音樂生成 Music Generation

成員：

110321015 陳奕羶　　110321018 張簡雲翔　　110321064 劉德權

指導老師：

楊峻權老師

Outline

- Introduction
- Background
- Model Architecture
- Dataset & Training
- Experiment
- Result

Introduction

Introduction

- Text to Text : ChatGPT , Gemini
- Text to Image : Stable Diffusion , Dall-E

Our goal :

- Text to Music : MusicGen
- Paper : ***Simple and Controllable Music Generation – (Meta , 2023)***

Background

Background

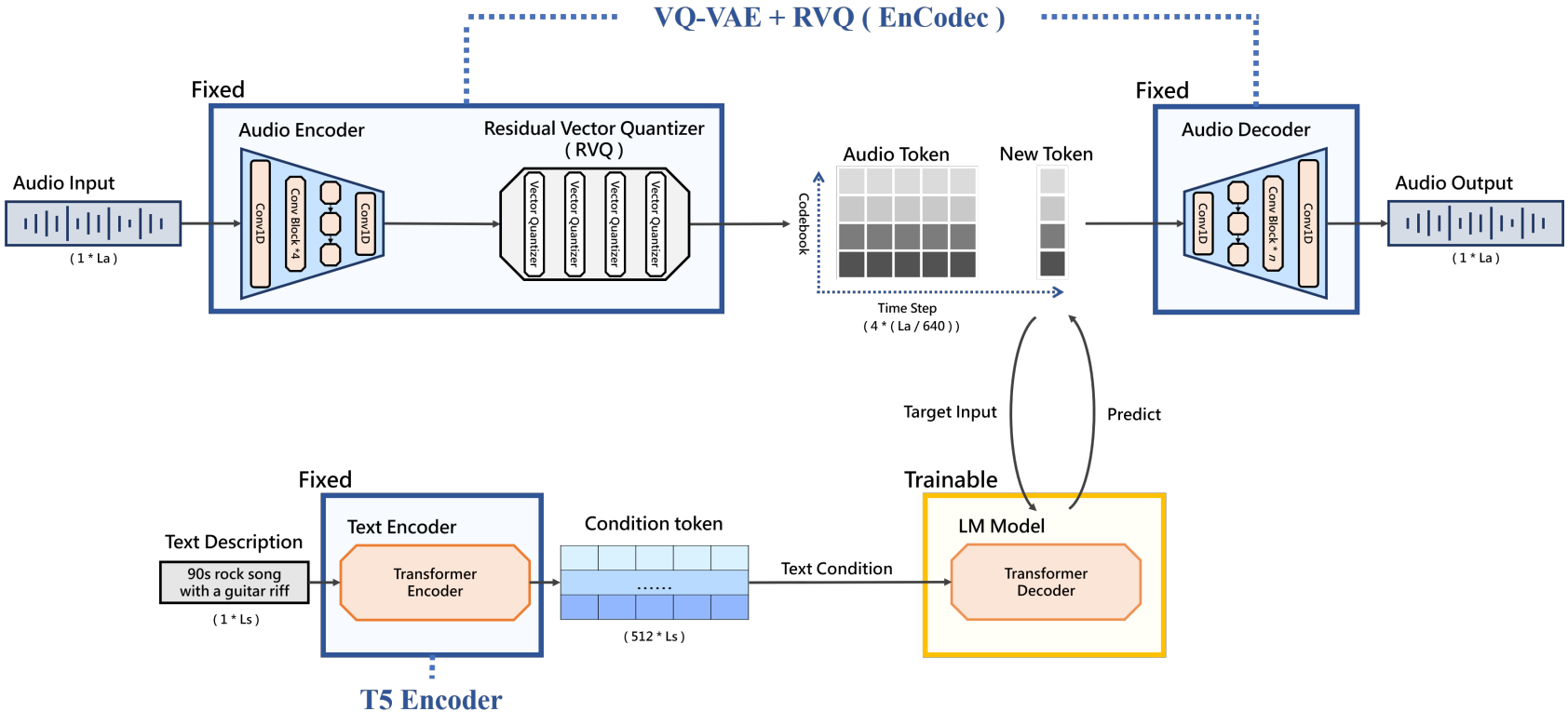
- Midi File
  - .midi
  - Token
  - Volume , Rhythm , Pitch , Timbre
- Audio
  - .wav , .mp3
  - Analog signal to digital signal
  - Quantized waveform

Background - Music Generation

- Symbolic Music Generation
  - Midi File
  - Text to Text
  - Music Transformer
- Audio Generation
  - Audio
  - Text to Text
  - Need compressor (VAE , VQ-VAE , …)
  - MusicGen, MusicLM

Model Architecture

Model Architecture - Overview

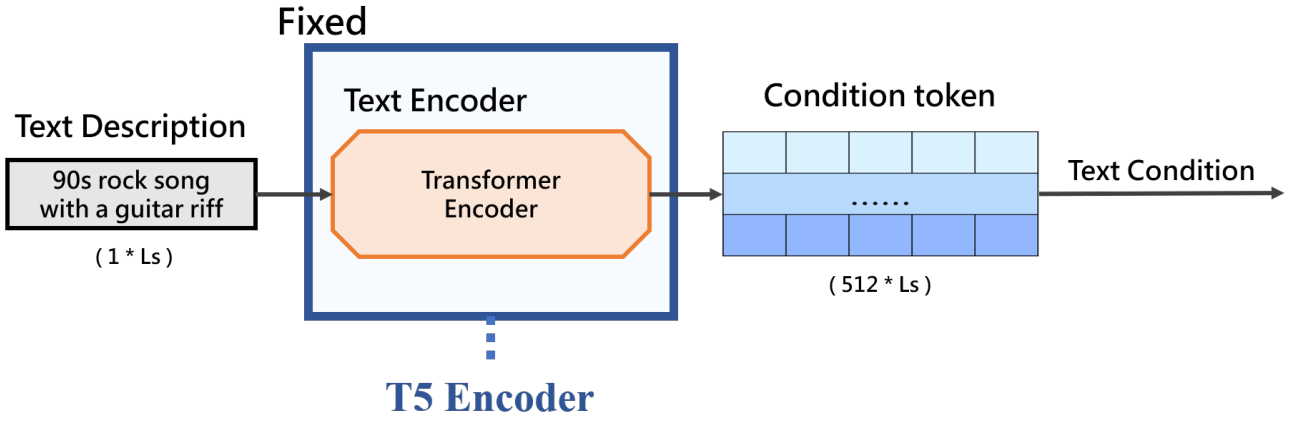


Model Architecture - Overview

- Fixed
  - Text Encoder (T5 Encoder)
  - VQ-VAE + RVQ (EnCodec)
- Trainable
  - LM Model

