BanglaTalk: Towards Real-Time Speech Assistance for Bengali Regional Dialects

Jakir Hasan
Shahjalal University of Science and Technology, BD
jakirhasan718@gmail.com

Shubhashis Roy Dipta
University of Maryland, Baltimore County, USA
sroydip1@umbc.edu

Abstract

Real-time speech assistants are becoming increasingly popular for ensuring improved accessibility to information. Bengali, being a low-resource language with a high regional dialectal diversity, has seen limited progress in developing such systems. Existing systems are not optimized for real-time use and focus only on standard Bengali. In this work, we present **BanglaTalk**, the first real-time speech assistance system for Bengali regional dialects. BanglaTalk follows the client-server architecture and uses the Real-time Transport Protocol (RTP) to ensure low-latency communication. To address dialectal variation, we introduce a dialect-aware ASR system, **BRDialect**, developed by fine-tuning the IndicWav2Vec model in ten Bengali regional dialects. It outperforms the baseline ASR models by 12.41-33.98% on the RegSpeech12 dataset. Furthermore, BanglaTalk can operate at a low bandwidth of 24 kbps while maintaining an average end-to-end delay of 4.9 seconds. Low bandwidth usage and minimal end-to-end delay make the system both cost-effective and interactive for real-time use cases, enabling inclusive and accessible speech technology for the diverse community of Bengali speakers.[1]

1 Introduction

Conversational speech assistants (Dutsinma et al., 2022) have transformed human-computer interaction, making information more accessible. Widely adopted tools such as Alexa, Siri, and Cortana demonstrate the profound impact of real-time speech assistants on human lives (Hoy, 2018). However, while significant progress has been made for high-resource languages such as English, Mandarin, and French, such tools are still underdeveloped for the low-resource Bengali language. Bengali is a morphologically rich Indo-Aryan language (Islam et al., 2025), spoken by approximately 260 million people worldwide. It exhibits significant regional dialectal diversity, with variations in phonology, vocabulary, and syntax (Hasan et al., 2024b). This linguistic diversity poses a major challenge in building robust speech assistants.

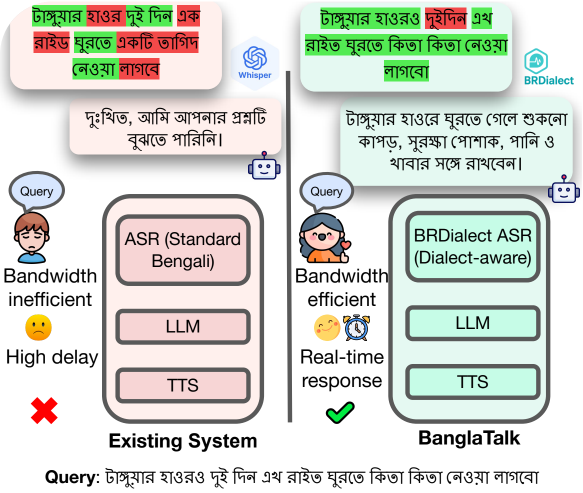

টাঙ্গুয়ার হাওর দুই দিন এক রাইড ঘুরতে একটি তাগিদ নেওয়া লাগবে
Whisper
দুঃখিত, আমি আপনার প্রশ্নটি বুঝতে পারিনি।
Query
Bandwidth inefficient
High delay
ASR (Standard Bengali)
LLM
TTS
Existing System
টাঙ্গুয়ার হাওরও দুইদিন এথ রাইত ঘুরতে কিতা কিতা নেওয়া লাগবো
BRDialect
টাঙ্গুয়ার হাওরে ঘুরতে গেলে শুকনো কাপড়, সুরক্ষা পোশাক, পানি ও খাবার সঙ্গে রাখবেন।
Query
Bandwidth efficient
Real-time response
BRDialect ASR (Dialect-aware)
LLM
TTS
BanglaTalk
Query: টাঙ্গুয়ার হাওরও দুই দিন এথ রাইত ঘুরতে কিতা কিতা নেওয়া লাগবো

Figure 1: Existing Bengali speech assistants (left) fail to understand queries in regional dialects due to reliance on standard Bengali ASR (incorrect transcriptions are shown in red). **BanglaTalk (right) successfully handles regional dialect queries through its dialect-aware ASR (BRDialect).** It is bandwidth efficient and operates in real-time due to the incorporation of the Real-Time Transport Protocol.

Automatic Speech Recognition (ASR) is a key component of speech assistant systems. Existing ASR systems are developed primarily for standard Bengali (Saha et al., 2021; Rakib et al., 2023b), and their performance is significantly degraded in regional dialects. As a result, existing speech assistant systems that integrate such ASR cannot support regional dialectal communication (Hasan et al., 2021; Arnab et al., 2023). Moreover, real-time deployment requires not only dialectal robustness, but also minimal end-to-end delay and efficient bandwidth usage. Previous works lack dialect-aware

[1] https://github.com/Jak57/BanglaTalk

ASR, systematic analysis of delay minimization techniques, bandwidth efficiency, and real-time communication.

In this work, we introduce BanglaTalk, the first real-time conversational speech assistant for Bengali regional dialects. BanglaTalk adopts a client-server architecture and incorporates the Real-time Transport Protocol (RTP) (Schulzrinne et al., 2003) to achieve low-latency communication. Robust audio encoding enables operation at 24 kbps (kilobits per second). As illustrated in Fig. 1, while existing speech assistants fail in interpreting regional dialectal queries, BanglaTalk transcribes them accurately through the dialect-aware ASR system. It responds to queries effectively and interactively in real-time using low bandwidth.

The BanglaTalk client integrates lightweight audio processing modules, including noise cancellation, dynamic range compression, and audio encoding. On the server side, a dialect-aware ASR system, a voice activity detector (VAD), a natural-sounding Text-to-Speech (TTS) system, and audio encoding modules form a complete pipeline for real-time speech assistance. Central to this system is BRDialect, a dialect-aware ASR model fine-tuned on ten Bengali regional dialects. BRDialect outperforms the baseline Whisper (Tugstugi, 2023) and IndicWav2Vec (Javed et al., 2022) models, achieving a word error rate of 74.1% and character error rate of 40.6% on the RegSpeech12 (Hassan et al., 2025) dataset.

Additionally, the integrated VITS (Kim et al., 2021) TTS model produces natural-sounding speech with a high mean opinion score (MOS) of 4.49, enhancing the user experience. With an average end-to-end delay of 4.9 seconds, BanglaTalk enables interactive real-time communication between the user and the speech assistant. This system will significantly impact the lives of Bengali speakers due to its dialect-aware ASR, low bandwidth usage, and real-time performance.

In summary, our main contributions are:

- We introduce BanglaTalk, the first real-time, bandwidth-efficient Bengali speech assistant designed to support regional dialects through a client-server architecture.

- We develop BRDialect, a dialect-aware ASR system that substantially outperforms existing ASR models on the RegSpeech12 dataset spanning twelve regions of Bangladesh.

- We provide a comprehensive analysis of audio processing latency, bandwidth usage, end-to-end delay, and generated speech quality, demonstrating the robustness of BanglaTalk for real-time, dialect-aware communication.

2 Methodology

BanglaTalk follows a client-server architecture, with lightweight audio processing on the client and computationally intensive tasks on a centralized server. The overall pipeline is illustrated in Fig. 2.

2.1 BanglaTalk Client

The client is responsible for capturing, processing, and transmitting audio to the server. As shown in Fig. 2 (left), its main modules include audio capture, dynamic range compression, noise suppression, encoding, and transmission.

Audio Capture The client captures audio in 20-ms (milliseconds) frames at a sample rate of 16 kHz (kilohertz). Each frame contains 320 samples in 16-bit PCM little-endian format (Dobson, 2000). Although the Opus codec (Valin et al., 2016) supports multiple sample rates (e.g., 8-48 kHz), we fix the sample rate at 16 kHz to align with the ASR and VAD modules. Opus allows frame durations of 2.5-100 ms. A frame duration of 20 ms (50 RTP packets per second) offers a balance between packet size and loss rate in real-time communication.

Dynamic Range Compression (DRC) Speech captured from the microphone often includes soft and excessively loud signals. Compressing the dynamic range helps maintain a consistent audio level, enhancing performance (Giannoulis et al., 2012). We develop the dynamic range compression algorithm described in Alg. 1 and apply compression only to the loud audio segments. Whenever the decibel level of a normalized audio sample exceeds -10 dBFS (decibels relative to full scale), we apply a compression ratio of 2:1. Samples outside this threshold remain unchanged.

RNNoise Cancellation Noise poses a major issue in audio communication. Background and foreground noise can be picked up by the microphone and transmitted to the network, degrading the overall performance of the system. To mitigate this, we perform noise suppression with RNNoise (Valin, 2018). It is a lightweight neural network-based denoiser capable of real-time operation.

