO, THIS IS VERY IMPORTANT. AND HERE.
Generated Summary (Extractive after Abstractive)	ভিডিওটিতে আপনারা দেখতে পাবেন যে, পাইথন কিভাবে শিখতে হয়। এছাড়াও আপনারা আরও অনেক কিছু শিখতে পারবেন। আশা করি আপনারা সবাই ভিডিওটি উপভোগ করবেন। পাইথ্যালাপি হচ্ছে একটি বাংলা ভিডিও। ভিডিওটি এখানে ক্লিক করুন।।তো এই হচ্ছে ইন্টারপ্রেটারের কাজ। আপনি যদি কম সংখ্যক লাইন লেখেন, সে ক্ষেত্রে অনেক বেশি কাজ করতে পারবেন। তো এটাই হবে আপনার শেখার কারণ। বাইতুল এখানে আমি যা শিখবো, তা হচ্ছে। আমি।।বাইতুল আপনি যা শিখবেন, বাইত।বাইত।।তাহলে আমি শিখব কিভাবে? ধন্যবাদ বং এখানে আপনি দেখতে পাচ্ছেন, গুগল, নাসা, সিআইএ এরকম আরো অনেক অর্গানাইজেশন আছে। এবং দেখুন, অনেক অ্যাপ্লিকেশন, ডাটা সায়েন্স পাইথন ব্যবহার করা হয়। এখানে দেখুন এখানে, এখানে, ডেটা সায়েন্স। HERE, HERE., এখানে। হ্যাঁ, হ্যাঁ। অনেক কাজে লাগে এখানে এবং অনেক ক্ষেত্রে, অ্যাপ্লিকেশন এখানে
Generated Summary (Abstractive	পাইথন বাংলা ভিডিও সিরিজের এই ভিডিওটিতে আমি আপনাদেরকে পাইথ্যালার্মের সাথে পরিচয় করিয়ে দেব। পাইথন হচ্ছে একটি হাই লেভেল প্রোগ্রামিং ল্যাংগুয়েজ। তো পাইথ্নেন হচ্ছে মানুষের ভাষার সাথে মিল আছে এমন সব প্রোগ্রামিং ভাষা। ই-থন হচ্ছে একটি ইন্টারপ্রেটার ল্যাঙ্গুয়েজ; এটি কিন্তু কম্পাইলার নয়। ইন্টারপ্রেটারের কাজ হচ্ছে আপনাকে হাই লেভেল প্রোগ্রামিং

after Extractive)	ল্যাংগুয়েজ থেকে লো লেভেল ল্যাংগুেজে প্রোগ্রাম করা। IN SHORT, THE JOB OF THE SOFTWARE IS TO DO THIS. তো এই বৈশিষ্ট্য হচ্ছে, সফটওয়্যার আপনাকে সাহায্য করবে করতে। তো তোেে (INTERPRET) সুতরাং সুতরাং এটি একটি সফটওয়্যার। I'M SORRY FOR MY PRONUNCIATION ERRORS, THAT MAY BE MY FAULT, FOR THAT TO BE SORRY. এটি ছিল একটি রিলিজ এটি 1991 সালে মুক্তি পেয়েছিল। খুব সহজে আপনি ল্যাঙ্গুয়েজটি শিখতে পারবেন। এখানে, আপনার অন্য কোন ল্যাঙ্গুয়েজ না জানলেও চলবে। সহজ, খুব সহজ এখানে কোন প্রবলেম নেই নেই এখানেটা জেনে নিন। এটি খুবই সিম্পল এখানে এটা সহজ এখানে পাইথন হচ্ছে সবচেয়ে বেশি ব্যবহৃত প্রোগ্রামিং ল্যাঙ্গুয়েজ ইন ইউনিভার্সিটি। পাইথউইন ল্যাংগুয়েজ হচ্ছে ইউনিভার্সিটিগুলোর শেখানো সবচেয়ে জনপ্রিয় প্রোগ্রামিং ভাষা। PYTHON IS THE MOST USED LANGUAGE IN THE WORLD. েোেোো:ো; ইউনিভার্সিটি: পাইওনিয়ার আপনি যদি ম্যাক, লিনাক্স অথবা ম্যাক অপারেটিং সিস্টেম ব্যবহার করেন, তাহলে আপনি যে কোনো অপারেটিং সিস্টেমে কাজ করতে পারবেন। THE MOST WIDELY USED ENGINEERING LANGUAGE IS PIONEER IN UNIVERSITIES. এখানে দেখতে পাচ্ছেন, গুগল, নাসা, সিআইএ এরকম অনেক কাজে ব্যবহৃত হয়।

We can see that the summary is more precise when we are using the sequential method of extractive after abstractive summarization.

