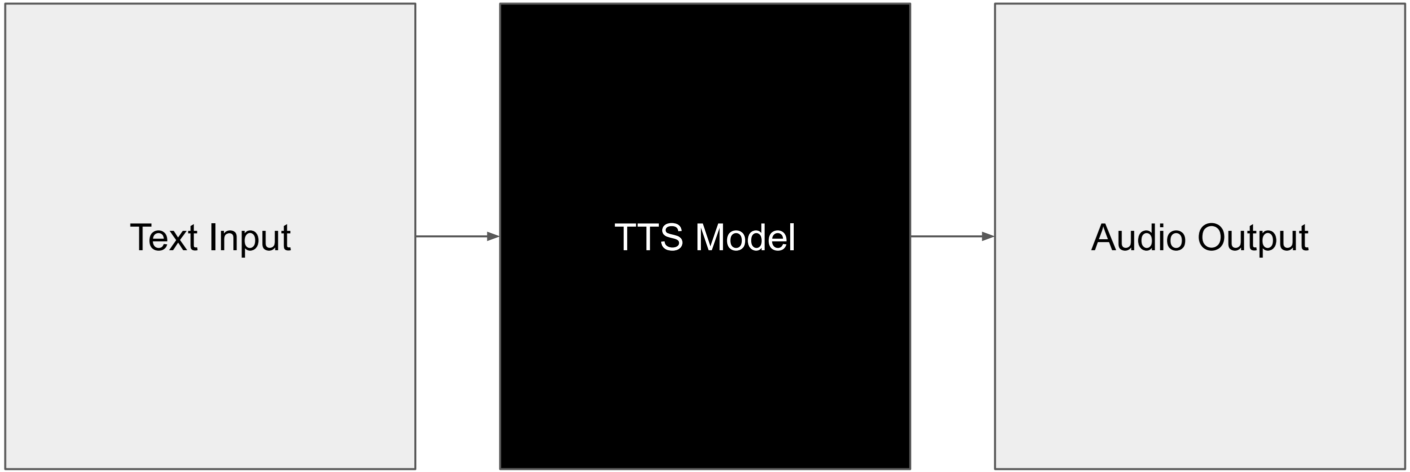
STFT-GradTTS: A Robust, Diffusion-based Speech Synthesis System with iSTFT decoder for Bangla

Mushahid Intesum
Abdullah Ibne Masud
Md Ashraful Islam
Dr Md Rezaul Karim

What is Speech Synthesis?

- Text-to-speech (TTS) systems turn user text into audible voice data.



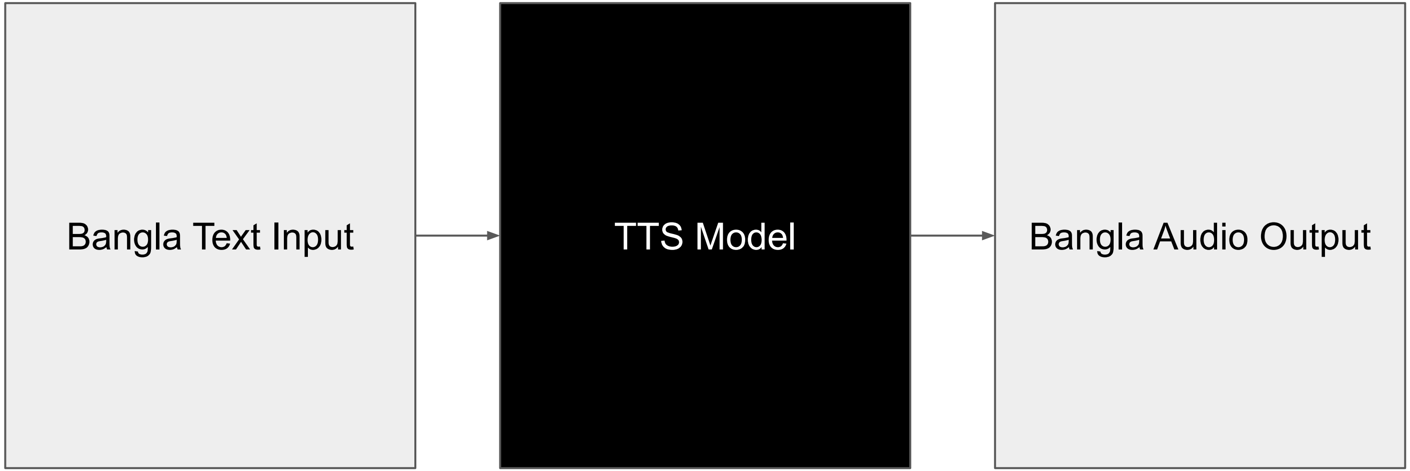
What’s Inside the Black Box?

Text Analysis Module
- Normalization
- Extraction of linguistic features such as POS tagging, word disambiguation and prosodic features

Acoustic Model
- Converts the text into an audio representation. Such as: mel spectrograms

Vocoder
- Uses the mel spectrograms to create audio signals

Problem Statement



Motivation For This Project

- A project by MIT where device on finger reads text on screen and audio is generated

Motivation For This Project

- Application utters prompts, instructions, lists, directions

Motivation For This Project

- Automate helpline service workflow

Existing Works

Model	Contribution	Limitations
Tacotron2	End-to-end TTS architecture	Did not allow parallel computation, resulting in slow inference
DeepVoice3	Used CNN, faster training	Could not handle long sequences

Existing Works

Model	Contribution	Limitations
VITS	Uses variational autoencoders to achieve state-of-the-art results	Slow training
GradTTS	Diffusion model generator that can produce good audio with fast training	Audio sometimes sound rushed

Existing Works

Model	Contribution	Limitations
Tanzir et al [1]	Aimed to produce a text-to-speech model for Bangla by text normalization	Lost phonetic characteristics due to normalization, robotic audio output
Khandaker et al [2]	Create Bangla TTS system by Romanizing Bangla text	Romanization loses phonetic characteristic of Bangla
Proposed Model	A diffusion-based generator with stochastic duration predictor. Prepared a large audio dataset	Unable to pronounce all word properly

[2] Khandaker Ahmed, Prianka Mandal, and B M Mainul Hossain. Text to speech synthesis for bangla language. International Journal of Information Engineering and Electronic

[1] Tanzir Islam Pial, Shahreen Salim, Shabbir Ahmed, an d Hasnain Heickal. End-to-end speech synthesis for bangla with text normalization. pages 66–71, 07 2018

Challenges in Bangla TTS

- Limited resources, very few datasets
- Complicated phonetic and phonological structure
- Limited works done in this field
- Lack of benchmarks