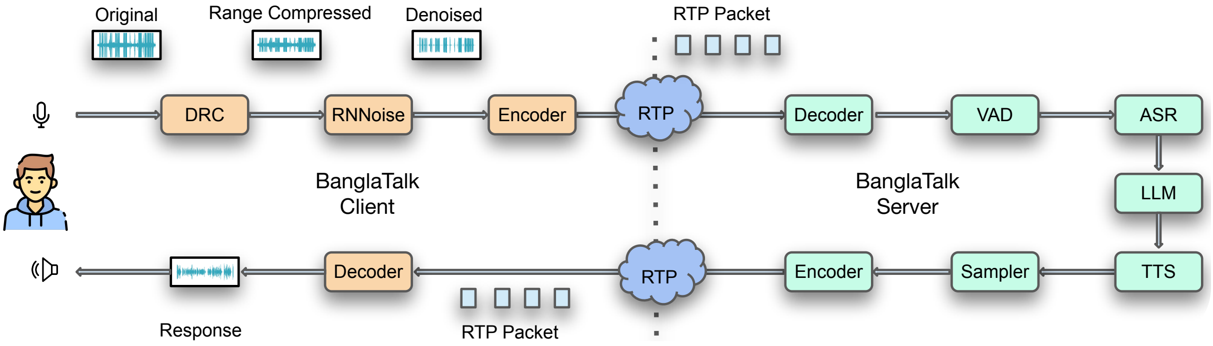


Figure 2: Client (left) and server-side (right) processing pipelines of the BanglaTalk System.

RNNoise is trained to remove noise from an audio frame of duration 10 ms at a 48 kHz sample rate. Since every captured audio frame duration is 20 ms at a 16 kHz sample rate, we apply audio segmentation, upsampling, and downsampling during noise suppression. Alg. 2 shows the pseudocode for upsampling. Linear interpolation (Xu and Xu, 2022) is used in upsampling from 16 to 48 kHz due to minimal computational overhead. After denoising, the audio is downsampled to 16 kHz using Alg. 3. Specifically, we skip intermediate sample values – from every three consecutive audio samples, the first one is retained and the remaining two are discarded. This simple technique is computationally efficient for downsampling.

Encoding with Opus Codec Opus is a high-quality audio codec for interactive speech and music transmission over the Internet (Valin et al., 2016). It is widely used in VoIP (Voice over IP) applications (Sundvall, 2014) due to the low latency processing and error concealment. Each audio frame contains a total of 320 samples (640 bytes). Without compression, audio frames are expensive to transmit over the Internet due to high bandwidth consumption. To mitigate this, each audio frame is encoded using the Opus codec with a low bitrate of 24 kbps. This ensures low bandwidth usage, which is a critical factor for greater accessibility. The incoming audio frames from the server are decoded with the same codec.

Packetization and Transmission Each encoded audio frame is encapsulated in an RTP packet following the Real-time Transport Protocol (Group et al., 1996). The first 12 bytes of each packet contain the header, and the remaining bytes contain the Opus-encoded payload. App. B describes the structure of the RTP packet. These packets are sent at 20 ms intervals to the server's public IP and the RTP packet receiver port (Postel, 1980).

2.2 BanglaTalk Server

The server is responsible for receiving the audio stream from the client and generating an appropriate audio response. The overall server-side processing pipeline is illustrated in Fig. 2 (right). To function effectively, the server employs several interconnected modules. First, the incoming audio frames are received and decoded using the Opus decoder. Next, voice activity detection is performed to identify speech segments. When a complete user query is detected, the corresponding speech segment is transcribed into text. This text is then processed by a large language model (LLM), which generates a suitable response. The response is subsequently converted into speech using a TTS system. Finally, the speech is resampled and the resulting audio frames are encoded with the Opus codec. Following the RTP protocol, the created RTP packets are transmitted to the client. The detailed workflow of these modules is described below.

RTP Packet Parser The server receives audio data from the client in the form of RTP packets. Each packet is parsed to extract header information, encoded data length, and encoded audio data in byte format.

Decoding with Opus Codec The extracted encoded audio is decoded using the Opus codec. Based on the encoded data length and audio bytes, the decoder reconstructs audio frames of 20 ms duration, corresponding to 320 samples at a 16 kHz sample rate.

Voice Activity Detection (VAD) Voice activity detection is a crucial component of real-time communication, as it identifies speech segments while discarding non-speech portions. This prevents unnecessary downstream processing, thus reducing la-

Algorithm 1 DRC compresses the amplitude of samples exceeding a -10 dBFS threshold by applying a 2:1 compression ratio. It leaves the quieter samples unchanged.

```
Require: Frame $x \in \mathbb{Z}^N$, threshold $\tau = -10$ dBFS, ratio $r = 2$
Ensure: Compressed frame $y \in \mathbb{Z}^N$
$y \leftarrow x$
for $i \leftarrow 1$ to $N$ do
    $s \leftarrow y_i$
    if $s = 0$ then
        continue
    end if
    $d \leftarrow 20 \cdot \log_{10}(|s|/32768)$
    if $d \leq \tau$ then
        continue
    end if
    $d' \leftarrow \tau + (d - \tau)/r$
    $\sigma \leftarrow \mathrm{sign}(s)$
    $s' \leftarrow \left\lfloor 10^{d'/20} \cdot \sigma \cdot 32767 \right\rfloor$
    $y_i \leftarrow s'$
end for
return $y$
```

tency and improving overall efficiency. We use the Silero VAD (Team, 2024) in the streaming mode, which is specifically designed for real-time applications. It works with a frame duration of 32 ms (512 samples) at a sample rate of 16 kHz and determines whether each audio frame marks the beginning, end of a speech segment, or none. Only the detected speech segment corresponding to the user query is forwarded to the ASR system.

Automatic Speech Recognition (ASR) We train a Wav2Vec2-based model (Baevski et al., 2020) using speech data from ten regional dialects from the Ben10 dataset (Humayun et al., 2024). For data processing, we follow Hasan et al. (2024a) and fine-tune the pre-trained IndicWav2Vec (Javed et al., 2022) for Bengali on that processed dataset. To evaluate the performance of our trained ASR model, we use the RegSpeech12 (Hassan et al., 2025) test set (Ben10 test set is not publicly available). A detailed description and analysis of these datasets are provided in App. C.1 and App. C.2.

End of Query Detection Accurate detection of the end of a user query is essential for real-time speech assistance systems (Liang et al., 2023). We have defined the end of query as a silence segment lasting at least 1.2 seconds. For silence detection, Silero VAD is utilized. Once an end-of-query is detected, the speech segment is passed to the ASR system to generate the transcription. The resulting query is then forwarded to the LLM, which produces the system's response.

Generating Response Using LLM Large Language Models (LLMs) are crucial for generating responses to user queries (Dam et al., 2024). To maintain coherent communication, responses are generated for valid queries, while invalid queries are discarded. We employ `GPT-4.1-nano` as the chat model, with the prompt template presented in App. D. To minimize latency, we use streaming mode, which delivers responses incrementally rather than waiting for a full response. The streamed text is segmented based on Bengali punctuation and forwarded to the TTS system.

Text-to-Speech (TTS) Several TTS models are available for Bengali (Raju et al., 2019). We experiment with MMS-TTS-Ben (Pratap et al., 2024) and two variants of VITS-Bengali (male and female voices) (Hossen, 2023). These models are selected because of their minimal processing delay. MMS-TTS-Ben produces speech at a 16 kHz sample rate, while VITS-Bengali outputs at 22.05 kHz. For system compatibility, the VITS-generated speech is resampled to 16 kHz.

Network Transmission The processed audio is segmented into 20 ms frames and encoded with the Opus codec at a bitrate of 24 kbps. Each RTP packet is constructed with a 12-byte header, followed by encoded audio data. The RTP packets are transmitted over the Internet to the client's public IP and port at 20 ms intervals, ensuring synchronized real-time playback.

3 Result & Discussion

3.1 Implementation Details

Since the system is intended for deployment across a diverse population in Bangladesh, we prioritize computational efficiency on the client side to ensure accessibility across devices with varying hardware capabilities. In contrast, the server must be sufficiently powerful to handle audio processing, response generation, and real-time communication with minimal latency. Accordingly, our experiments were conducted with an Intel Core i7 CPU (without GPU) as the client and a server equipped

Model	WER ↓	CER ↓
Whisper-medium-Bengali	0.846	0.562
IndicWav2Vec-Bengali	0.897	0.615
BRDialect	**0.741**	**0.406**

Table 1: Performance of ASR systems on the test set of the RegSpeech12 dataset, covering twelve Bengali regional dialects.

with an NVIDIA GeForce RTX 4090 GPU for efficient processing.

3.2 Evaluating BRDialect

To evaluate the performance of our dialect-aware ASR system, **BRDialect**, we use the test set from the RegSpeech12 (Hassan et al., 2025) dataset. It includes dialects from twelve regions of Bangladesh – Rangpur, Sylhet, Chittagong, Noakhali, Narail, Kishoreganj, Barishal, Habiganj, Comilla, Tangail, Sandwip, and Narsingdi, totaling 2132 audio files.

For evaluation, we report the Word Error Rate (WER) and Character Error Rate (CER), following the formulas described in App. E.1. To refine ASR predictions, we apply beam search decoding with a 5-gram KenLM language model (Heafield, 2011). We also investigated the impact of the preprocessing and postprocessing steps, including noise cancellation, normalization, and punctuation removal, as these factors significantly affect the overall performance of the ASR.