4.5.2 Quantitative Results

We have calculated ROUGE score for all the experiments.

Table 4.4: Qualitative results

Method	ROUGE-1	ROUGE-2	ROUGE-L
Abstractive only	2.04%	0.1%	2.04%
Extractive after Abstractive	**4%**	**1.1%**	**4%**
Abstractive after Extractive	3.01%	0.2%	3.01%

4.6 Objectives Achieved

We are hoping that the proposed system will effectively meet the objectives outlined in the introduction:

- We will develop a Bangla video summarization system by integrating both abstractive (Bangla-T5) and extractive (Bangla-*BERTSum*) models, producing concise and meaningful summaries of video content.
- By summarizing video subtitles, the system significantly reduces the time users need to watch lengthy videos, allowing quick access to essential information.
- We efficiently handled spoken language data by extracting text from video audio and processing it for summarization, ensuring accuracy in transcription and summarization.
- The two-stage methodology combining abstractive chunking and extractive refinement preserved the context and key messages, ensuring coherent and relevant summaries.
- The system addressed the challenges of low-resource Bangla language processing by proposing a novel framework that automates the summarization of Bangla video content.
- Our work improves accessibility to Bangla video content by allowing users to efficiently identify relevant videos.
- Lastly, we contributed to Bangla NLP technology advancement by fine-tuning and implementing state-of-the-art models like Bangla-T5 and Bangla-*BERTSum* for summarization.

This is how we are projecting to meet these objectives.

4.7 Morality or Ethical Issues

Ensuring ethical compliance in academic research, particularly in projects involving copyrighted material and personal data, is paramount. This section outlines the ethical standards and practices adopted in this thesis to address such concerns effectively.

- **Copyright Compliance**

The video subtitles used in this research were carefully selected to ensure that all copyrighted materials were handled appropriately. All video content processed for subtitle

summarization was either in the public domain or available under open-source licenses that permit academic use. For any content where copyright was applicable, permissions were obtained from the copyright holders prior to inclusion in the dataset, ensuring compliance with copyright law.

- **Proper Citation and Use of Resources**

Proper attribution and citation practices were strictly followed throughout the thesis to ensure academic integrity. All tools, datasets, and previous research leveraged during the project were appropriately credited, with detailed references provided in the bibliography. Citation standards adhered to the guidelines set by the academic community, ensuring that all resources and prior works are properly acknowledged.

- **Data Privacy and Anonymity**

In cases where the research involved proprietary data or personal information, measures were taken to protect privacy. Any data that could potentially identify individuals were anonymized before analysis. Data handling protocols were established in accordance with institutional review board (IRB) guidelines to ensure that all data were processed ethically.

- **Avoidance of Plagiarism**

To maintain the integrity of the academic work, sophisticated plagiarism detection software was utilized to ensure that all written content is original or properly cited. Drafts of the thesis were periodically reviewed to detect and rectify any inadvertent plagiarism or improper paraphrasing prior to final submission. This thesis upholds the highest standards of academic ethics and integrity, ensuring that all research activities and outputs adhere to legal and ethical norms. Such rigorous ethical considerations reinforce the reliability of the research findings and contribute to a foundation of trust and scholarly respect in the academic community.