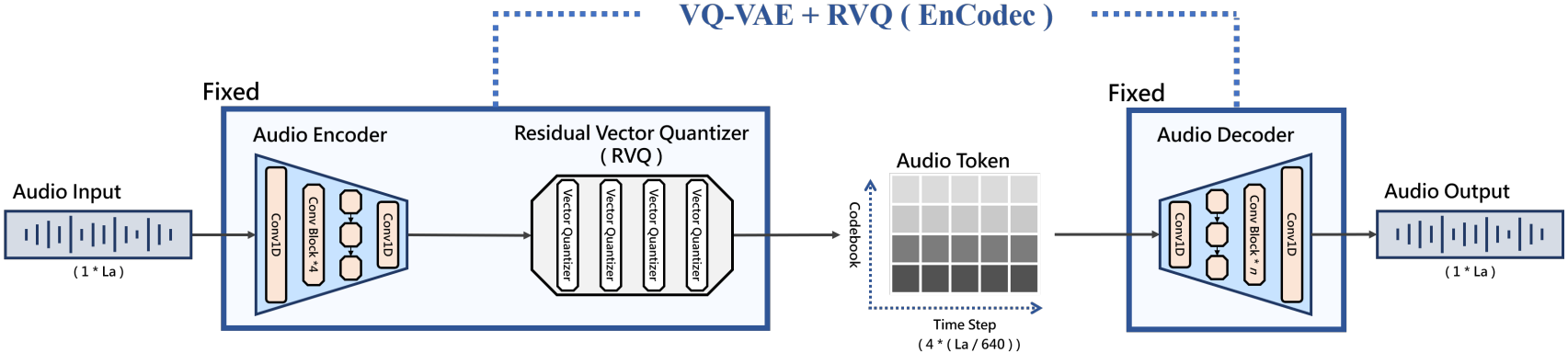
Model Architecture - Text Encoder

- Text Description -> Text Condition
- Transformer Encoder
- Pretrained Model (Fixed)
- Number of Parameter : 50M



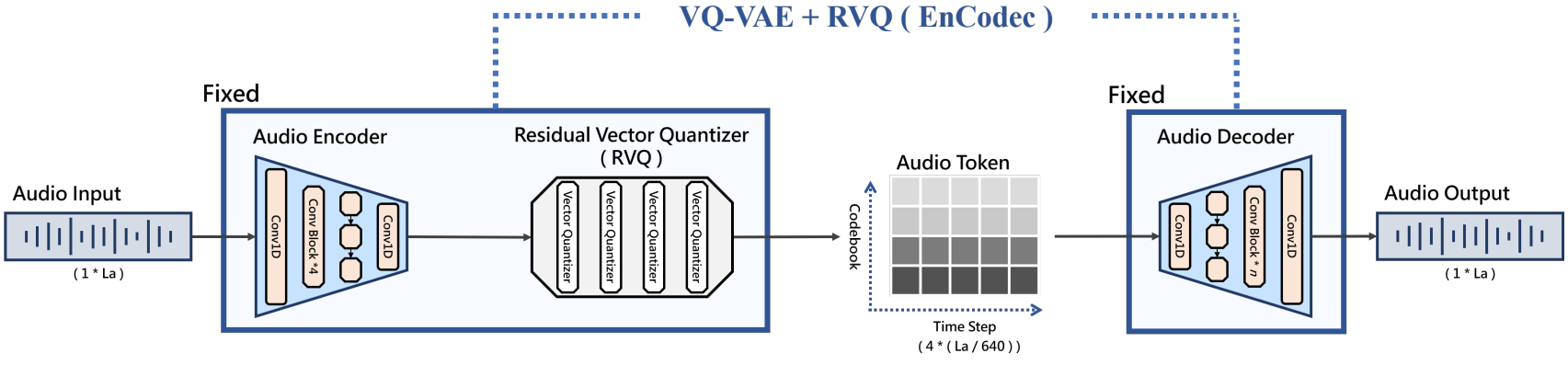
Model Architecture - VQ-VAE + RVQ (EnCodec)

- Audio Encoder、Quantizer
  - Audio Input -> Audio Token
- Audio Decoder
  - Audio Token -> Audio Output



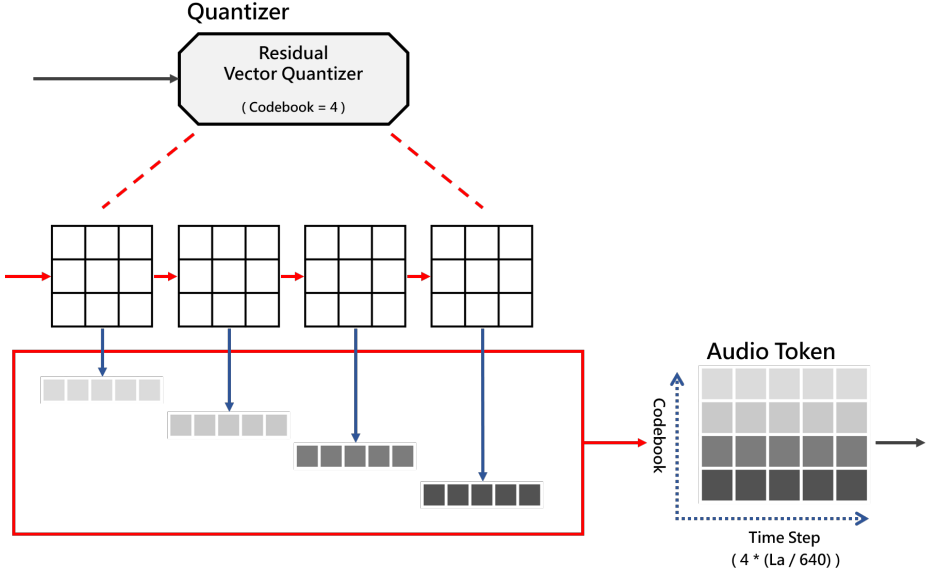
Model Architecture - VQ-VAE + RVQ (EnCodec)

- Pretrained Model (Fixed)
- Audio Input / Output Sampling rate : 32kHz
- Compression Ratio : 640
- Frame rate : 50



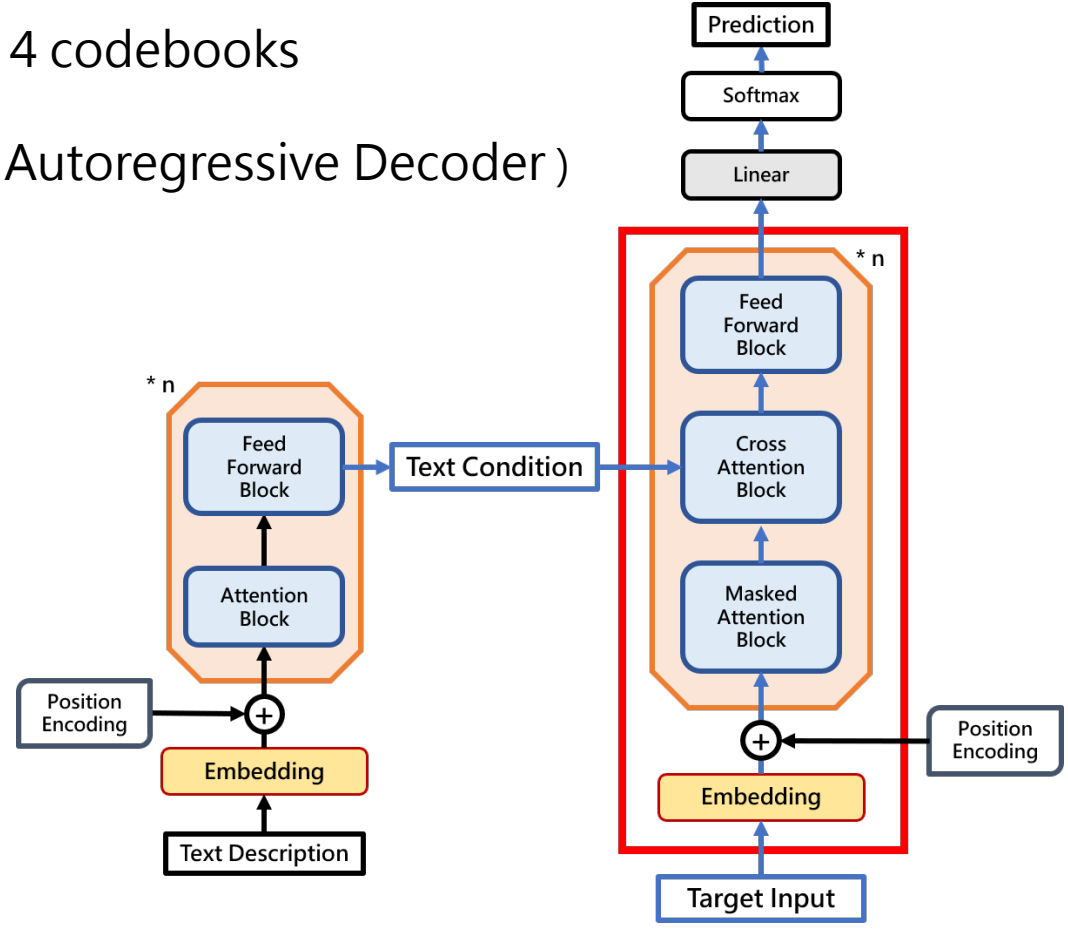
Model Architecture - VQ-VAE + RVQ (EnCodec)