Dataset

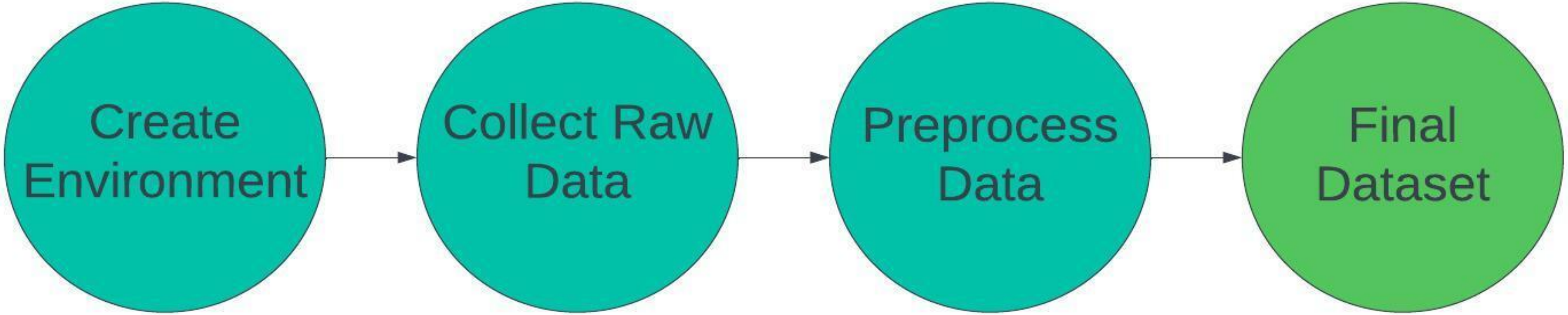
- Existing dataset for bangla TTS
  - Google Bangla TTS dataset (2.94 hours)
  - Mozilla Bangla TTS dataset (1.272 hours)
- Existing datasets are not up to the standard
  - Small sized dataset
  - Multiple dialect (Bangladeshi Bangla, Indian Bangla)
  - Multi speaker (Mozilla has 22897 total voices)
  - Noisy

Dataset

Hence, we need a new dataset

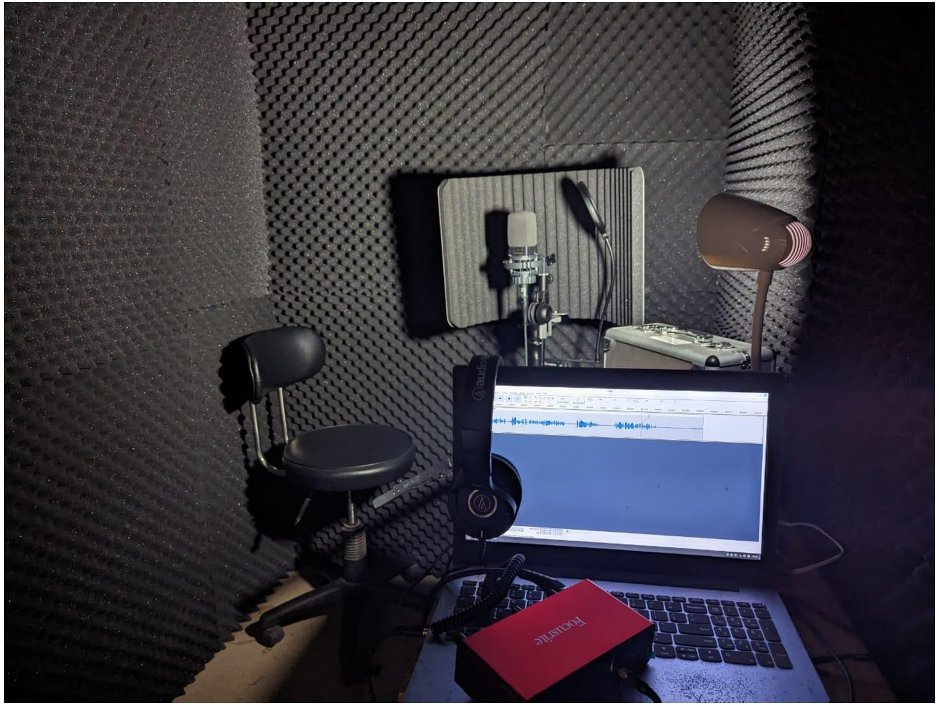
- Existing dataset for bangla TTS
  - Google Bangla TTS dataset (2.94 hours)
  - Mozilla Bangla TTS dataset (1,272 hours)
- Existing datasets are not up to the standard
  - Small sized dataset
  - Multiple dialect (Bangladeshi Bangla, Indian Bangla)
  - Multi speaker (Mozilla has 22897 total voices)
  - Noisy

Dataset Creation



Environment Setup

- Room Setup:
  - Echo/proof room design using acoustic foam
  - Noise reduction measures to deter external noises
- Equipment Utilized:
  - High-quality Neumann TLM 103 microphone
  - Digital-to-analog converter (DAC)
  - Reflector for optimal sound capture



Raw Data Collection

Collecting Raw Data

- 27.5 hours of diverse audio data
  - Average track size : 12 minutes
  - Average time taken to make a track : 25 minutes
- Only included texts written in চলিত প্রমিত ভাষা

Raw Data Collection (*Cont.)*

Collecting Raw Data

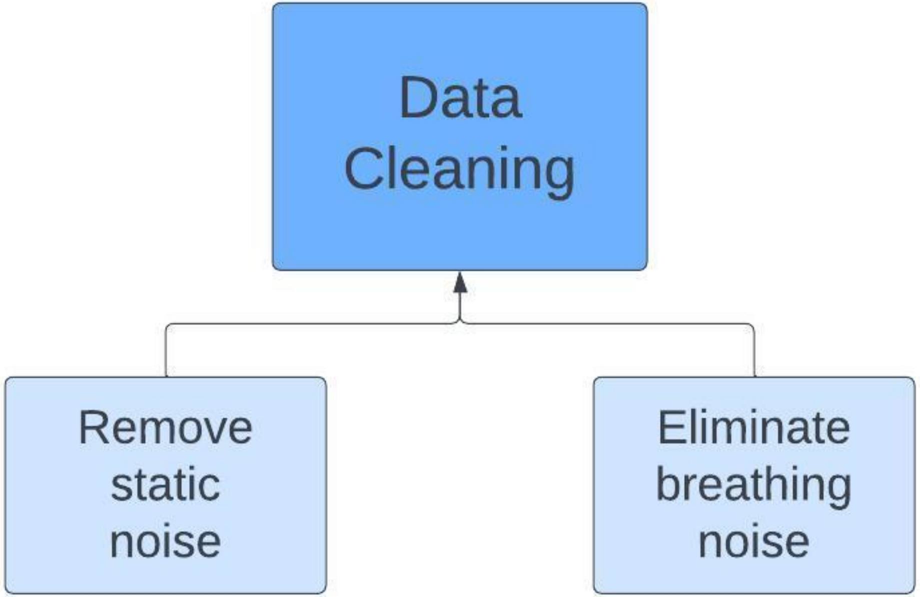
- Data Selection Criteria:
    - Different genres (drama, novel,autobiography, news article etc)
    - Incorporating complex and compound sentences
        - আমি যখন আসি তখন সে চলে যায়।
        - বিদ্যালয়ে যাব এবং মন দিয়ে পড়া শুনবো