As shown in Table 1, **BRDialect** outperforms baseline models in both WER and CER. Specifically, it achieves the lowest WER of 0.741 and CER of 0.406 when decoded with a 5-gram KenLM model, combined with Unicode normalization and punctuation removal, but without noise cancellation.

BRDialect consistently outperforms `Whisper-medium-Bengali` (Tugstugi, 2023) and `IndicWav2Vec-Bengali` (Javed et al., 2022), achieving a 12.41–17.39% relative improvement in WER and a 27.77–33.98% improvement in CER. The comparatively poor performance of the baseline models suggests that dialectal variation is not adequately addressed during their training, limiting their suitability for regional speech-to-text tasks. BRDialect highlights the importance of dialect-aware fine-tuning for building a robust ASR system.

Processing	WER ↓	CER ↓
Noise Cancellation	0.876	0.497
No Noise Cancellation	0.865	0.452
5-gram KenLM Decoding	0.827	0.442
Unicode Normalization	0.796	0.420
Punctuation Removal	**0.741**	**0.406**

Table 2: Impact of different types of processing on the performance of the BRDialect ASR system. Processing pipelines are combined from top to bottom consecutively for the group without noise cancellation.

Performance across Regions We further evaluate BRDialect across individual regions. As shown in Fig. 3, the ASR system performs well across most regions, with WER below 70% in seven out of the twelve regions. The lowest WER, 0.438, is achieved for the Comilla region. Although the Comilla dialect is not included in the training data of BRDialect, its low WER demonstrates the model's strong generalization capability to unseen dialects.

3.2.1 Ablation Study

We analyze the effect of different preprocessing and postprocessing techniques on ASR performance. Table 2 summarizes the improvements observed with BRDialect.

Impact of Noise Cancellation We experiment with denoising the input audio before transcribing. The experimental results show that denoising with RNNoise slightly increases WER by 1.27%. Aggressive denoising can remove speech cues necessary for the accurate transcription of regional dialects. For the remaining processing steps, we keep the original audio without denoising and combine processing pipelines.

Impact of 5-gram KenLM Decoding Integrating a 5-gram KenLM model during decoding improves WER from 0.865 to 0.827, a 4.39% reduction. This confirms the value of training a KenLM language model with a Bengali regional text corpus for robust ASR performance (Rakib et al., 2023b).

Impact of Unicode Normalization Normalizing text further reduces WER to 0.796. This step is crucial since many Bengali characters can be represented in multiple ways, causing transcription inconsistencies. Normalizing text with the `BnUnicodeNormalizer` (Ansary et al., 2023) en-

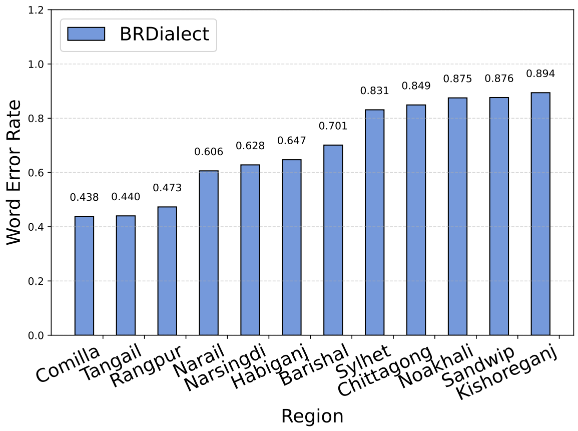


Figure 3: Regionwise word error rate distribution of the test set of the RegSpeech12 dataset. Transcriptions are generated using the BRDialect ASR system.

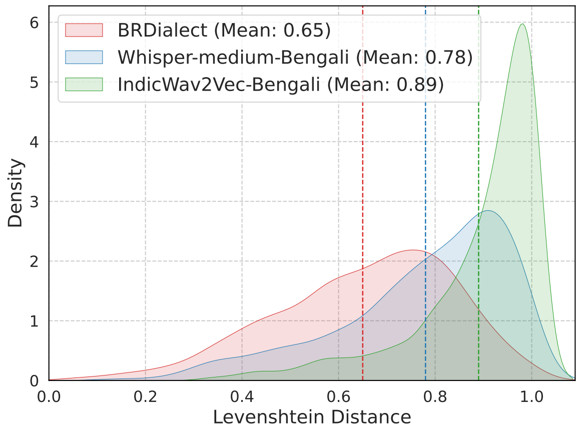


Figure 4: Distribution of Levenshtein distance for the best processing settings - without noise cancellation, Bangla unicode normalization, and punctuation removal by three ASR systems on the RegSpeech12 dataset.

sures uniform representation, improving ASR performance.

Impact of Punctuation Removal Since baseline ASR models do not generate punctuation, we remove punctuation from the RegSpeech12 transcriptions to ensure a fair comparison. Combined with previous steps, this yields the lowest WER of 0.741 and CER of 0.406. A detailed analysis of the processing configurations and their impact on the BRDialect ASR is provided in App. E.2.

Impact of High WER A Word Error Rate (WER) of 74.1% is relatively high for a general-purpose ASR system. However, dialectal speech recognition is inherently more challenging than the standard ASR task. The poor performance of the baseline models (`Whisper-medium-Bengali` and `IndicWav2Vec-Bengali`) validates this difficulty. In this context, BRDialect achieves a relative improvement of 12.41-33.98% over the baseline models, representing a substantial achievement.

Within the BanglaTalk system, the integrated large language model (`GPT-4.1-nano`) effectively compensates for minor transcription errors. Leveraging its robust contextual understanding, the LLM can infer user intent even from noisy or imperfect ASR outputs, as illustrated in Fig. 10. Most observed errors involve minor substitutions that do not significantly affect the intended meaning.

Furthermore, explicitly prompting the LLM to interpret dialectal queries (Fig. 7) enhances the system's ability to generate appropriate responses. The combination of a dialect-aware ASR model (BRDialect) and the powerful LLM (`GPT-4.1-nano`) ensures that the BanglaTalk system remains usable and effective despite relatively imperfect transcriptions.

3.2.2 Levenshtein Distance Analysis

Fig. 4 illustrates the distribution of normalized Levenshtein distance between the ground-truth transcriptions and the outputs of the evaluated ASR systems under the best processing configuration – No noise cancellation, Unicode normalization, and punctuation removal. For BRDialect, transcription quality is further refined during decoding using our trained 5-gram KenLM language model. A lower Levenshtein distance indicates higher transcription accuracy.

Among the models, BRDialect achieves the lowest mean distance of 0.65, demonstrating strong alignment with the reference transcriptions. `Whisper-medium-Bengali` performs moderately with a mean distance of 0.78, while `IndicWav2Vec-Bengali` consistently underperforms. The distribution of `IndicWav2Vec-Bengali` peaks sharply at 1, with the highest mean distance 0.89, highlighting substantial deviation from the ground truth.

In contrast, BRDialect exhibits a broader distribution, reflecting variability in performance – some utterances are transcribed with high accuracy, while others are less. It achieves a measurable improvement of 16.67-26.97% compared to the baseline models, highlighting its effectiveness in handling dialectal diversity.

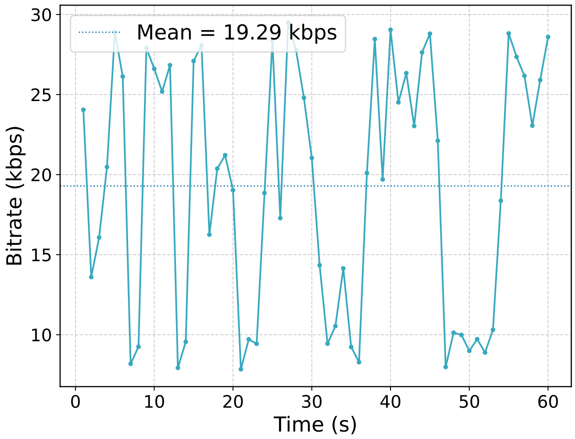


Figure 5: Uploading bitrate on the client side for a duration of one minute.

3.3 Evaluating BanglaTalk

Low Latency Audio Processing On the client side, each captured audio frame undergoes three sequential processing steps – dynamic range compression, noise cancellation, and encoding with the Opus codec. The average processing time for an audio frame of duration 20 ms, is reported in Table 3.

Among these, the encoding with the Opus codec is the most efficient, requiring only 0.56 ms per frame. This is consistent with the codec's design, which targets real-time, low-latency audio applications (Valin et al., 2016). Similarly, the DRC module introduces minimal computational overhead, averaging 1.31 ms per frame. In contrast, the noise cancellation with RNNoise introduces the highest processing time of 6.51 ms.

RNNoise applies a lightweight neural network for speech enhancement, making it more resource-intensive than traditional signal processing methods. Additionally, RNNoise operates exclusively at a 48 kHz sample rate, while the BanglaTalk system relies on 16 kHz to ensure compatibility with ASR and VAD modules (see §2.1). As a result, each frame must first be upsampled from 16 kHz to 48 kHz prior to noise cancellation and subsequently downsampled back to 16 kHz. These resampling operations introduce an additional delay of 1.40 ms for upsampling and 0.34 ms for downsampling. Compared to other ML-based noise cancellation systems (Cha et al., 2023), RNNoise provides an excellent balance of speech quality and efficiency, making it highly suitable for real-time speech applications (Valin, 2018).