- Quantizer
  - Residual Vector Quantizer (RVQ)
  - 4 Codebooks
  - 2048 Centroids Each Codebook

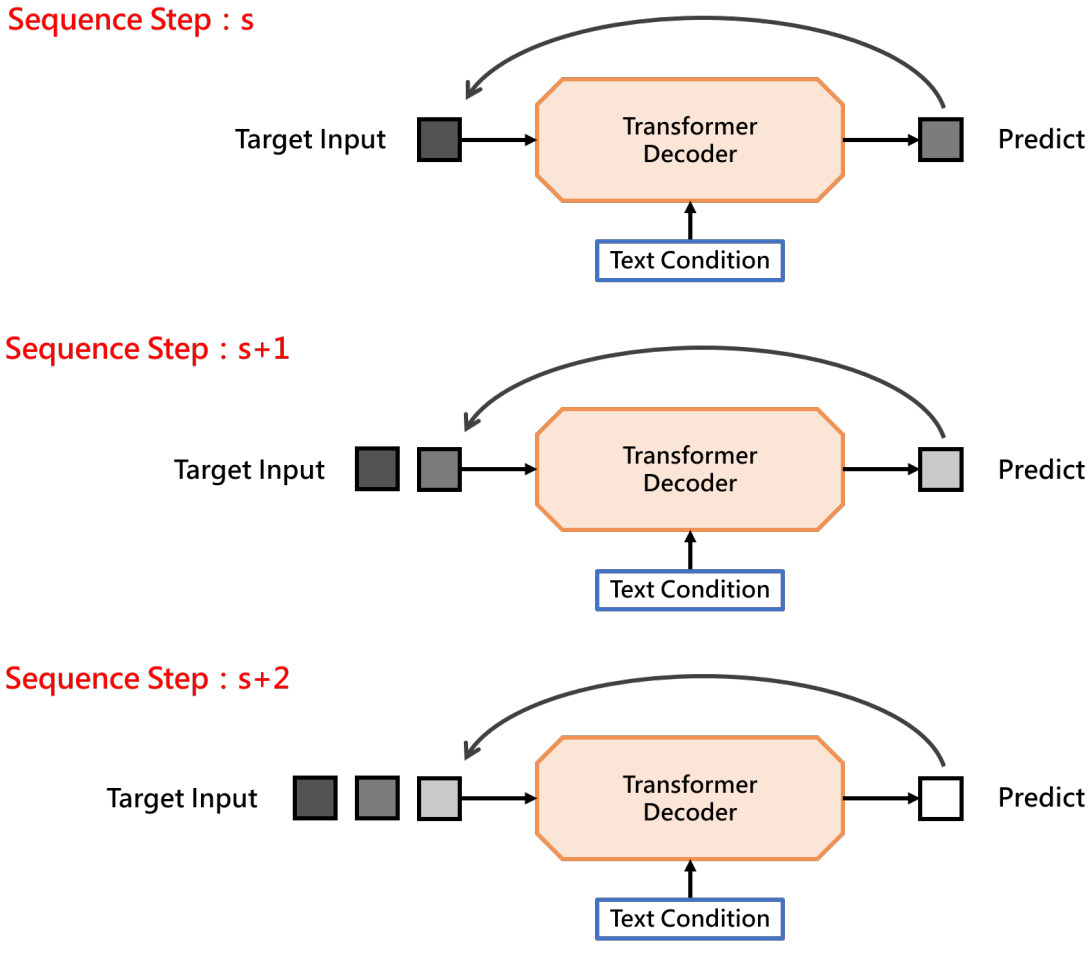


Model Architecture - LM Model

- Predict distributions of 4 codebooks
- Transformer Decoder (Autoregressive Decoder)

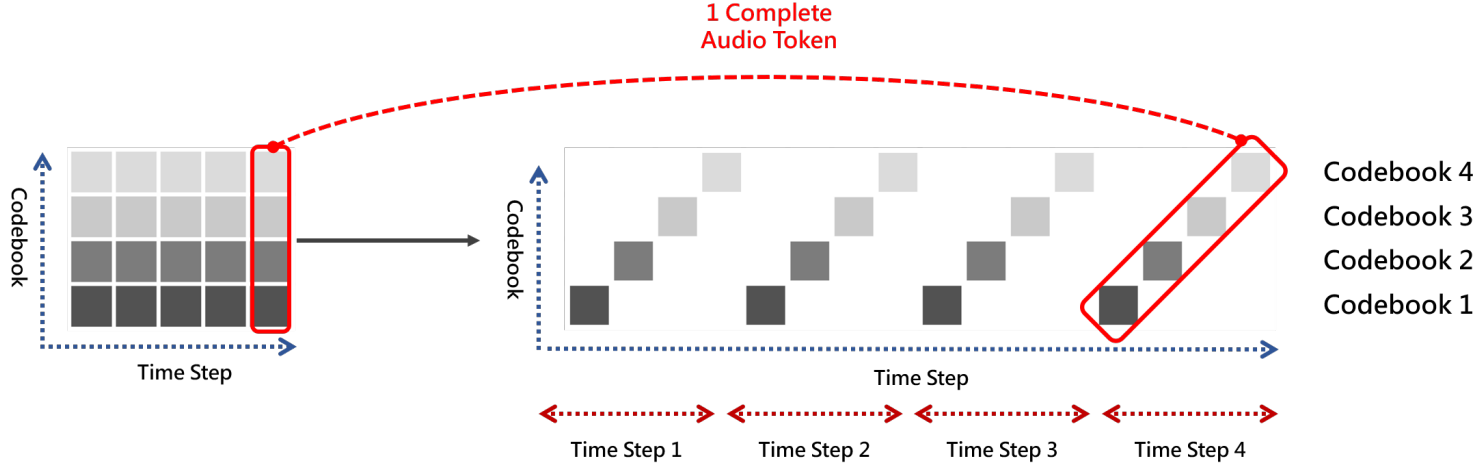


Model Architecture - LM Model



Model Architecture - Flattening Pattern

- Flatten codebooks
- Predict distribution of 1 codebook from 1 audio token one time
- Take 4 sequence steps to predict one complete audio token
- Time - consuming