Raw Data Collection (*Cont.)*

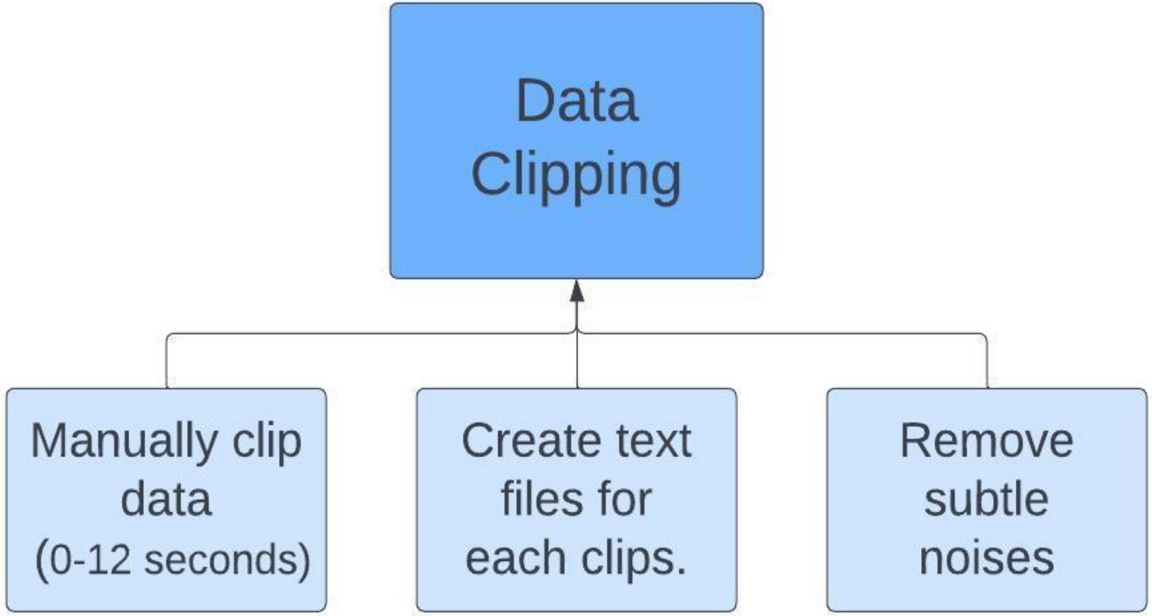
- Data Selection Criteria
  - Including first-person,second-person and third-person sentences
    - আমি পড়াশোনা করি
    - তুমি পড়াশোনা কর
    - সে পড়াশোনা করে

Data Preprocessing

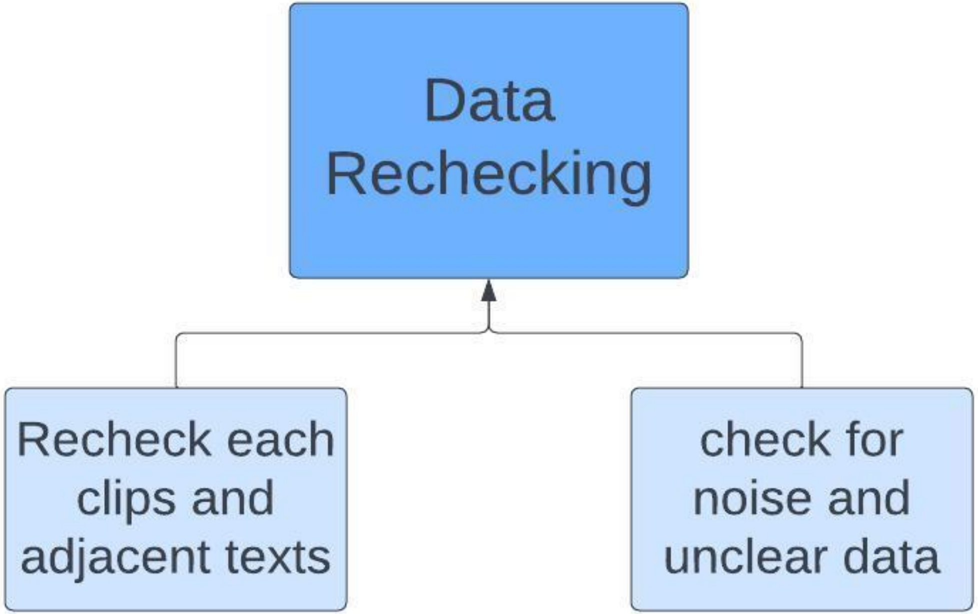
Data Preprocessing (cont.)



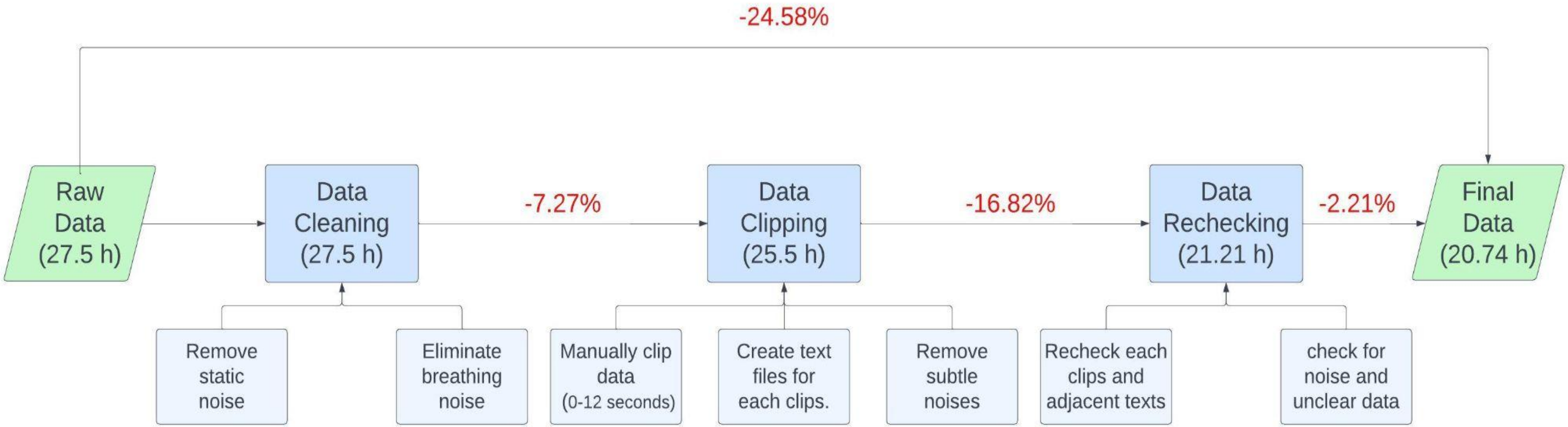
Data Preprocessing (cont.)



Data Preprocessing (cont.)