Despite this, the total processing time (8.38 ms) remains well below the 20 ms frame duration, ensuring that all processing completes before the arrival of the next frame. This guarantees uninterrupted, real-time streaming without additional latency.

Module	**Process Time (ms)**
DRC	1.31
RNNoise	6.51
Encoding	**0.56**
Total	8.38

Table 3: Average processing times of a single audio frame of duration 20 ms.

Adaptive Bitrate Control In real-time assistive speech technologies, both end-to-end latency and bandwidth efficiency are crucial factors (Kaur et al., 2019). Bandwidth efficiency is particularly important for underserved populations, such as those in rural areas of Bangladesh, where high-speed or expensive Internet connections are often unavailable.

To address this challenge, our system prioritizes bandwidth efficiency without compromising performance. Audio streams are encoded at a target bitrate of 24 kbps, significantly reducing bandwidth usage in the upload and download streams during client-server communication. Fig. 5 shows the upload bitrate of client-side for a one-minute audio stream.

We employ the variable bitrate (VBR) mode of the Opus codec, which dynamically adjusts the bitrate based on the characteristics of the audio signal – allocating more bitrate to speech segments and conserving bitrate during silence. In the example shown in Fig. 5, the average bitrate over one minute is only 19.29 kbps. This adaptive bitrate usage makes the system more accessible to users in economically disadvantaged regions (Osuagwu et al., 2013), who might otherwise be unable to benefit from assistive speech technologies.

Although applying RNNoise noise cancellation slightly increases the ASR system's Word Error Rate (WER) by 1.27% (Table 2), it is retained in the client application due to its substantial benefit in reducing bandwidth usage. By removing background and foreground noise, the audio signal becomes more compressible. Without noise cancellation, the average upload bitrate of the audio (as shown in Fig. 5) rises from 19.29 kbps to 23.6 kbps – an 18.26% increase that is inefficient for client-side transmission. Given that the large language model (GPT-4.1-nano) demonstrates strong robustness in understanding minor transcription errors and generating accurate responses (Fig. 10), the advantages of incorporating RNNoise into the client application outweigh its modest negative impact on ASR performance.

End-to-End Latency For real-time assistive speech systems, minimizing the delay between the end of a user's query and the beginning of the system's audio response is essential to ensure natural interactivity. The BanglaTalk system is designed with this principle in mind, introducing minimal overhead in both client and server-side processing.

We conduct rigorous testing and time analysis to measure the BanglaTalk system's end-to-end delay. As detailed in App. E.3, the BRDialect ASR module of the BanglaTalk system introduces only a small delay, making it well-suited for real-time applications. The TTS systems evaluated in our study also exhibit low processing delay. App. F presents a detailed evaluation of TTS systems. Among them, the VITS-Bengali (male variant) is the best-performing model, achieving a high Mean Opinion Score (MOS) of 4.49 on the subset of the BanSpeech (Samin et al., 2024) dataset.

To quantify the overall end-to-end delay of the BanglaTalk system, we simulate ten user queries in a conversation setting and measure the time elapsed from the end of the user queries to the start of the system's response. As shown in App. G, the system achieves an average end-to-end delay of 4.9 seconds. This latency is acceptable for real-time assistive applications, ensuring smooth interactivity and an enhanced user experience.

Preliminary User Study To evaluate the real-world usability of the BanglaTalk system, we conduct a limited user study involving four native speakers of the Sylhet and Mymensingh dialects. The Sylhet dialect is selected due to its relative high WER (0.831), while the Mymensingh dialect is included to assess the generalizability of BRDialect to unseen dialects, as it is not part of the training set. Two native speakers from each region interact with BanglaTalk on general information and everyday task queries. Participants rate their interaction experience on a 1-5 scale, where 1 indicates a poor experience and 5 indicates an excellent experience. Table 9 summarizes the results of five queries per user. Overall, BanglaTalk achieves a mean rating of 3.62, indicating a generally positive user experience.

As illustrated in Fig. 10, users from both regions interact successfully with the system despite minor transcription errors. Notably, users from the Mymensigh region, whose dialect is not included in BRDialect's training data, report a mean satisfaction rating of 3.73, further demonstrating the model's strong generalization capability to unseen dialects. Participants also note that, despite their strong accent in regional dialects, BanglaTalk responds accurately and naturally, providing an interactive and effective speech-based experience.

4 Related Work

4.1 Bengali Speech Assistants

Several speech assistant systems have been developed for the Bengali language in recent years. The ALAPI system (Hasan et al., 2021) introduces an open-domain Bengali conversational agent that processes recorded audio queries from users and generates response audio using AI techniques alongside a custom-built database. Shohojogi (Arnab et al., 2023), designed for the banking sector, provides voice-based customer support in Bengali. It integrates Wav2Vec2-based ASR (Baevski et al., 2020), query summarization, Google Text-to-Speech (gTTS), and a doc2vec model to retrieve relevant information for responses. Adrisya Sahayak (Sultan et al., 2021) presents a desktop-based virtual speech assistant for visually impaired Bengali speakers, supporting computer operations, peripheral devices, and home appliance control.

In the healthcare domain, a voice-enabled Artificial Conversational Entity has been developed to automate service interactions in Bengali (Pranto et al., 2021). This system leverages a domain-specific database and similarity-matching strategies to generate responses to user queries. Extending beyond monolingual assistants, Disha (Ullah et al., 2024) is a humanoid virtual assistant capable of interacting in both Bengali and English. It can address financial queries and perform real-time transactions. Beyond general-purpose applications, Bengali voice assistants have also been deployed in specialized domains such as providing information on metro rail services in Bangladesh (Rahman et al., 2024), assisting farmers with agricultural queries (Divakar et al., 2021), and offering accessible feminine healthcare support for marginalized women (Puja et al., 2024).

4.2 English Speech Assistants

Voice assistants such as Amazon Alexa, Apple Siri, Microsoft Cortana, and Google Assistant have become integral to modern human-computer interaction (López et al., 2017). These systems, powered by artificial intelligence, are increasingly influencing daily life and societal practices (Subhash et al., 2020). Their development has been driven by advances in both signal processing and machine learning, which form the technological foundation of voice-based interaction (Haeb-Umbach et al., 2019).

Beyond their technical design, researchers have examined the broader impact of these systems. For example, Flavián et al. (2023) demonstrated that, compared to text-based recommendations, voice-based recommendations delivered by assistants have a stronger influence on consumer decision-making. User studies also indicate that voice assistants are most frequently employed for activities such as music, information search, and smart home (IoT) control (Ammari et al., 2019). Furthermore, their role has extended into education and healthcare contexts, where they support learning environments and assist older adults in managing everyday activities (Terzopoulos and Satratzemi, 2020; Oewel et al., 2023).

4.3 Automatic Speech Recognition in Bengali

Research on Automatic Speech Recognition (ASR) in Bengali has been addressed through both system development and survey-driven studies (Mridha et al., 2022; Tasnia et al., 2023; Sultana et al., 2021). The availability of large-scale datasets, such as Bengali Common Voice (Alam et al., 2022), OOD-Speech (Rakib et al., 2023a), and RegSpeech12 (Hassan et al., 2025), has created significant opportunities for advancing Bengali ASR systems. Continuous Bengali ASR has been benchmarked on the SHRUTI corpus using DNN-HMM and GMM-HMM-based models (Al Amin et al., 2019), achieving relatively low error rates. Leveraging CNN-RNN architectures, Saha et al. (2021) developed a gender- and speaker-independent ASR system. In Bangla-Wave (Rakib et al., 2023b), the integration of an n-gram language model is shown to yield notable performance improvements. Furthermore, comparative analysis demonstrates that the Wav2Vec-BERT model outperforms Whisper on the Bengali Common Voice dataset (Ridoy et al., 2025).

5 Conclusion & Future Work

In this work, we present BanglaTalk, the first real-time, end-to-end conversational speech assistant designed specifically for Bengali regional dialects. The system integrates both client and server applications, with low-latency audio signal processing implemented on both ends. To ensure efficient network transmission, speech data is transmitted at a low bandwidth of 24 kbps, making the system accessible to a broad range of users. Our developed BRDialect ASR system, integrated into the BanglaTalk pipeline, effectively transcribes Bengali regional dialects, achieving an overall word error rate of 74.1% and a character error rate of 40.6% on the RegSpeech12 dataset, which spans 12 regions of Bangladesh. For speech synthesis, the VITS-Bengali TTS model (male) incorporated into the system attains a mean opinion score (MOS) of 4.49 on a subset of the BanSpeech dataset, enhancing the naturalness and human-like quality of the generated voice. Furthermore, the system maintains a low end-to-end delay of 4.9 seconds, ensuring a highly interactive user experience. These performance metrics demonstrate the effectiveness of the system for real-time communication.