Model Architecture - Flattening Pattern

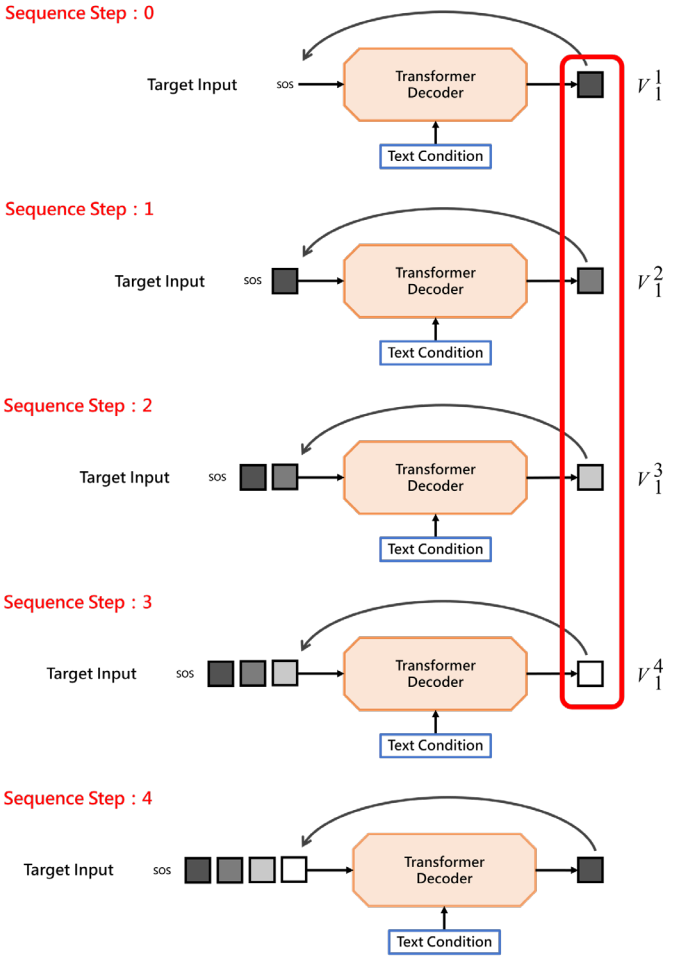
- Predict $\tilde{V}^{(s+1) \bmod 4 + 1}_{(s+1)/4+1}$ at sequence step s
- Target Input : $\tilde{V}$
- Text Condition : U

$$P(V^{s \bmod 4 + 1}_{s/4+1} \mid \text{SOS}, \tilde{V}_{s-1}, U)$$

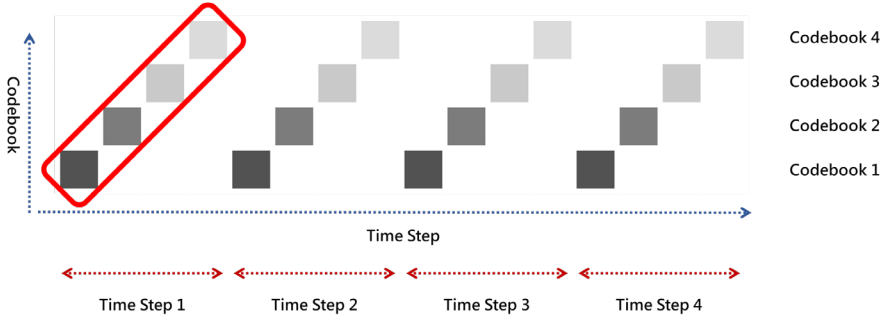
$$\tilde{V}_{s-1} = \{ \tilde{V}_1^{1 \sim 4}, \tilde{V}_2^{1 \sim 4}, \dots, \tilde{V}^{(s-1) \bmod 4 + 1}_{(s-1)/4+1} \}$$

$$U = \{ U_i, 0 < i < Ls \}$$

Model Architecture - Flattening Pattern

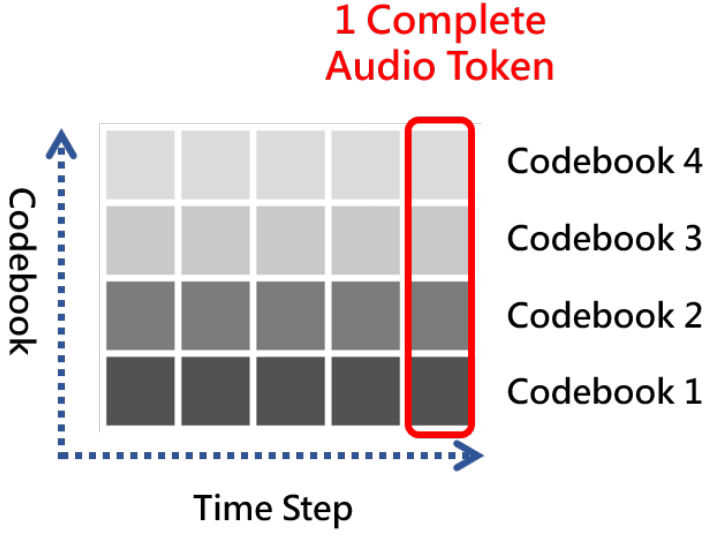


$$\mathrm{P}\left(V_{s/4+1}^{s \bmod 4+1} \mid \mathrm{SOS}, \widetilde{V}_{s-1}, U\right)$$



Model Architecture - Parallel pattern

- Predict distributions of 4 codebooks from 1 audio token one time
- Take 1 sequence step to predict one complete audio token
- Model may be too strong so that performance is not good



Model Architecture - Parallel pattern

- Predict $\widetilde{V}_{s+1} = \{ \widetilde{V}_{s+1}^{1}, \widetilde{V}_{s+1}^{2}, \widetilde{V}_{s+1}^{3}, \widetilde{V}_{s+1}^{4} \}$ at sequence step s
- Target Input : $\widetilde{V}$
- Text Condition : U

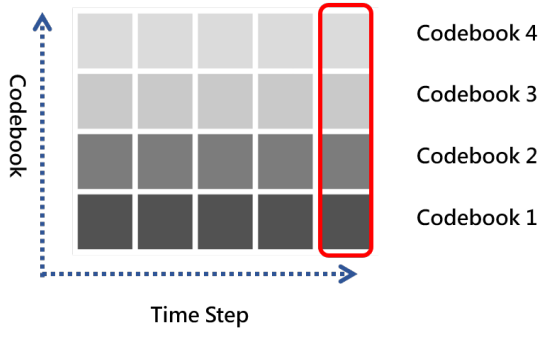
$$\mathrm{P}\left(V_s \mid \mathrm{SOS}, \widetilde{V}_{S-1}, U\right)$$

$$\widetilde{V}_{s-1} = \{ \widetilde{V}_{1}^{1 \sim 4}, \widetilde{V}_{2}^{1 \sim 4}, \ldots, \widetilde{V}_{(s-1)}^{1 \sim 4} \}$$