4.8 Socio-Economic Impact and Sustainability

The research on Bangla video subtitle summarization aims to enhance accessibility and understanding of video content for the Bangla-speaking population, particularly in educational and informational contexts. This initiative can have significant socio-economic impacts:

- **Societal, Health, Safety, Legal, and Cultural Issues**

The implementation of video subtitle summarization can bridge communication gaps for the hearing-impaired and non-native speakers, promoting inclusivity and equal access to information. By making educational content more comprehensible, the project encourages lifelong learning and enhances literacy rates within the community. Furthermore, it supports mental health by providing resources that individuals can engage with at their own pace, reducing the stress associated with language barriers. Legally, ensuring that content is accessible can help organizations comply with regulations regarding equal access to information. Culturally, the initiative aids in the preservation and dissemination of Bangla culture by making local content more accessible to diverse audiences. Quantitatively, the project is expected to significantly increase the viewership and engagement rates of Bangla video content.

- **Environmental Impact and Sustainability**

The project promotes sustainability by utilizing digital platforms for content dissemination, thereby reducing the reliance on printed materials. By making video content more accessible, the initiative can decrease the carbon footprint associated with traditional educational methods. The digital nature of the project also allows for continuous updates and improvements to the summarization algorithms, ensuring that the content remains relevant and aligned with environmental standards. Moreover, increased accessibility of educational resources can foster a more informed populace that actively engages in discussions surrounding environmental issues and sustainability practices. This could lead to a greater public awareness of environmental challenges and encourage sustainable behaviors within the community as well as save valuable time and cost.

4.9 Financial Analyses and Budget

For the successful implementation of this thesis, a financial budget was carefully planned and outlined. The key components of the budget cover the necessary software tools, hardware resources, and other incidental costs required to complete the project. This initial budget planning ensures that all necessary financial resources are allocated to complete the thesis successfully. Here, the initial budget planning is given in Table 4.1.

Table 4.1: Initial Budget Planning

Item	**Description**	**Cost (in BDT)**	**Use**
Computer/Workstation	Required for development/testing	80,000	Used for model training, testing, and processing
GPU	Optional for faster training	50,000	Accelerates model training processes
Python Libraries (e.g., NLTK, BNLTK, docx)	Open-source libraries for text preprocessing	0	Essential for text normalization, tokenization, and processing
Kaggle (Compute Credits)	For model training using large datasets	0 (Free Tier)	Free resources for running models
Internet Charges	For cloud-based model training	2,000 (monthly)	Required for cloud usage, model training, and data access
Data Storage (Cloud)	Google Drive/Other services	1,000 (monthly)	For storing data, model outputs, and other large files
Total Estimated Cost		**133,000+**	

4.10 Conclusion

The results demonstrate that the Bangla-T5 model for abstractive summarization significantly improves the readability and coherence of the generated summaries, particularly for informal content like academic videos, where the text may be less structured. On the other hand, *Bangla-BERTSum* for extractive summarization performs well in preserving key factual details but may not be as effective in maintaining fluency and context across longer texts. The hybrid approach of combining both techniques offers a balanced solution, ensuring that the generated summaries are both informative and concise. However,

we have also seen that the sequence of abstractive after extractive summarization performs better than the sequence of extractive after abstractive summarization.

CHAPTER V

Conclusion

5.1 Summary

This research introduces a novel approach to summarizing video content by extracting text from audio. We used *Pydub* Audio Extraction to convert video to audio, followed by Google's Speech Recognition API to extract text. The method sequentially combines abstractive summarization (Bangla-T5) for capturing the context and extractive summarization (*Bangla-BERTSum*) to ensure sentence consistency and coherence, creating concise and meaningful summaries. Datasets were manually collected for informal content (e.g., academic videos) and existing datasets were used for formal content (e.g., news). Both models were fine-tuned with our datasets, with necessary preprocessing to improve summary quality. This dual-stage approach enhances the system's ability to generate contextually accurate and coherent Bangla video summaries.

5.2 Limitations

This research on Bangla video summarization encountered several challenges, primarily due to the scarcity of large-scale annotated datasets specific to Bangla. The limited availability of pre-trained Bangla language models also constrained the effectiveness of certain NLP tasks. Additionally, speech-to-text accuracy for Bangla remains an issue, as existing APIs struggle with variations in pronunciation and accent. The computational demands of abstractive and extractive summarization models further limited experimentation on larger datasets.