Limitations

The BRDialect ASR system is trained on regional speech data covering ten regions of Bangladesh. Regions not included in the training data may experience less accurate transcriptions when using the BanglaTalk system. Incorporating speech data from the remaining regions of Bangladesh is expected to significantly improve the performance of the BRDialect ASR system. Although BanglaTalk reduces end-to-end delay and is highly interactive, several limitations remain:

- User interruption: In the current system, the assistant continues speaking until the utterance is completed. Adding the capability to handle user interruptions would make conversations more natural and interactive.
- Speaker verification: The system does not verify the speaker. If another person speaks during the user's conversation, their speech is transcribed, which negatively affects performance. Incorporating speaker verification would mitigate this issue.
- Concurrent conversations: At present, only one conversation can be executed at a time.

Supporting multiple concurrent conversations would increase system availability and usability.

- User study coverage: Feedback has so far been collected from a limited set of regional speakers. A broader user study covering all regions of Bangladesh would provide deeper insights into system performance, user acceptance, and overall impact.

References

Md Alif Al Amin, Md Towhidul Islam, Shafkat Kibria, and Mohammad Shahidur Rahman. 2019. Continuous bengali speech recognition based on deep neural network. In *2019 international conference on electrical, computer and communication engineering (ECCE)*, pages 1–6. IEEE.

Samiul Alam, Asif Sushmit, Zaowad Abdullah, Shahrin Nakkhatra, MD Ansary, Syed Mobassir Hossen, Sazia Morshed Mehnaz, Tahsin Reasat, and Ahmed Imtiaz Humayun. 2022. Bengali common voice speech dataset for automatic speech recognition. *arXiv preprint arXiv:2206.14053*.

Tawfiq Ammari, Jofish Kaye, Janice Y Tsai, and Frank Bentley. 2019. Music, search, and iot: How people (really) use voice assistants. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 26(3):1–28.

Nazmuddoha Ansary, Quazi Adibur Rahman Adib, Tahsin Reasat, Asif Shahriyar Sushmit, Ahmed Imtiaz Humayun, Sazia Mehnaz, Kanij Fatema, Mohammad Mamun Or Rashid, and Farig Sadeque. 2023. Unicode normalization and grapheme parsing of indic languages. *arXiv preprint arXiv:2306.01743*.

Kabir Abdur Rahman Arnab, Ashiqual Hossain, Istihad Nabi, and Muhammad Iqbal Hossain. 2023. *Shohojogi: An automated voice chat system in the bangla language for the banking system*. Ph.D. thesis, Ph. D. Dissertation. https://doi. org/10. 13140/RG. 2.2. 34757.22240/1.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460.

Young-Jin Cha, Alireza Mostafavi, and Sukhpreet S Benipal. 2023. Dnoisenet: Deep learning-based feedback active noise control in various noisy environments. *Engineering Applications of Artificial Intelligence*, 121:105971.

Sumit Kumar Dam, Choong Seon Hong, Yu Qiao, and Chaoning Zhang. 2024. A complete survey on llm-based ai chatbots. *arXiv preprint arXiv:2406.16937*.

MS Divakar, Vimal Kumar, Martina Jaincy DE, RA Kalpana, Sanjai Kumar RM, and 1 others. 2021. Farmer's assistant using ai voice bot. In *2021 3rd International Conference on Signal Processing and Communication (ICPSC)*, pages 527–531. IEEE.

Richard W Dobson. 2000. Developments in audio file formats. In *ICMC*.

Faruk Lawal Ibrahim Dutsinma, Debajyoti Pal, Suree Funilkul, and Jonathan H Chan. 2022. A systematic review of voice assistant usability: An iso 9241–11 approach. *SN computer science*, 3(4):267.

Carlos Flavián, Khaoula Akdim, and Luis V Casaló. 2023. Effects of voice assistant recommendations on consumer behavior. *Psychology & Marketing*, 40(2):328–346.

Dimitrios Giannoulis, Michael Massberg, and Joshua D Reiss. 2012. Digital dynamic range compressor design—a tutorial and analysis. *Journal of the Audio Engineering Society*, 60(6):399–408.

Audio-Video Transport Working Group, H Schulzrinne, S Casner, R Frederick, V Jacobson, and 1 others. 1996. Rfc1889: Rtp: A transport protocol for real-time applications.

Reinhold Haeb-Umbach, Shinji Watanabe, Tomohiro Nakatani, Michiel Bacchiani, Bjorn Hoffmeister, Michael L Seltzer, Heiga Zen, and Mehrez Souden. 2019. Speech processing for digital home assistants: Combining signal processing with deep-learning techniques. *IEEE Signal processing magazine*, 36(6):111–124.

Jakir Hasan, Md. Ataullha Saim, and Radeen Mostafa. 2024a. Improving bengali asr for regional dialects with indicwav2vec: A competition approach. Preprint, ResearchGate.

Md Mehedi Hasan, Auronno Roy, and Md Tariq Hasan. 2021. Alapi: An automated voice chat system in bangla language. In *2021 International Conference on Electronics, Communications and Information Technology (ICECIT)*, pages 1–4. IEEE.

Md Nahid Hasan, Raiyan Azim, and Sadia Sharmin. 2024b. Credibility analysis of robot speech based on bangla language dialect. In *2024 IEEE International Conference on Computing, Applications and Systems (COMPAS)*, pages 1–6. IEEE.

Md. Rezuwan Hassan, Azmol Hossain, Kanij Fatema, Rubayet Sabbir Faruque, Tanmoy Shome, Ruwad Naswan, Trina Chakraborty, Tawsif Tashwar Dipto, Md Foriduzzaman Zihad, Nazmuddoha Ansary, Asif Sushmit, Ahmed Imtiaz Humayun, Tahsin Reasat, Md Mehedi Hasan Shawon, Md. Golam Rabiul Alam, and Farig Sadeque. 2025. Regspeech12: A regional corpus of bengali spontaneous speech across dialects. Kaggle Dataset. Accessed: 2025-09-19.

Kenneth Heafield. 2011. Kenlm: Faster and smaller language model queries. In *Proceedings of the sixth workshop on statistical machine translation*, pages 187–197.

Mobassir Hossen. 2023. Comprehensive bangla tts. Kaggle dataset. Accessed: 2025-09-19.

Matthew B Hoy. 2018. Alexa, siri, cortana, and more: an introduction to voice assistants. *Medical reference services quarterly*, 37(1):81–88.

Ahmed Imtiaz Humayun, farigys, Mohaymen Ul Anam, Rubayet Sabbir Faruque, S. M. Jishanul Islam, Sushmit, and Tahsin. 2024. Asr for regional dialects. Kaggle competition. Accessed: 2025-09-19.

Md Fuadul Islam, Jakir Hasan, Md Ashikul Islam, Prato Dewan, and M Shahidur Rahman. 2025. Banglalem: a transformer-based bangla lemmatizer with an enhanced dataset. *Systems and Soft Computing*, page 200244.

Tahir Javed, Sumanth Doddapaneni, Abhigyan Raman, Kaushal Santosh Bhogale, Gowtham Ramesh, Anoop Kunchukuttan, Pratyush Kumar, and Mitesh M Khapra. 2022. Towards building asr systems for the next billion users. In *Proceedings of the aaai conference on artificial intelligence*, volume 36, pages 10813–10821.

Ravneet Kaur, Ravtej Singh Sandhu, Ayush Gera, Tarlochan Kaur, and Purva Gera. 2019. Intelligent voice bots for digital banking. In *Smart Systems and IoT: Innovations in Computing: Proceeding of SSIC 2019*, pages 401–408. Springer.

Jaehyeon Kim, Jungil Kong, and Juhee Son. 2021. Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In *International Conference on Machine Learning*, pages 5530–5540. PMLR.

Dawei Liang, Hang Su, Tarun Singh, Jay Mahadeokar, Shanil Puri, Jiedan Zhu, Edison Thomaz, and Mike Seltzer. 2023. Dynamic speech endpoint detection with regression targets. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Gustavo López, Luis Quesada, and Luis A Guerrero. 2017. Alexa vs. siri vs. cortana vs. google assistant: a comparison of speech-based natural user interfaces. In *International conference on applied human factors and ergonomics*, pages 241–250. Springer.

Muhammad Firoz Mridha, Abu Quwsar Ohi, Md Abdul Hamid, and Muhammad Mostafa Monowar. 2022. A study on the challenges and opportunities of speech recognition for bengali language. *Artificial Intelligence Review*, 55(4):3431–3455.

Bruna Oewel, Tawfiq Ammari, and Robin N Brewer. 2023. Voice assistant use in long-term care. In *Proceedings of the 5th International Conference on Conversational User Interfaces*, pages 1–10.

OE Osuagwu, S Okide, D Edebatu, and E Udoka. 2013. Low and expensive bandwidth remains key bottleneck for nigeria's internet diffusion: A proposal for a solution model. *West African Journal of Industrial and Academic Research*, 7(1):14–30.

Jon Postel. 1980. User datagram protocol. Technical report.

Shehan Irteza Pranto, Rahad Arman Nabid, Ahnaf Mozib Samin, Nabeel Mohammed, Farhana Sarker, Mohammad Nurul Huda, and Khondaker A Mamun. 2021. Human-robot interaction in bengali language for healthcare automation integrated with speaker recognition and artificial conversational entity. In *2021 3rd International Conference on Electrical & Electronic Engineering (ICEEE)*, pages 13–16. IEEE.