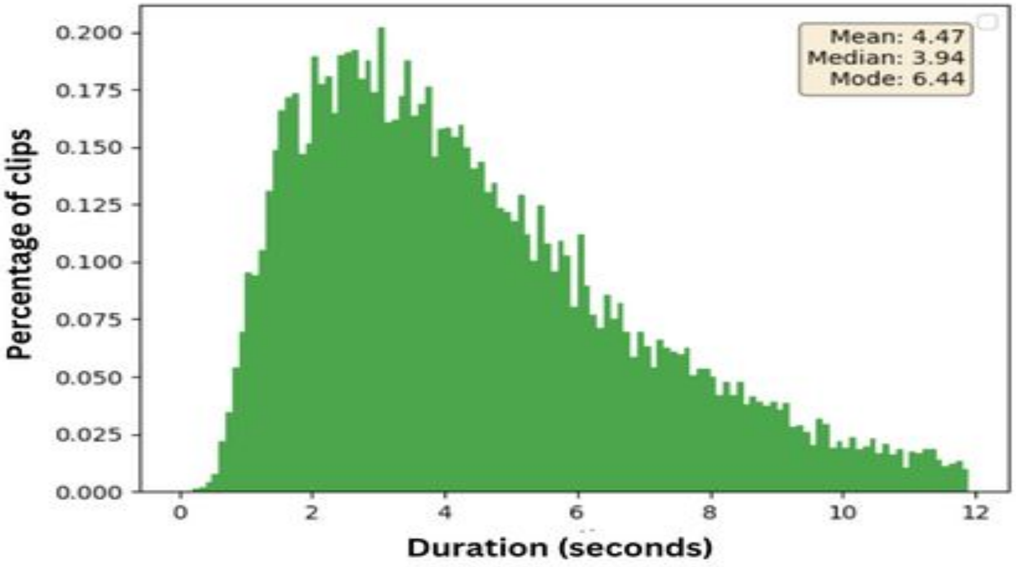


Data Preprocessing (cont.)



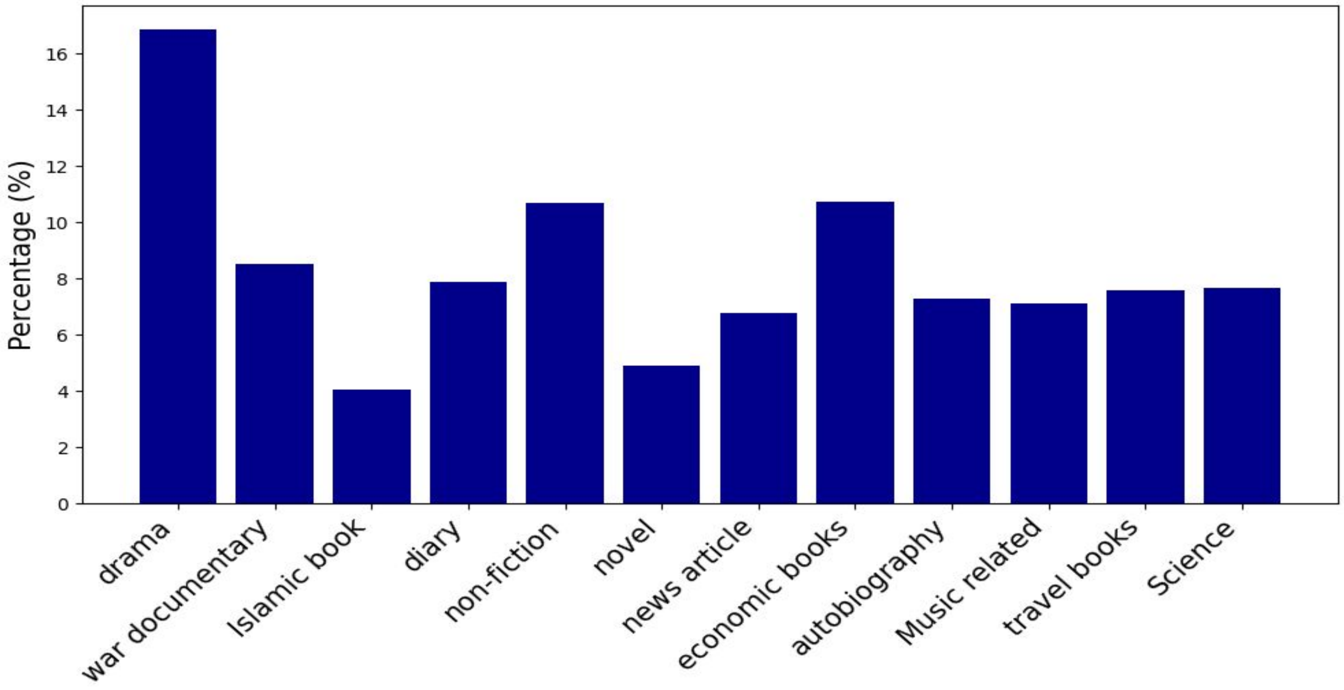
Data Metrics & Statistics

Audio clips frequency (in seconds)