REFERENCES

[1] R. R. Chowdhury, M. T. Nayeem, T. T. Mim, M. S. R. Chowdhury, and T. Jannat, "Unsupervised abstractive summarization of Bengali text documents," *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume* (P. Merlo, J. Tiedemann, and R. Tsarfaty, eds.), (Online), pp. 2612–2619, Association for Computational Linguistics, Apr. 2021.

[2] J. Madhuri and R. Ganesh Kumar, "Extractive text summarization using sentence ranking," *2019 International Conference on Data Science and Communication (IconDSC)*, pp. 1–3, 2019.

[3] V. Rennard, G. Shang, J. Hunter, and M. Vazirgiannis, "Abstractive meeting summarization: A survey," *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 861–884, 2023.

[4] M. Talukder, S. Abujar, A. K. M. Masum, and S. Hossain, "Bengali abstractive text summarization using sequence to sequence RNNs," July 2019.

[5] P. Bhattacharjee, M. S. Islam, M.-E.-J. Mukta, and A. Mallick, "Bengali abstractive news summarization (BANS): A neural attention approach," Dec. 2020.

[6] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," pp. 7871–7880, Jan. 2020.

[7] A. Bhattacharjee, T. Hasan, W. Ahmad, and R. Shahriyar, "BanglaNLG and BanglaT5: Benchmarks and resources for evaluating low-resource natural language generation in Bangla," pp. 726–735, Jan. 2023.

[8] S. M. Hayat, A. Das, and M. Hoque, "Abstractive Bengali text summarization using transformer-based learning," pp. 1–6, Dec. 2023.

[9] A. R. Thakare and P. Voditel, "Extractive text summarization using LSTM-based encoder-decoder classification," *ECS Transactions*, vol. 107, p. 11665, Apr. 2022.

[10] D. Miller, "Leveraging BERT for extractive text summarization on lectures," *Georgia Institute of Technology*, 2024. Available online.

[11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (J. Burstein, C. Doran, and T. Solorio, eds.), (Minneapolis, Minnesota), pp. 4171–4186, Association for Computational Linguistics, June 2019.

[12] Y. Liu and M. Lapata, "Text summarization with pretrained encoders," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 163–171, Association for Computational Linguistics, 2019.

[13]]Y. Liu, "Bertsum: Extractive summarization using bert," in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 3006–3015, Association for Computational Linguistics, 2020.

[14] H. Zhang, X. Liu, and J. Zhang, "Extractive summarization via chatgpt for faithful summary generation," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (University of California, Davis, CA, USA), 2023.

[15] S. Sotudeh, A. Cohan, and N. Goharian, "On generating extended summaries of long documents," in *Proceedings of the 2021 Conference on Empirical Methods 36 in Natural Language Processing*, (Washington D.C., USA), Association for Computational Linguistics, 2021.

[16] T. A. Foysal, M. A. Mahadi, M. M. H. Nahid, and A. Tasnim, "Bangla-extrasum: Comparative analysis of different methods in automated extractive bengali text summarization," in *2021 5th International Conference on Electrical Engineering and Information Communication Technology (ICEEICT)*, pp. 1–6, 2021.

[17] A. Bhattacharjee, T. Hasan, W. U. Ahmad, K. Samin, M. S. Islam, A. Iqbal, M. S. Rahman, and R. Shahriyar, "Banglabert: Language model pretraining and benchmarks for low-resource language understanding evaluation in bangla," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, (Bangladesh University of Engineering and Technology (BUET)), 2021.

[18] M. Kowsher, A. A. Sami, N. J. Prottasha, M. S. Arefin, P. K. Dhar, and T. Koshiba, "Bangla-bert: Transformer-based efficient model for transfer learning and language understanding," *IEEE Access*, vol. 10, pp. 91855–91870, 2022.

[19] S. Wu and M. Dredze, “Are all languages created equal in multilingual bert?,” in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (Johns Hopkins University), Association for Computational Linguistics, 2020.

[20] K. M. Hasib, M. A. Rahman, M. I. Masum, F. D. Boer, S. Azam, and A. Karim, “Bengali news abstractive summarization: T5 transformer and hybrid approach,” in *2023 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pp. 539–545, 2023.