Vineel Pratap, Andros Tjandra, Bowen Shi, Paden Tomasello, Arun Babu, Sayani Kundu, Ali Elkahky, Zhaoheng Ni, Apoorv Vyas, Maryam Fazel-Zarandi, and 1 others. 2024. Scaling speech technology to 1,000+ languages. *Journal of Machine Learning Research*, 25(97):1–52.

Sreya Sanyal Puja, Nahian Noor Neha, Ofia Rahman Alif, Tarannaum Jahan Sultan, Md Golam Zel Asmaul Husna, Ishrat Jahan, and Jannatun Noor. 2024. Exploring the barriers to feminine healthcare access among marginalized women in bangladesh and facilitating access through a voice bot. *Heliyon*, 10(14).

MD Rahman, Adnan Alamgir, Shaheedul Haque Chowdhury, Maliha Mushtari, Wasim Anzum, and 1 others. 2024. *A comprehensive NLP-based voice assistant system for streamlined information retrieval in metro rail services of Bangladesh*. Ph.D. thesis, Brac University.

Rajan Saha Raju, Prithwiraj Bhattacharjee, Arif Ahmad, and Mohammad Shahidur Rahman. 2019. A bangla text-to-speech system using deep neural networks. In *2019 International Conference on Bangla Speech and Language Processing (ICBSLP)*, pages 1–5. IEEE.

Fazle Rabbi Rakib, Souhardya Saha Dip, Samiul Alam, Nazia Tasnim, Md Istiak Hossain Shihab, Md Nazmuddoha Ansary, Syed Mobassir Hossen, Marsia Haque Meghla, Mamunur Mamun, Farig Sadeque, and 1 others. 2023a. Ood-speech: A large bengali speech recognition dataset for out-of-distribution benchmarking. *arXiv preprint arXiv:2305.09688*.

Mohammed Rakib, Md Ismail Hossain, Nabeel Mohammed, and Fuad Rahman. 2023b. Bangla-wave: Improving bangla automatic speech recognition utilizing n-gram language models. In *Proceedings of the 2023 12th International Conference on Software and Computer Applications*, pages 297–301.

Md Sazzadul Islam Ridoy, Sumi Akter, and Md Aminur Rahman. 2025. Adaptability of asr models on low-resource language: A comparative study of whisper and wav2vec-bert on bangla. *arXiv preprint arXiv:2507.01931*.

Srijoni Saha and 1 others. 2021. Development of a bangla speech to text conversion system using deep learning. In *2021 Joint 10th International Conference on Informatics, Electronics & Vision (ICIEV) and 2021 5th International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*, pages 1–7. IEEE.

Ahnaf Mozib Samin, M Humayon Kobir, Md Mushtaq Shahriyar Rafee, M Firoz Ahmed, Mehedi Hasan, Partha Ghosh, Shafkat Kibria, and M Shahidur Rahman. 2024. Banspeech: A multi-domain bangla speech recognition benchmark toward robust performance in challenging conditions. *IEEE Access*, 12:34527–34538.

Henning Schulzrinne, Stephen Casner, Ron Frederick, and Van Jacobson. 2003. Rtp: A transport protocol for real-time applications. Technical report.

Robert C Streijl, Stefan Winkler, and David S Hands. 2016. Mean opinion score (mos) revisited: methods and applications, limitations and alternatives. *Multimedia Systems*, 22(2):213–227.

S Subhash, Prajwal N Srivatsa, S Siddesh, A Ullas, and B Santhosh. 2020. Artificial intelligence-based voice assistant. In *2020 Fourth world conference on smart trends in systems, security and sustainability (WorldS4)*, pages 593–596. IEEE.

Md Rakibuz Sultan, Md Moinul Hoque, Farah Ulfath Heeya, Iftiquar Ahmed, Md Redwanul Ferdouse, and Shikder Mejbah Ahmed Mubin. 2021. Adrisya sahayak: A bangla virtual assistant for visually impaired. In *2021 2nd international conference on robotics, electrical and signal processing techniques (ICREST)*, pages 597–602. IEEE.

Sadia Sultana, M Shahidur Rahman, and M Zafar Iqbal. 2021. Recent advancement in speech recognition for bangla: A survey. *International Journal of Advanced Computer Science and Applications*, 12(3).

Mika Sundvall. 2014. Opus audio codec in mobile networks.

Nabila Tasnia, Mahidul Islam, Mahi Shahriar Rony, Nishat Tanzim, Khan Md Hasib, and Mohammad Shafiul Alam. 2023. An overview of bengali speech recognition: Methods, challenges, and future direction. In *2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 0873–0878. IEEE.

Silero Team. 2024. Silero vad: pre-trained enterprise-grade voice activity detector (vad), number detector and language classifier. https://github.com/snakers4/silero-vad.

George Terzopoulos and Maya Satratzemi. 2020. Voice assistants and smart speakers in everyday life and in education. *Informatics in Education*, 19(3):473–490.

Tugstugi. 2023. Bengali ai asr submission. Kaggle dataset. Accessed: 2025-09-19.

Md Rifat Ullah, Md Nafees Mahbub, Md Azizul Hakim, Yeasmin Sultana, and Iftakhar Alam. 2024. Disha: A bilingual humanoid virtual assistant. In *2024 6th International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT)*, pages 457–462. IEEE.

Jean-Marc Valin. 2018. A hybrid dsp/deep learning approach to real-time full-band speech enhancement. In *2018 IEEE 20th international workshop on multimedia signal processing (MMSP)*, pages 1–5. IEEE.

Jean-Marc Valin, Gregory Maxwell, Timothy B Terriberry, and Koen Vos. 2016. High-quality, low-delay music coding in the opus codec. *arXiv preprint arXiv:1602.04845*.

Yijie Xu and Runqi Xu. 2022. Research on interpolation and data fitting: Basis and applications. *arXiv preprint arXiv:2208.11825*.

A Algorithms

A.1 Upsampling

The upsampling algorithm described in Alg. 2 increases the audio sample rate by generating intermediate samples through linear interpolation.

Algorithm 2 Upsampling with linear interpolation increases the sample rate by inserting new sample values between consecutive samples. The intermediate values are calculated using the previous-current sample pair and the slope between them.

```
Require: Input frame $x \in \mathbb{Z}^N$, ratio $r$, previous value $p$
Ensure: Upsampled array $y \in \mathbb{Z}^{N \cdot r}$
$y \leftarrow$ array of zeros with length $N \cdot r$
$k \leftarrow 1$
for $i \leftarrow 1$ to $N$ do
    $cur \leftarrow x[i]$
    $\Delta \leftarrow (cur - p)/r$
    for $j \leftarrow 0$ to $r - 1$ do
        $v \leftarrow p + j \cdot \Delta$
        Clip $v$ into range $[-32768, 32767]$
        $y[k] \leftarrow \lfloor v \rceil$; $k \leftarrow k + 1$
    end for
    $p \leftarrow cur$
end for
return $y$
```

A.2 Downsampling

The downsampling algorithm described in Alg. 3 reduces the audio sample rate by discarding intermediate samples.

Algorithm 3 Downsampling by an integer factor r reduces the sample rate by keeping every $r - th$ sample from the speech signal.

```
Require: Input frame $x \in \mathbb{Z}^L$, ratio $r \in \mathbb{N}, r \geq 1$
Ensure: Downsampled array $y \in \mathbb{Z}^{\lfloor L/r \rfloor}$
$N \leftarrow \left\lfloor \frac{L}{r} \right\rfloor$
$y \leftarrow$ array of zeros with length $N$
$k \leftarrow 1$
for $i \leftarrow 1$ to $L$ step $r$ do
    $y[k] \leftarrow x[i]$;    $k \leftarrow k + 1$
    if $k > N$ then break
    end if
end for
return $y$
```

B The RTP Packet Structure

The structure of a standard RTP packet, which comprises a 12-byte header, is shown in Fig. 6.

C Dataset

C.1 Ben10 Dataset Analysis

The Ben10 dataset (Humayun et al., 2024) comprises speech recordings from 373 speakers across ten distinct regions of Bangladesh. The training set contains over 63 hours of audio sampled at 16 kHz. Table 4 presents the region-wise distribution of audio files within the training set. Since the transcripts of the Ben10 test set are not publicly available, we instead employ the test set of the RegSpeech12 dataset (Hassan et al., 2025) for evaluation in our study.

Training Set	
Region	**Audio File Count**
Barishal	796
Chittagong	1406
Habiganj	940
Kishoreganj	1638
Narail	1488
Narsingdi	1098
Rangpur	1037
Sandwip	1049
Sylhet	2903
Tangail	987
Total	**13342**

Table 4: Training set distribution of the Ben10 dataset

C.2 RegSpeech12 Dataset Analysis

The RegSpeech12 dataset (Hassan et al., 2025) is a spontaneous speech corpus encompassing twelve regional dialects of Bangladesh, comprising approximately 100 hours of speech data. In this study, we utilize the test split of the dataset, which contains around 10 hours of recordings. Table 5 presents the region-wise distribution of the test set.

D Prompt to LLM

To generate responses to spoken user queries transcribed by the dialect-aware ASR system, we provide the prompt illustrated in Fig. 7 to the GPT-4.1-nano model.