Data Metrics & Statistics

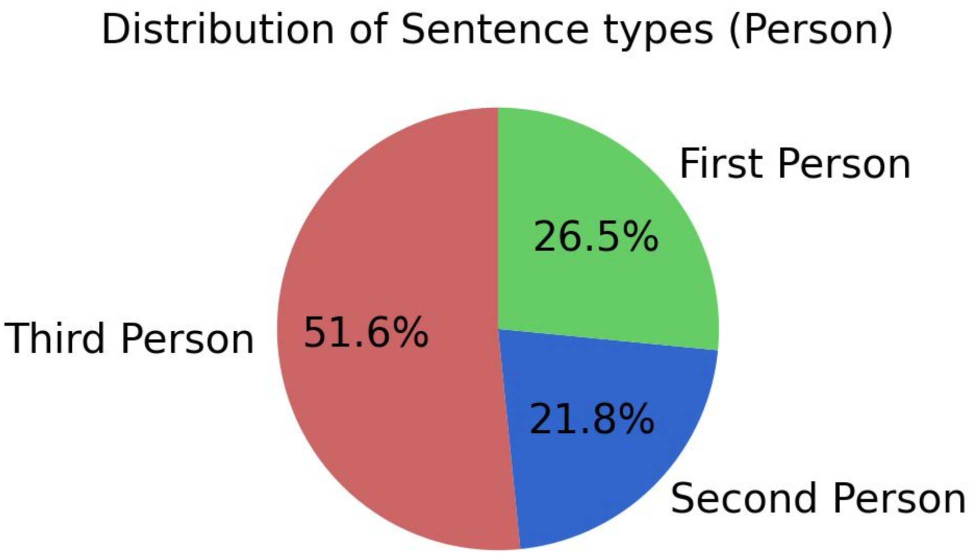
Distribution of different genre



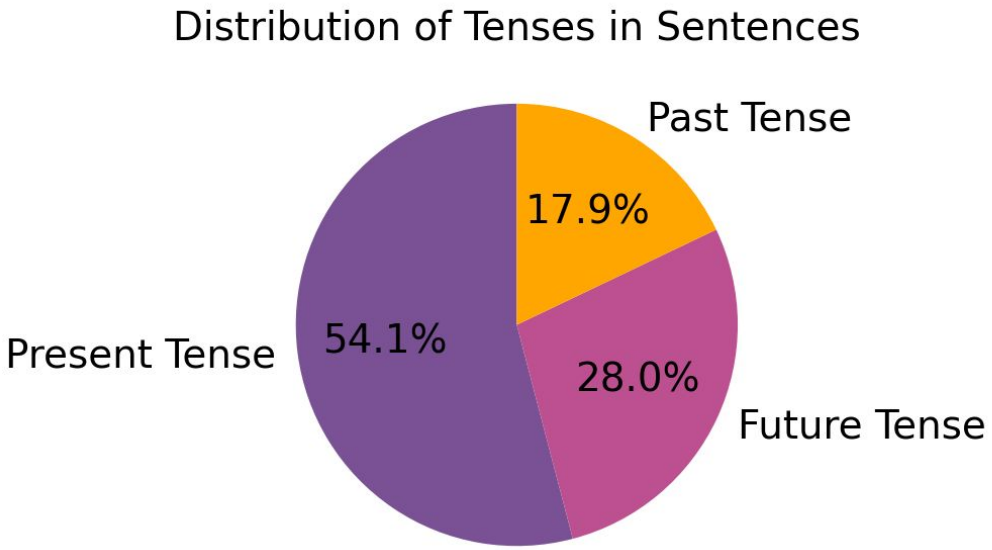
Data Metrics & Statistics

Metrics	Value
Clips count	14989
Total length of clips	20.74 hours
Mean length of clips	165486
Word count	4.98
Average word count in a clip	11.04
Unique Word count	26448
Unique Zuktakkhor	230
Total number of sentences	17051
Interrogative sentence	1275
Exclamatory sentence	380

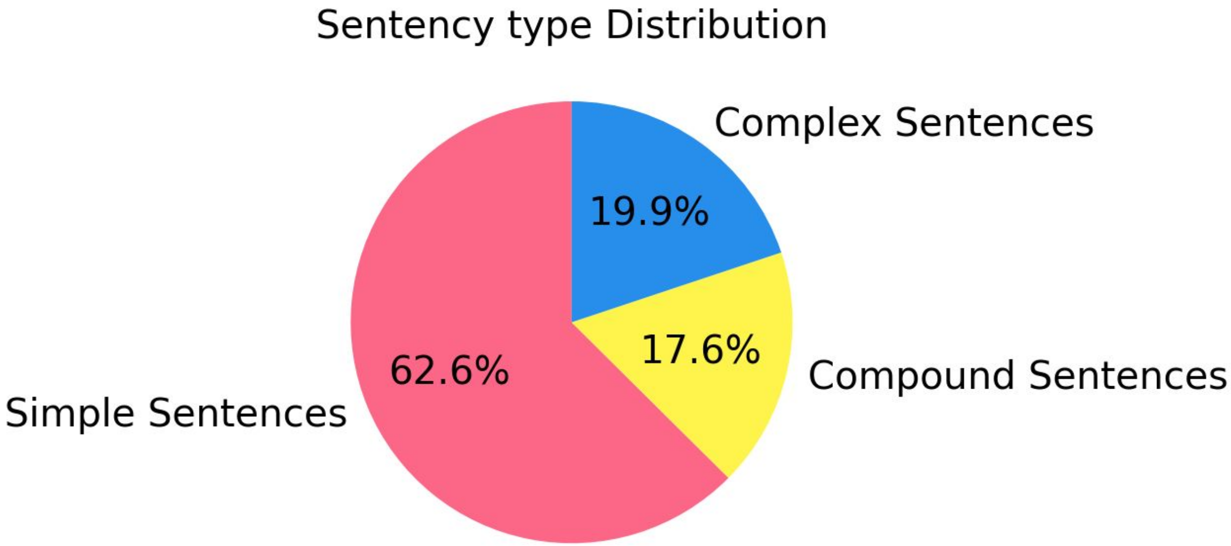
Data Metrics & Statistics



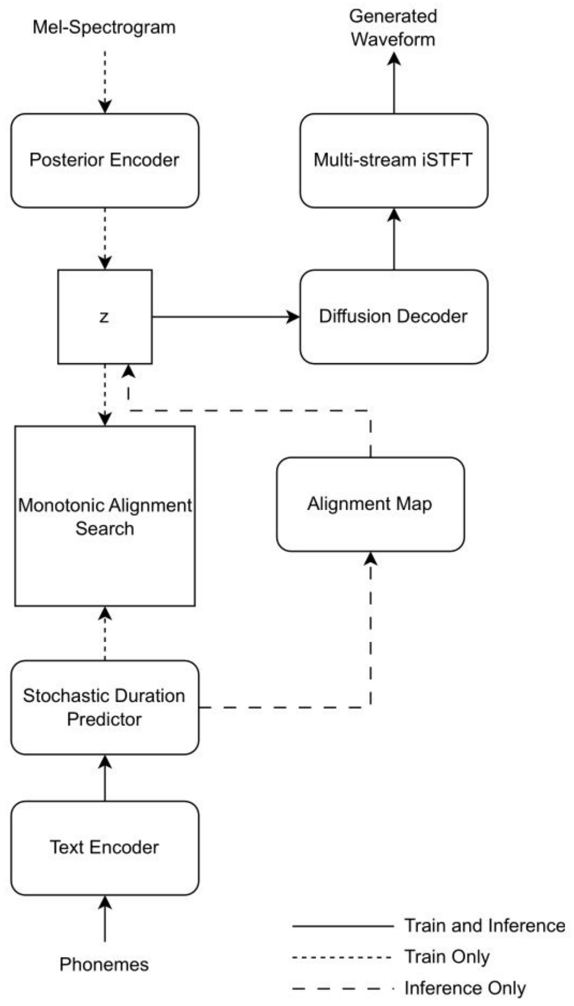
Data Metrics & Statistics



Data Metrics & Statistics

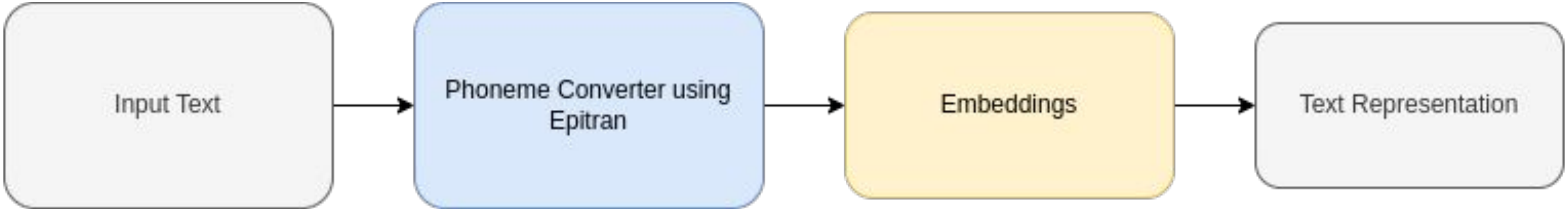


Proposed Model



Proposed Model (contd.)