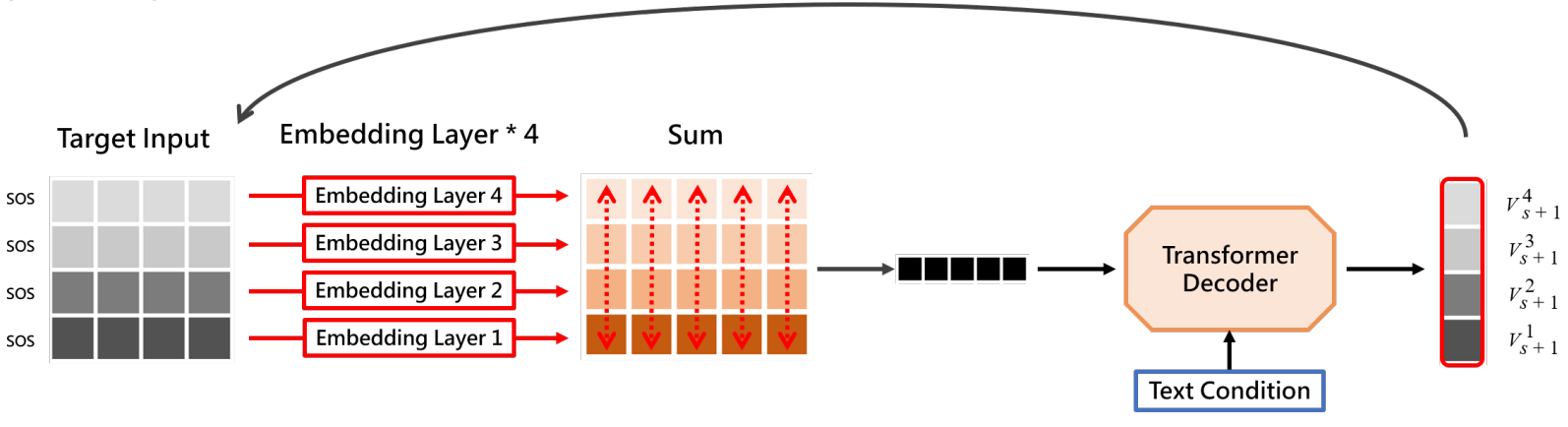
$$U = \{ U_i, 0 < i < Ls \}$$

Model Architecture - Parallel pattern

$$P(V_s \mid \mathrm{SOS}, \tilde{V}_{S-1}, U)$$

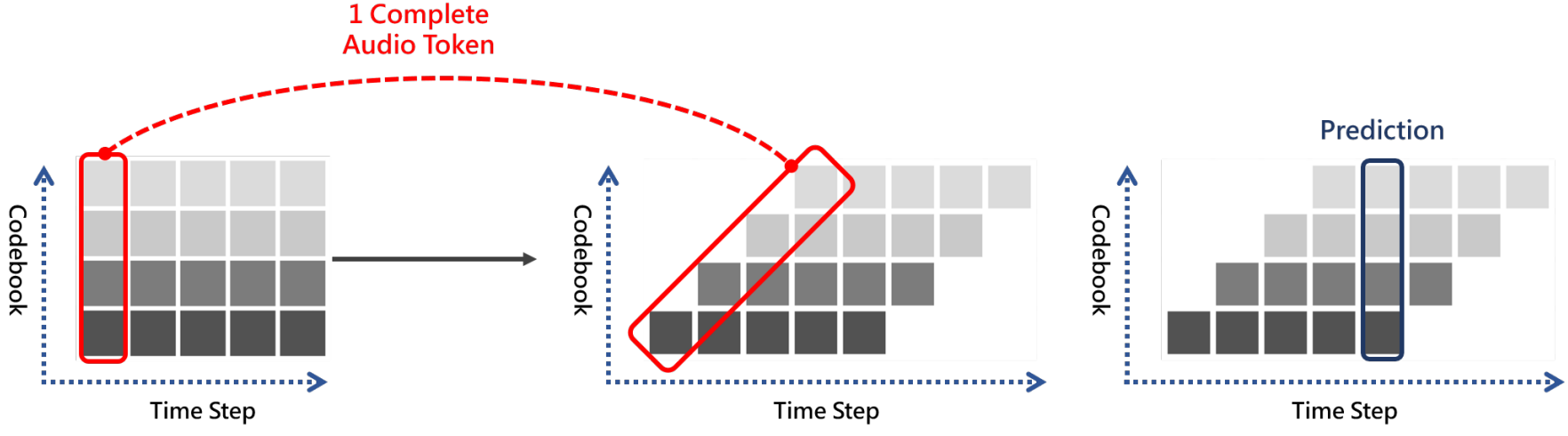


Sequence Step : s



Model Architecture - Delay pattern

- Shift each codebook
- Predict distributions of 4 codebooks from different audio tokens
- Take 4 sequence step to predict one complete audio token



Model Architecture - Delay pattern

- Predict $\widetilde{C}_{s+1} = \{ \widetilde{V}^{c}_{(s+1)-c+1}, c = 1 \sim 4 \}$ at sequence step s
- Target Input : $\widetilde{C}$
- Text Condition : U

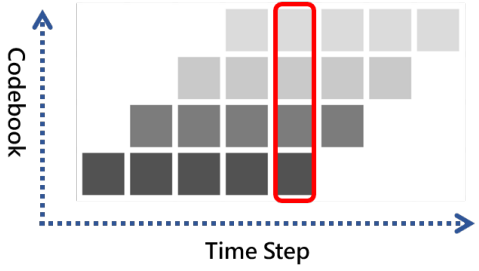
$$\mathrm{P}\left(C_s \mid \mathrm{SOS}, \widetilde{C}_{s-1}, U\right)$$

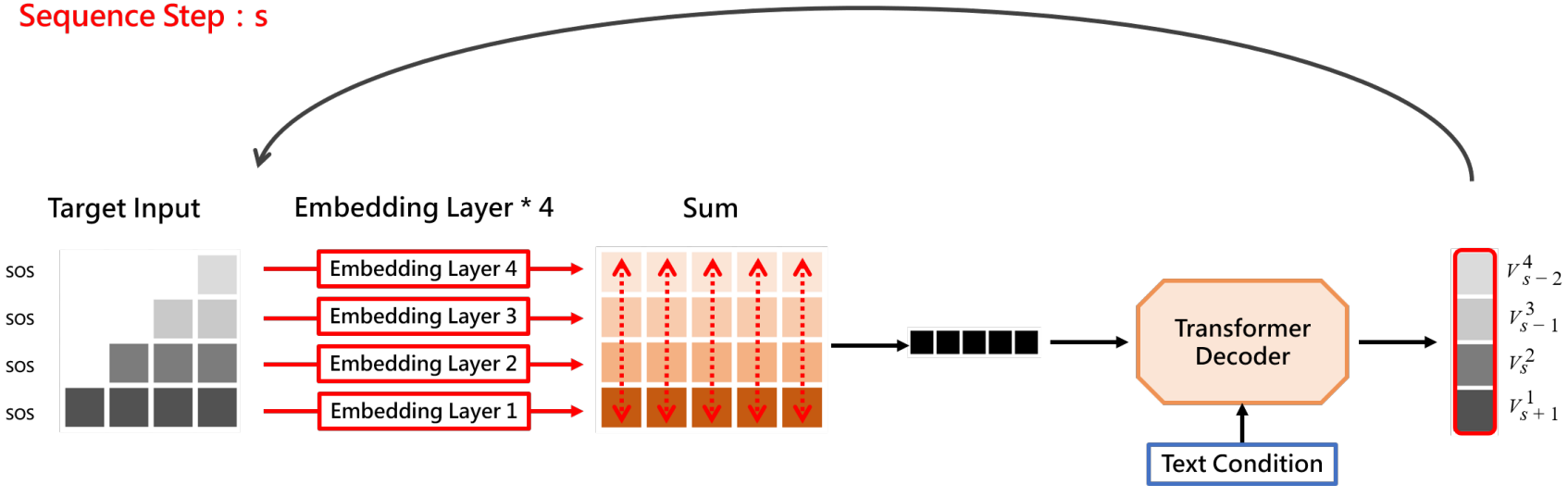
$$\widetilde{C}_{(s-1)} = \{ \widetilde{V}_1^{1\sim4}, \widetilde{V}_2^{1\sim4}, \ldots, \widetilde{V}_{s-4}^{1\sim4}, \widetilde{V}_{s-3}^{1\sim3}, \widetilde{V}_{s-2}^{1\sim2}, \widetilde{V}_{s-1}^{1} \}$$

$$U = \{ U_i, 0 < i < Ls \}$$

Model Architecture - Delay pattern

$$P\left(C_S \mid \mathrm{SOS}, \widetilde{C}_{S-1}, U\right)$$





Model Architecture - Loss Function

- Evaluate cross entropy with
  - Predicted Distributions $P_{c=1} \sim P_{c=4}$
  - Ground Truth $\widetilde{C}_{s+1} = \{ \widetilde{V}^{c}_{(s+1)-c+1}, c = 1 \sim 4 \}$

$$\text{Loss} = -\frac{1}{4} \log \left(P_{c=1}(\widetilde{V}^{1}_{s+1}) + P_{c=2}(\widetilde{V}^{2}_{s}) + P_{c=3}(\widetilde{V}^{3}_{s-1}) + P_{c=4}(\widetilde{V}^{4}_{s-2}) \right)$$