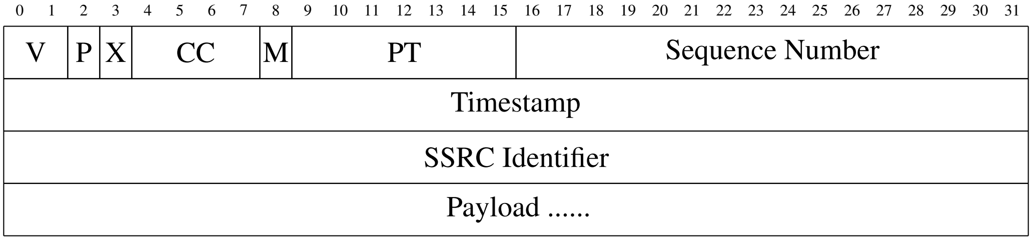


Figure 6: RTP packet structure with a 12-byte header followed by the audio payload.

Test Set	
Region	**Audio File Count**
Barishal	101
Chittagong	176
Comilla	32
Habiganj	117
Kishoreganj	205
Narail	186
Narsingdi	137
Noakhali	28
Rangpur	130
Sandwip	131
Sylhet	762
Tangail	127
Total	**2132**

Table 5: Test set distribution of the RegSpeech12 dataset

System Prompt: You are a helpful chatbot who understands Bengali regional dialects and only speaks standard Bengali. Please be concise and end every sentence with `{|}`.

User Prompt: Please generate a response for only the valid query. For an invalid query, print only a `{$}`. Here is the query in the Bengali regional dialect `{user_query}`.

Figure 7: Prompt for LLM to generate response for query in Bengali regional dialect.

E ASR System Evaluation

E.1 ASR Evaluation Metrics

Word Error Rate (WER) and Character Error Rate (CER) are the standard metrics for evaluating the performances of ASR systems. The formula for Word Error Rate is:

$$\text{WER} = \frac{S + D + I}{N} \times 100\%, \qquad (1)$$

where S is the number of substitutions, D is the number of deletions, I is the number of insertions, and N is the total number of words in the reference transcription.

Similarly, the formula for Character Error Rate is:

$$\text{CER} = \frac{S_c + D_c + I_c}{N_c} \times 100\%, \qquad (2)$$

where S_c is the number of substitutions, D_c is the number of deletions, and I_c is the number of insertions at the character level. N_c is the total number of characters in the reference transcription.

E.2 Impact of Processing on BRDialect ASR System

Processing pipelines play a critical role in shaping the performance of ASR systems. Table 6 presents the impact of 5-gram KenLM decoding, noise cancellation, text normalization, and punctuation removal on the BRDialect ASR system.

E.3 Processing Times of ASR Systems

The inference time for audio files varies across different ASR systems, as shown in Table 7. The Whisper model exhibits significantly slower processing compared to BRDialect and IndicWav2Vec. Although the IndicWav2Vec model is slightly faster than BRDialect, its overall performance is considerably lower. In contrast, the processing time of BRDialect falls within an acceptable range for real-time communication applications.

F TTS Systems Evaluation

In this study, we conduct extensive experiments with three open-source TTS systems, all of which are based on the VITS architecture (Kim et al., 2021): MMS-TTS-Ben (Pratap et al., 2024), and the VITS-Bengali Male and Female variants (Hossen, 2023). To evaluate the quality of the synthesized speech, we employ the mean opinion score (MOS) metric.

KD	NC	UN	PR	WER	CER
✓	×	×	×	0.827	0.442
✓	×	×	✓	0.796	0.420
✓	×	✓	×	0.781	0.430
✓	×	✓	✓	**0.741**	**0.406**
✓	✓	×	×	0.838	0.493
✓	✓	×	✓	0.813	0.474
✓	✓	✓	×	0.801	0.479
✓	✓	✓	✓	0.770	0.458
×	×	×	×	0.865	0.452
×	×	×	✓	0.834	0.444
×	×	✓	×	0.834	0.429
×	×	✓	✓	0.793	0.419
×	✓	×	×	0.876	0.497
×	✓	×	✓	0.853	0.487
×	✓	✓	×	0.853	0.478
×	✓	✓	✓	0.822	0.467

Table 6: Performance of the BRDialect ASR system on the processing settings of 5-gram KenLM Decoding (KD), Noise Cancellation (NC), Unicode Normalization (UN), and Punctuation Removal (PR).

For a robust comparison, we curate a diverse set of texts from the BanSpeech dataset (Samin et al., 2024), which contains audio-text pairs across thirteen categories. From each category, one representative text sample is selected, forming the test dataset, as detailed in App. F.1. Each text sample is synthesized using all three TTS models. To assess naturalness and perceived audio quality, an experienced human rater with expertise in audio signal processing independently rates each generated audio on a 1 to 5 scale, where 1 indicates very poor quality and 5 represents excellent quality (Streijl et al., 2016).

The results, summarized in Fig. 8, indicate that the VITS-Bengali Male model achieves the highest average MOS score of 4.49, producing speech that is perceived as highly natural and pleasant. The VITS-Bengali Female model achieves an average MOS of 4.40, which is also suitable for end-to-end speech assistant systems. In contrast, MMS-TTS-Ben performs the lowest, with an average MOS score of 3.66, approximately 22.7% lower than VITS-Bengali Male, indicating reduced suitability for end-to-end applications.

F.1 Text Samples from the BanSpeech Dataset

The BanSpeech dataset (Samin et al., 2024) comprises audio–text pairs spanning diverse categories. For evaluating the TTS systems, we use the text samples from this dataset, as illustrated in Fig. 9.

Audio	**Preprocessing Times (s) ↓**		
	Whisper	IndicWav2Vec	BRDialect
sylhet_1	3.07	0.65	**0.77**
sylhet_2	3.61	0.59	**1.00**
sylhet_3	3.99	1.36	**1.31**
sylhet_4	3.49	0.80	**0.97**

Table 7: Processing time analysis of four audio files with an average duration of 8.75 s from the Sylhet region.

User Query	**End-to-End Delay (s) ↓**
Query_1	5
Query_2	4
Query_3	5
Query_4	5
Query_5	4
Query_6	6
Query_7	5
Query_8	5
Query_9	5
Query_10	5
Average	**4.9**

Table 8: End-to-end delay analysis of the BanglaTalk system. Delay is calculated from the end of the user query to the start time of getting the audio response from the speech assistant.

G End-to-End delay of BanglaTalk System

To quantify the end-to-end delay of the BanglaTalk system, the delay for ten user queries is measured. Table 8 shows the results of this experiment. The BanglaTalk system has a low end-to-end delay of 4.9 s.

H User Study

To evaluate the user experience with the BanglaTalk system, a rigorous qualitative analysis is performed. Table 9 presents a summary of the experimental results, while Fig. 10 illustrates user interactions with the BanglaTalk system from two regions of Bangladesh – Sylhet and Mymensingh.

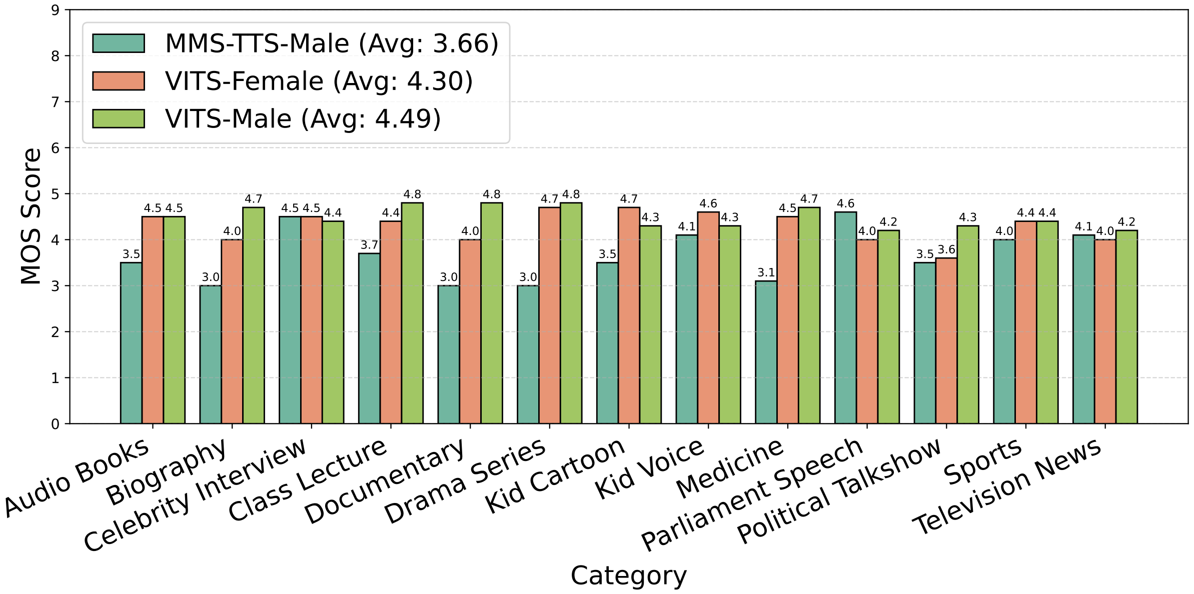
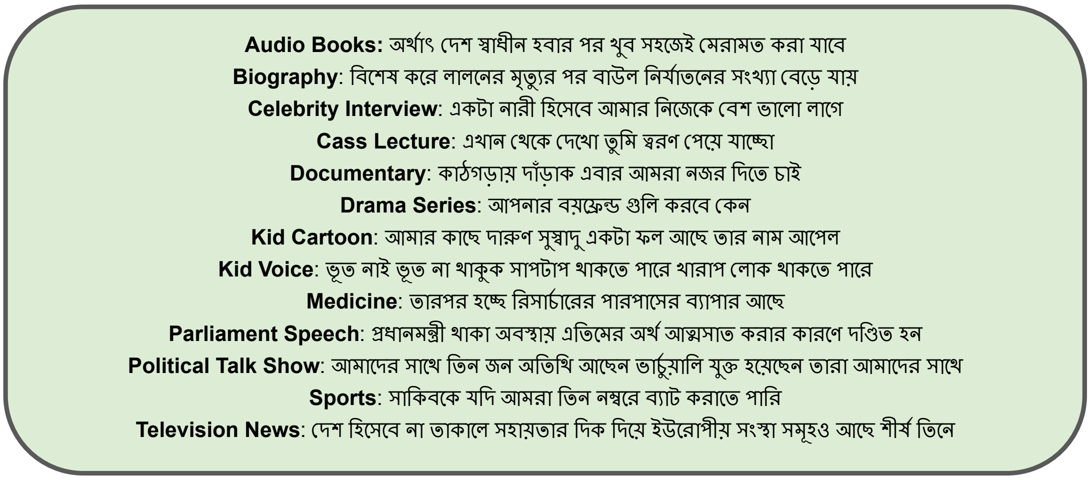