- Text Encoder
  - Converts text to phonemes and applies embeddings



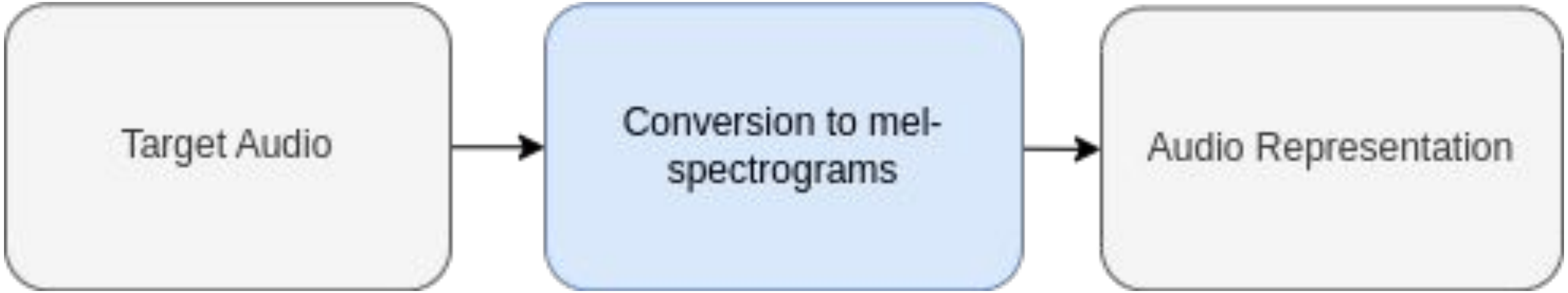
Why Phonemes as Text Representations?

- Textual representations cannot capture the difference of pronunciation of the same letter in different words
- Phonemes provide a mapping of such characteristic

- অস্পৃশ্য : ɔspriʃʃo
- আসা : aʃa

Proposed Model (contd.)

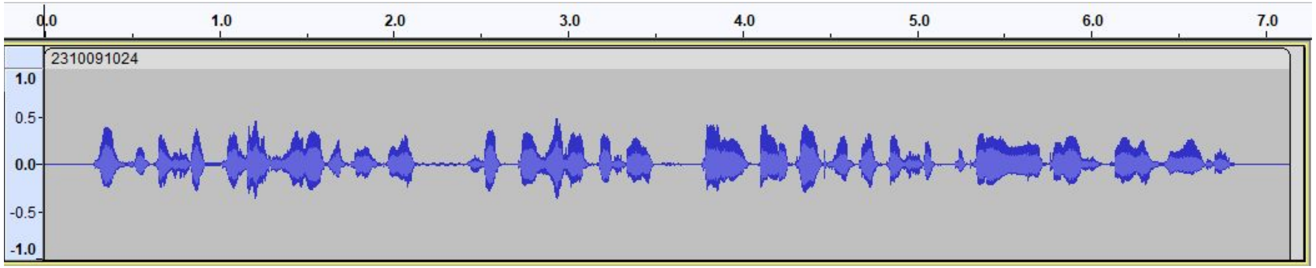
- Audio Encoder
  - Converts audio to mel-spectrograms

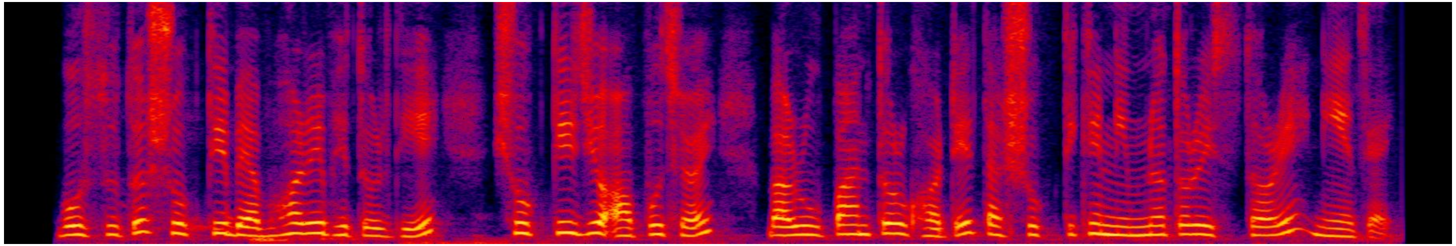


Why Mel Spectrograms?

- Humans perceive audio logarithmically; better able to discern pitch of audio at lower frequencies than at higher frequencies
- Regular spectrograms cannot capture that as those map audio based on frequencies
- Mel spectrograms map audio at a logarithmic scale

Why Mel Spetrograms? (contd)





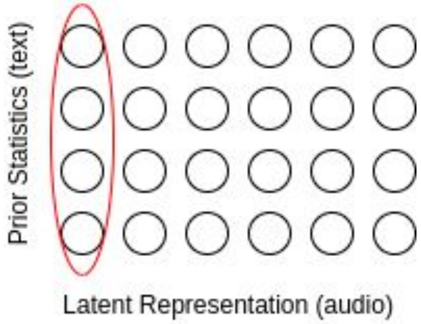
Proposed Model (contd.)

- Alignment Module
  - Aligns duration predicted text embeddings and audio representations
  - Done using a DP algorithm called Monotonic Alignment Search
  - Necessary to find a direct mapping between audio and textual representations

Proposed Model (contd.)

- Stochastic Duration Prediction
  - Main idea: Humans read same sentence at different lengths
  - Predict a phoneme duration distribution instead of a fixed value
  - The duration predictor is a normalizing flow network

Stochastic Duration Predictor



- Duration values for input text tokens, *d*, are taken summing columns
- Random variables *u* and *v* introduced for variational dequantization and variational data augmentation respectively
- u ∈ [0,1) to keep *d-u* positive
- *v, d* concatenated channel-wise to make a higher dimensional latent representation

Stochastic Duration Predictor

$$log p_\theta(d|c_{text}) \geq E_{q_\phi(u,v|d,c_{text})}[log \frac{p_\theta(d-u, v|c_{text})}{q_\phi(u, v|d, c_{text})}]$$

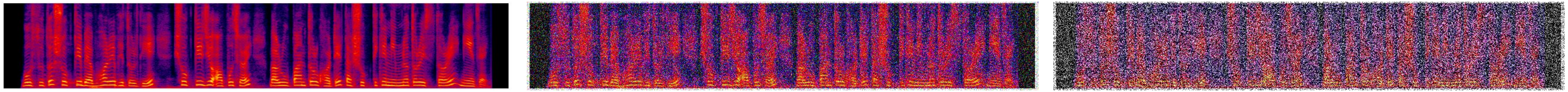
pΘ = prior distribution of tokens conditioned on input text
qΦ = posterior distribution of tokens conditioned on input text

Proposed Model (contd.)

- Diffusion-based Generator
    - Uses a diffusion-based generator to generate audio from text
    - Adds noise to target audio
    - Generate audio from corrupted noise which is aligned to duration aligned input text
    - Model calculates reconstruction loss of target audio from noisy data

Diffusion-based Generator (contd.)

- Forward Diffusion: *convert spectrogram to standard Gaussian noise*

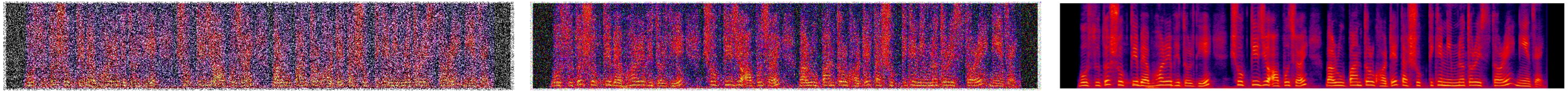


$$dX_t = \frac{1}{2}\Sigma^{-1}(\mu - X_t)\beta_t dt + \sqrt{\beta_t}W_t$$

Here
$\sum$=covariance, βt, Wt=noise schedule parameters

Diffusion-based Generator (contd.)