Dataset & Training

Dataset & Training - Dataset

- MusicCap
  - 5419 Data
  - 48kHz stereo .wav file
  - Text Description

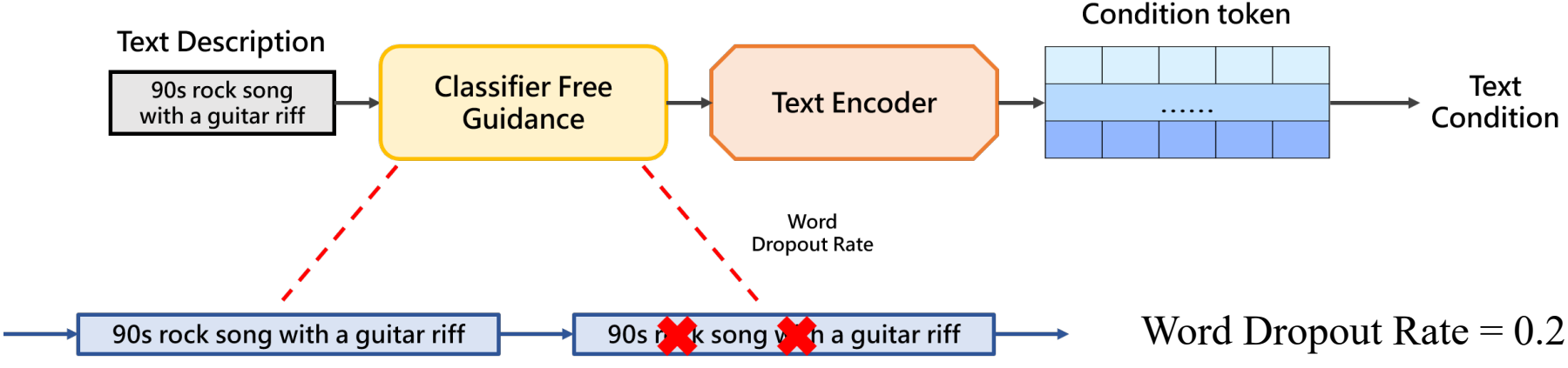
Dataset & Training - Data Splitting

- Non – Filtered Data
  - Training Set : 3522 (65 %)
  - Validation Set : 813 (15 %)
  - Test Set : 1084 (20 %)

- Filtered Data
  - Training Set : 1343
  - Validation Set : 302
  - Test Set : 360

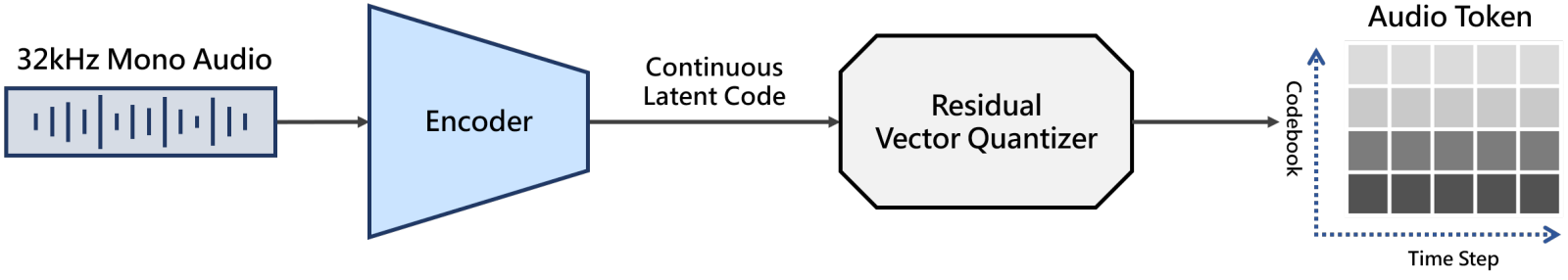
Dataset & Training - Data Preprocessing

- Text Description
  - Drop words with word dropout rate 0.2
  - Take first 250 words
  - Convert to text condition



Dataset & Training - Data Preprocessing

- Audio
  - Convert to 32kHz mono audio
  - Convert to audio token



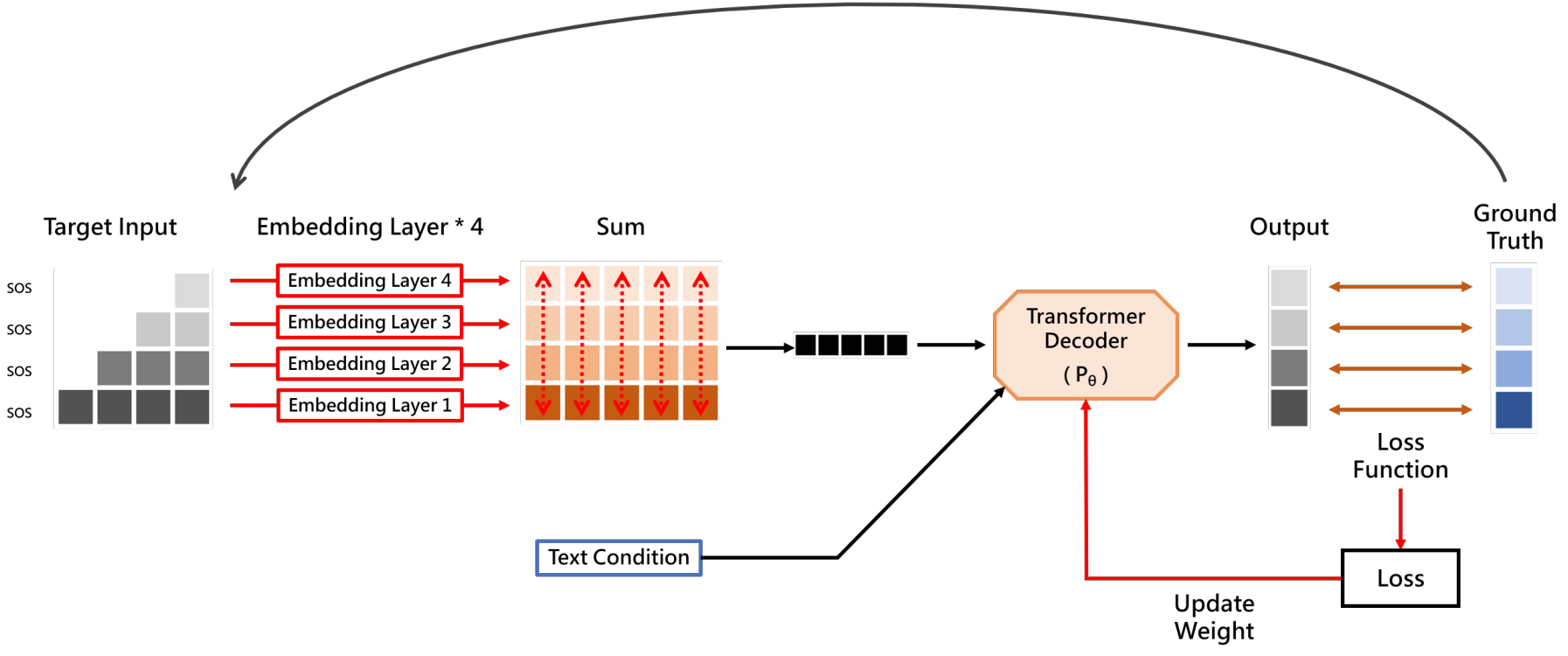
Dataset & Training - Data Preprocessing

- Audio token
  - Training Set : Ground Truth
  - Validation Set / Test Set : Evaluate performance metric
- Set batch size with 30