Figure 8: Mean Opinion Score (MOS) of three TTS models on the subset of the BanSpeech dataset.

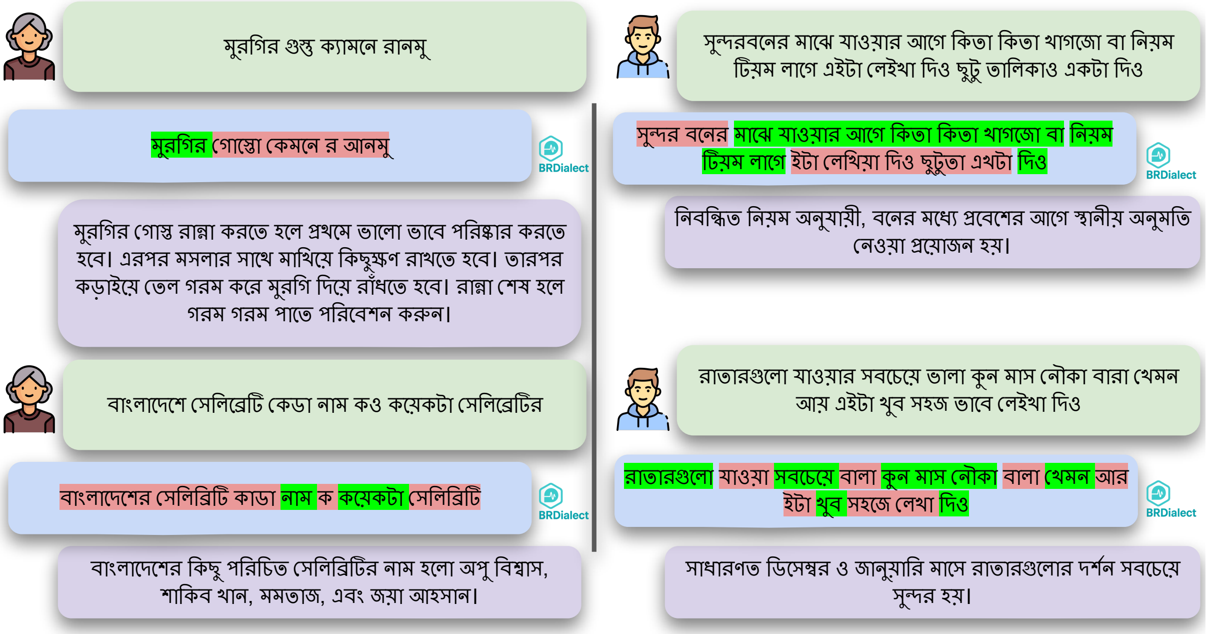


Audio Books: অর্থাৎ দেশ স্বাধীন হবার পর খুব সহজেই মেরামত করা যাবে
Biography: বিশেষ করে লালনের মৃত্যুর পর বাউল নির্যাতনের সংখ্যা বেড়ে যায়
Celebrity Interview: একটা নারী হিসেবে আমার নিজেকে বেশ ভালো লাগে
Cass Lecture: এখান থেকে দেখো তুমি ত্বরণ পেয়ে যাচ্ছো
Documentary: কাঠগড়ায় দাঁড়াক এবার আমরা নজর দিতে চাই
Drama Series: আপনার বয়ফ্রেন্ড গুলি করবে কেন
Kid Cartoon: আমার কাছে দারুণ সুস্বাদু একটা ফল আছে তার নাম আপেল
Kid Voice: ভূত নাই ভূত না থাকুক সাপটাপ থাকতে পারে খারাপ লোক থাকতে পারে
Medicine: তারপর হচ্ছে রিসার্চারের পারপাসের ব্যাপার আছে
Parliament Speech: প্রধানমন্ত্রী থাকা অবস্থায় এতিমের অর্থ আত্মসাত করার কারণে দণ্ডিত হন
Political Talk Show: আমাদের সাথে তিন জন অতিথি আছেন ভার্চুয়ালি যুক্ত হয়েছেন তারা আমাদের সাথে
Sports: সাকিবকে যদি আমরা তিন নম্বরে ব্যাট করাতে পারি
Television News: দেশ হিসেবে না তাকালে সহায়তার দিক দিয়ে ইউরোপীয় সংস্থা সমূহও আছে শীর্ষ তিনে

Figure 9: Selected text samples from the BanSpeech dataset to evaluate the performance of the TTS systems by calculating MOS scores.

User	**Region**	**Rating**					**Mean ↑**
		Query_1	Query_2	Query_3	Query_4	Query_5	
User_1	Sylhet	3.0	4.0	2.5	3.0	5	3.5
User_3	Sylhet	3.4	3.7	3.2	3.3	4.0	3.52
User_2	Mymensingh	2.0	4.0	3.8	4.2	3.3	3.46
User_4	Mymensingh	5.0	3.0	4.5	3.5	4.0	4.0
Mean							**3.62**

Table 9: User ratings on a 1–5 scale (where 1 indicates a poor experience and 5 indicates an excellent experience) for conversational interactions with the BanglaTalk speech assistant. Ratings are collected from four users representing two regional dialects – Sylhet and Mymensingh.



মুরগির গুস্ত ক্যামনে রানমু

মুরগির গোস্তো কেমনে র আনমু

মুরগির গোস্ত রান্না করতে হলে প্রথমে ভালো ভাবে পরিষ্কার করতে হবে। এরপর মসলার সাথে মাখিয়ে কিছুক্ষণ রাখতে হবে। তারপর কড়াইয়ে তেল গরম করে মুরগি দিয়ে রাঁধতে হবে। রান্না শেষ হলে গরম গরম পাতে পরিবেশন করুন।

বাংলাদেশে সেলিব্রেটি কেডা নাম কও কয়েকটা সেলিব্রেটির

বাংলাদেশের সেলিব্রিটি কাডা নাম ক কয়েকটা সেলিব্রিটি

বাংলাদেশের কিছু পরিচিত সেলিব্রিটির নাম হলো অপু বিশ্বাস, শাকিব খান, মমতাজ, এবং জয়া আহসান।

সুন্দরবনের মাঝে যাওয়ার আগে কিতা কিতা খাগজো বা নিয়ম টিয়ম লাগে এইটা লেইখা দিও ছুটু তালিকাও একটা দিও

সুন্দর বনের মাঝে যাওয়ার আগে কিতা কিতা খাগজো বা নিয়ম টিয়ম লাগে ইটা লেখিয়া দিও ছুটুতা এখটা দিও

নিবন্ধিত নিয়ম অনুযায়ী, বনের মধ্যে প্রবেশের আগে স্থানীয় অনুমতি নেওয়া প্রয়োজন হয়।

রাতারগুলো যাওয়ার সবচেয়ে ভালা কুন মাস নৌকা বারা খেমন আয় এইটা খুব সহজ ভাবে লেইখা দিও

রাতারগুলো যাওয়া সবচেয়ে বালা কুন মাস নৌকা বালা খেমন আর ইটা খুব সহজে লেখা দিও

সাধারণত ডিসেম্বর ও জানুয়ারি মাসে রাতারগুলোর দর্শন সবচেয়ে সুন্দর হয়।

Figure 10: Conversational interactions with the BanglaTalk speech assistant using the regional dialects of Mymensingh (left) and Sylhet (right). Each example includes the user's query, the transcript generated by the BRDialect ASR system, and the corresponding LLM-generated response. **Although the WER of BRDialect is relatively high in some cases, the LLM effectively captures the context and produces appropriate responses.**