- Reverse Diffusion: *convert corrupted spectrogram back to original form*



$$dX_t = \frac{1}{2}(\Sigma^{-1}(\mu - X_t) - \Delta log p_t(X_t))\beta_t dt$$

Why a Diffusion-based Generator?

- Better results than Encoder-Decoder (*Tacotron2*) and CNN (*DeepVoice* models) based models
- More stable training than GAN-based (*VITS*) models

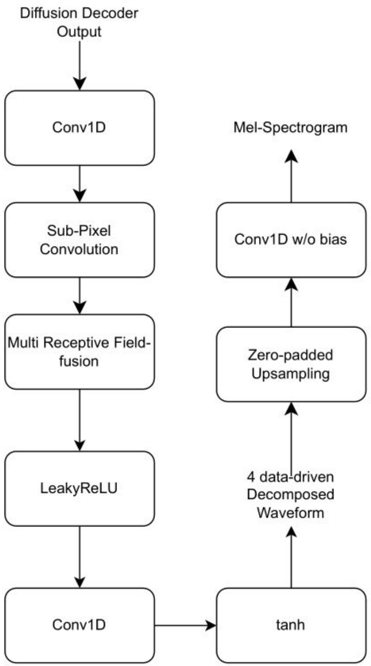
Multi-Stream iSTFT Block

- To further enhance synthesized speech quality, we implement a multi-resolution STFT loss during training
- This loss function evaluates the discrepancy between predicted and ground truth signals in the frequency domain across multiple resolutions, comprising:
  - Spectral Convergence Loss
  - Log STFT Magnitude Loss

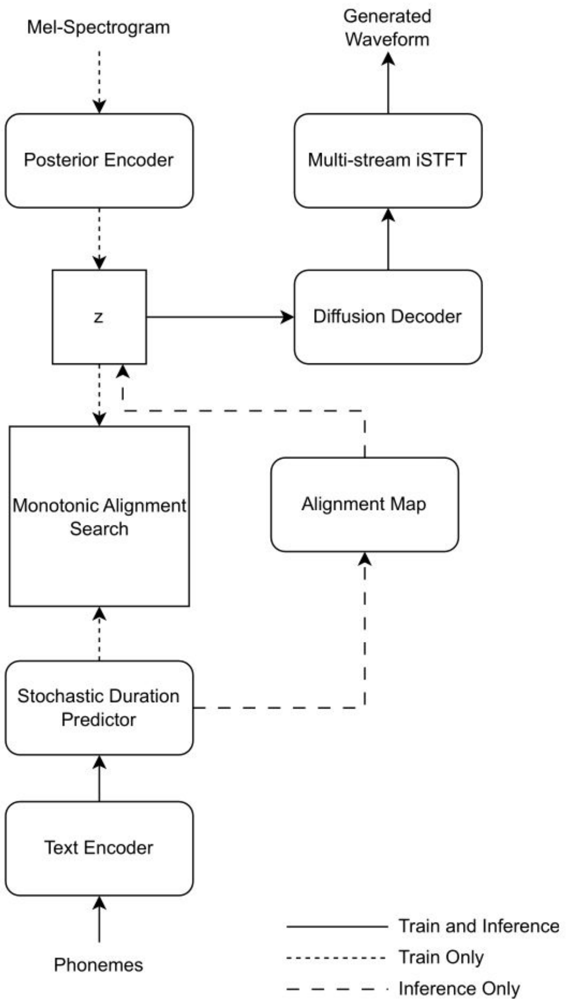
Multi-Stream iSTFT Block (Contd.)

- **Spectral Convergence Loss:** Measures differences in overall spectral structure between predicted and ground truth signals
- **Log STFT Magnitude Loss:** Quantifies differences in log-scale magnitudes of STFT spectra, preserving fine-grained spectral details

Multi-Stream iSTFT Block (Contd.)



Model Overview



Loss Functions

- Encoder Loss
  - The mean-square error loss between the target mel-spectrogram and aligned input text

$$L_{enc} = -\sum_{j=1}^{F} log\phi(y_j; \mu_{A(j)}, I)$$

- Diffusion Loss
  - Average of noise estimations from generator at different timesteps

$$L_{diff} = E_{X_0,t}[\lambda E_\eta[\|s_\theta(X_t, \mu, t) + \frac{\eta_t}{\sqrt{\lambda_t}}\|_2^2]]$$

Loss Functions (contd.)

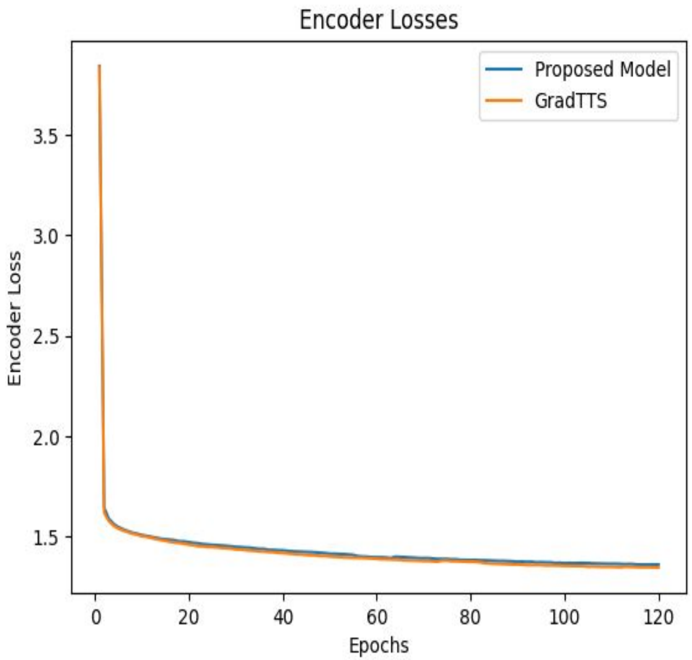
- Duration Loss
  - Negative variational lower-bound of tokenized input conditioned on input text

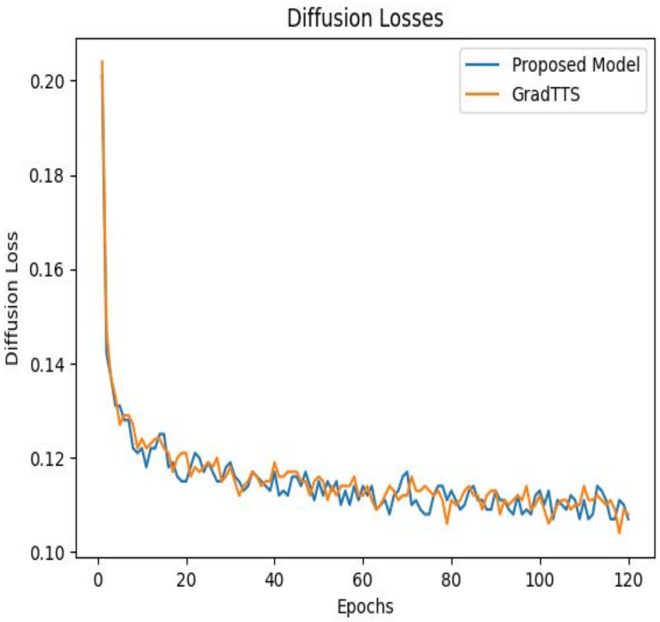
$$L_{dur} = -E_{q_\phi(u,v|d,c_{text})}[log\frac{p_\theta(d-u,v|c_{text})}{q_\phi(u,v|d,c_{text})}] + E_{q_\phi}[log(q_\phi(c_{text}))]$$