Dataset & Training - Training

- Take first n data from each batch
- Evaluate loss with loss function every batch
- Update weights P_θ every 20 batch (Gradient Accumulation)

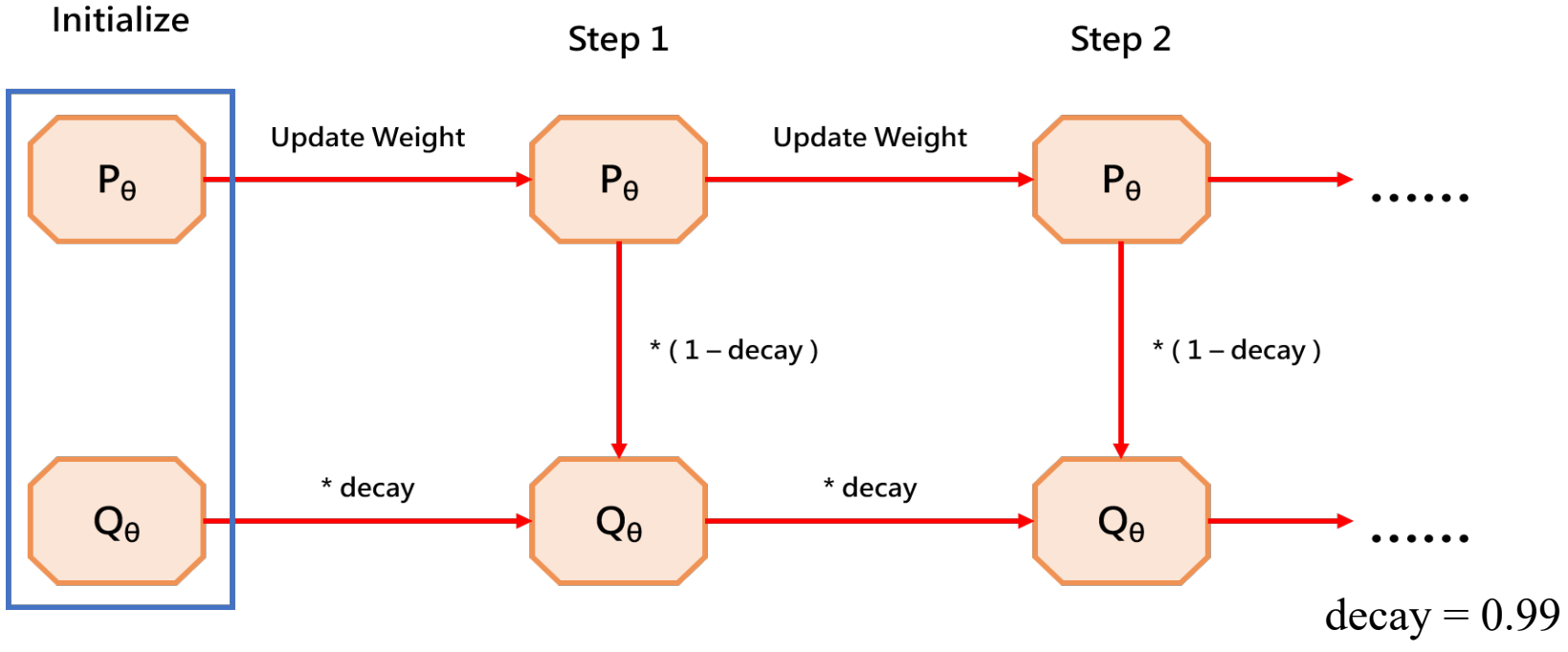
Dataset & Training - Training



Dataset & Training - Training

- Use exponential moving average with decay 0.99 to update weights Q_θ

$$Q_\theta = Q_\theta * \text{decay} + P_\theta * (1 - \text{decay})$$



EXPERIMENT

Experiment - Inference

- Load weights Q_θ
- Use classifier free guidance to inference
  - Predict $P_{conditional}$ with target input and text condition
  - Predict $P_{unconditional}$ with target input
  - Get P_{new} with linear combination

$$P_{new} = \gamma * P_{conditional} + (1 - \gamma) * P_{unconditional}$$

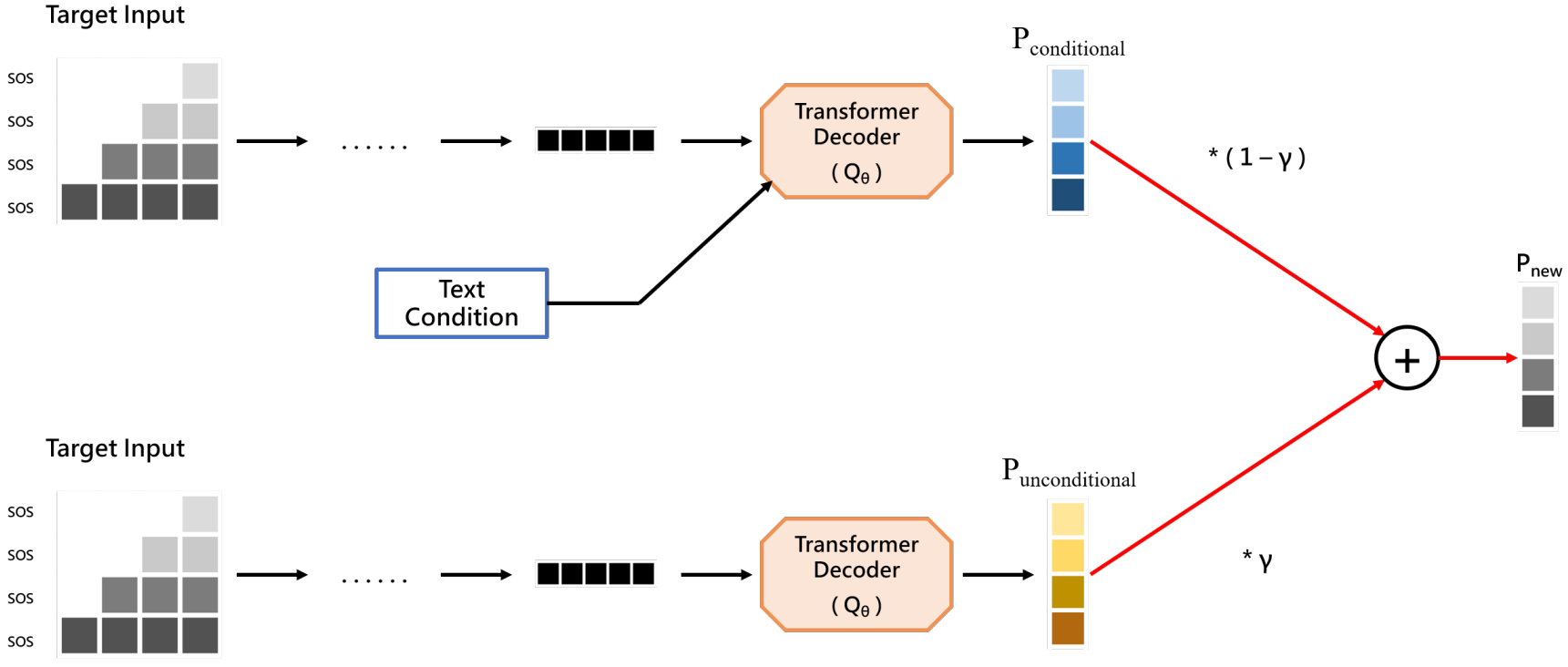
$$P_{conditional} = \mathrm{P}\left(C_S + 1 \mid \mathrm{SOS},\ C_S, U\right)$$