Model Evaluation

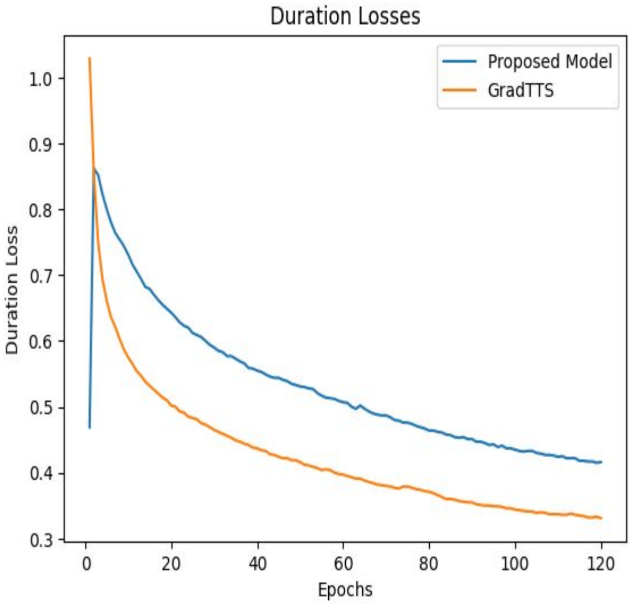
- We compare our model's performance with respect to GradTTS
- Compare model convergence, audio quality, speech variation quality, quality of audio with context predictor and inference speeds

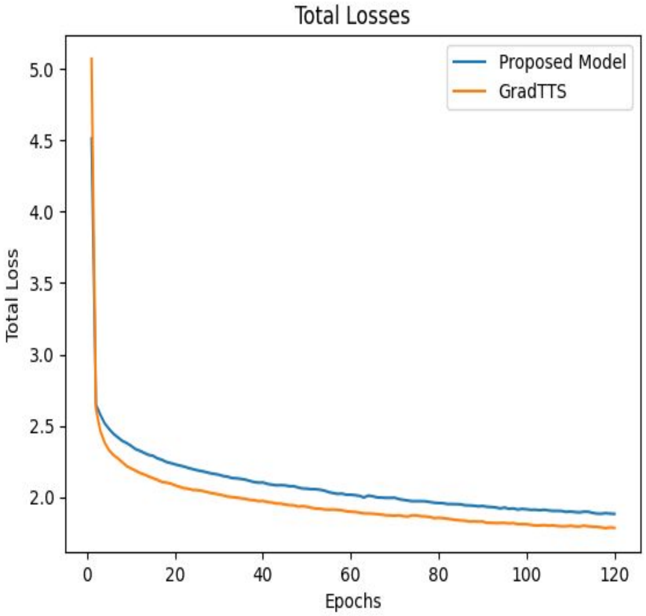
Model Convergence





Model Convergence





Performance Metrics

- Evaluation using Mean Opinion Score
- A subjective metric where audio samples are given to people to rate them between a scale of 1 to 5
- Average score is taken
- Higher the score, the better

Why Mean Opinion Score?

- There is an absence of a truly objective metric
- TTS output heavily relies on human subjectivity rather than objective values
- People will score values based on naturalness, clarity and expressiveness, which cannot be calculated with objective metrics
- It introduces variability and objectivity

Results

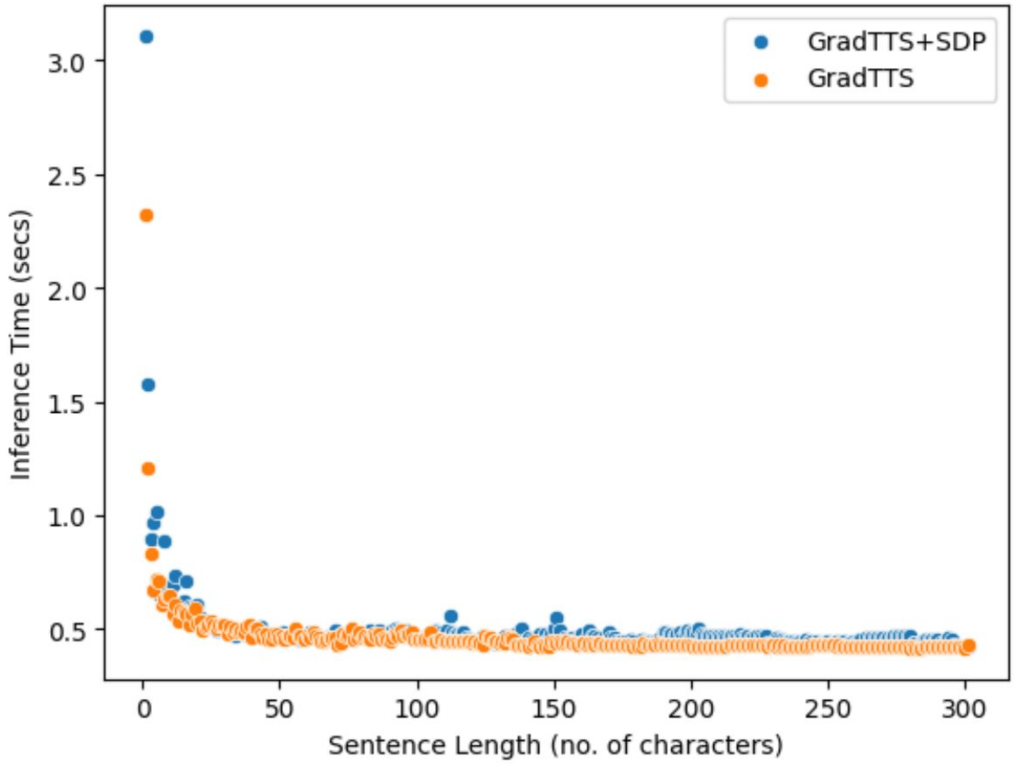
- Mean Opinion Scores calculated with a confidence interval of 95%

Model	Parameters	MOS
Ground Truth	-	4.56 ± 0.06
GradTTS	15,180,889	3.34 ± 0.06
GradTTS with SDP	16,225,928	3.47 ± 0.07
STFT-GradTTS	17,213,061	3.89 ± 0.05

Model Efficiency Scores

- Measured model efficiency with Real-Time Factor score, the time it takes to generate 1 second of audio
- Higher parameter counts have not negatively affected inference times

Model Efficiency Scores



Conclusion

- Prepared a single-speaker audio dataset that is more than 18 hour long
- Prepared a audio dataset metric for future data collection
- Proposed a TTS system for Bangla that focuses on producing more audio that have more natural sounding duration
- Showed our model has better expressiveness and naturalness than our baseline, GradTTS

Thank You