$$P_{unconditional} = \mathrm{P}\left(C_S + 1 \mid \mathrm{SOS},\ C_S\right)$$

$$\gamma = 0.2$$

Experiment - Inference

$$P_{new} = \gamma * P_{conditional} + (1 - \gamma) * P_{unconditional} \qquad \gamma = 0.2$$



Experiment - Inference

- Sample P_{new} with
  - Top-k , k = 250
  - Temperature = 1.0

Experiment - Model and Hyperparameter

MusicGen-1

- d_model : 1024
- nheads : 8
- nlayer : 8
- d_hid : 2048

- Total Parameter : 119,171,072

MusicGen-2

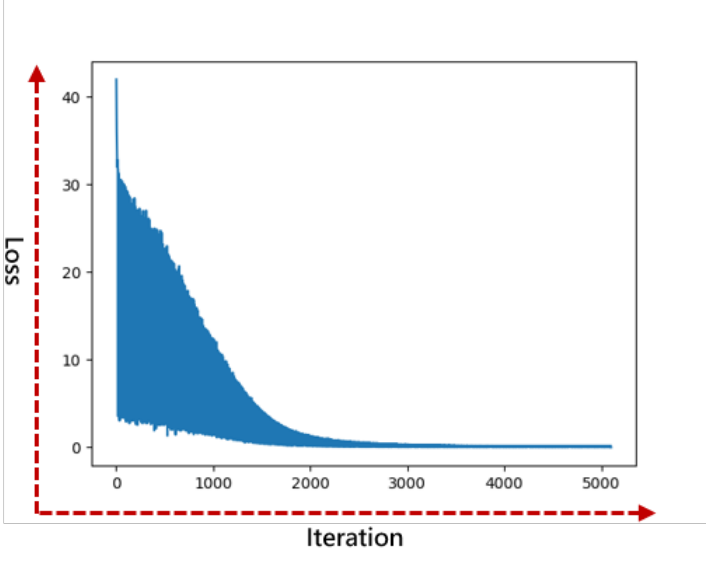
- d_model : 2048
- nheads : 16
- nlayer : 16
- d_hid : 4096

- Total Parameter : 844,689,408

Experiment - Training Result

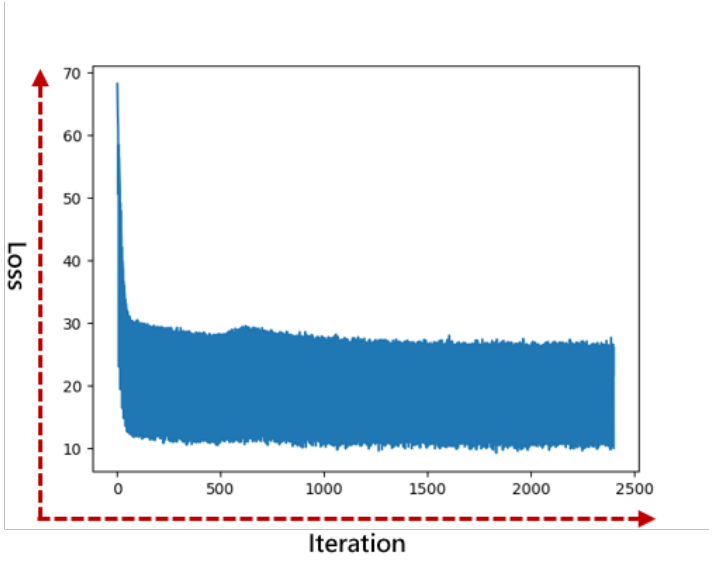
MusicGen-1

- Non-Filtered Data 300 Data



MusicGen-2

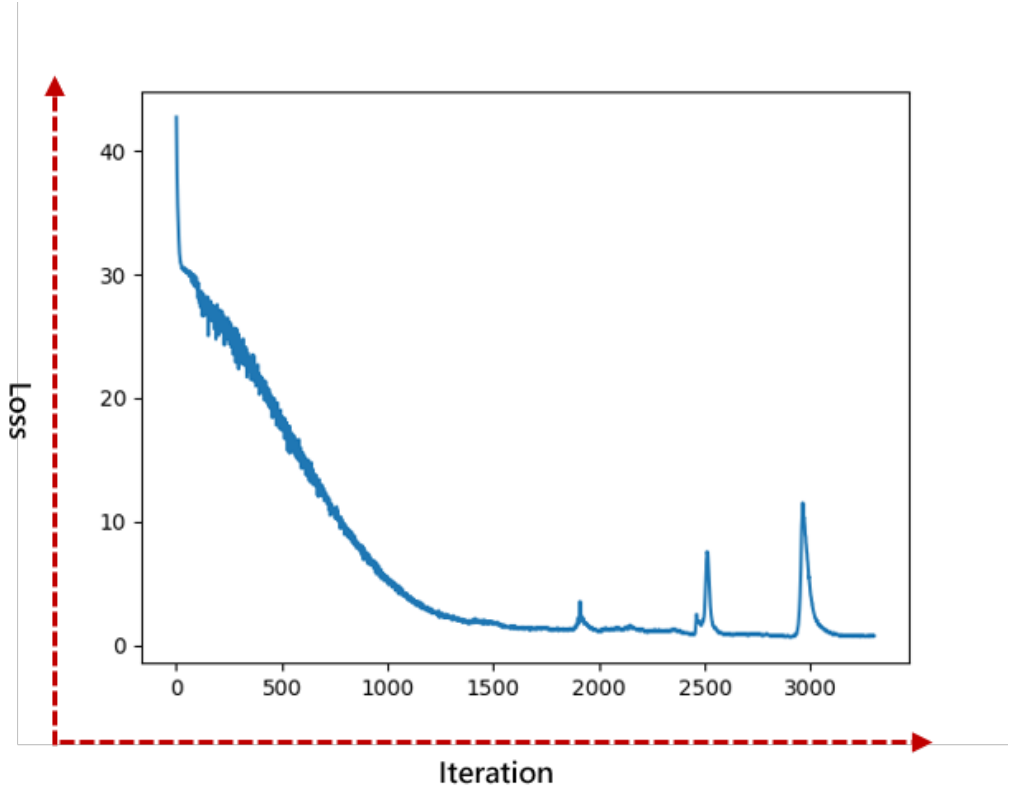
- Non-Filtered Data 600 Data



Experiment - Training Result

MusicGen-1

- Filtered Data 300 Data

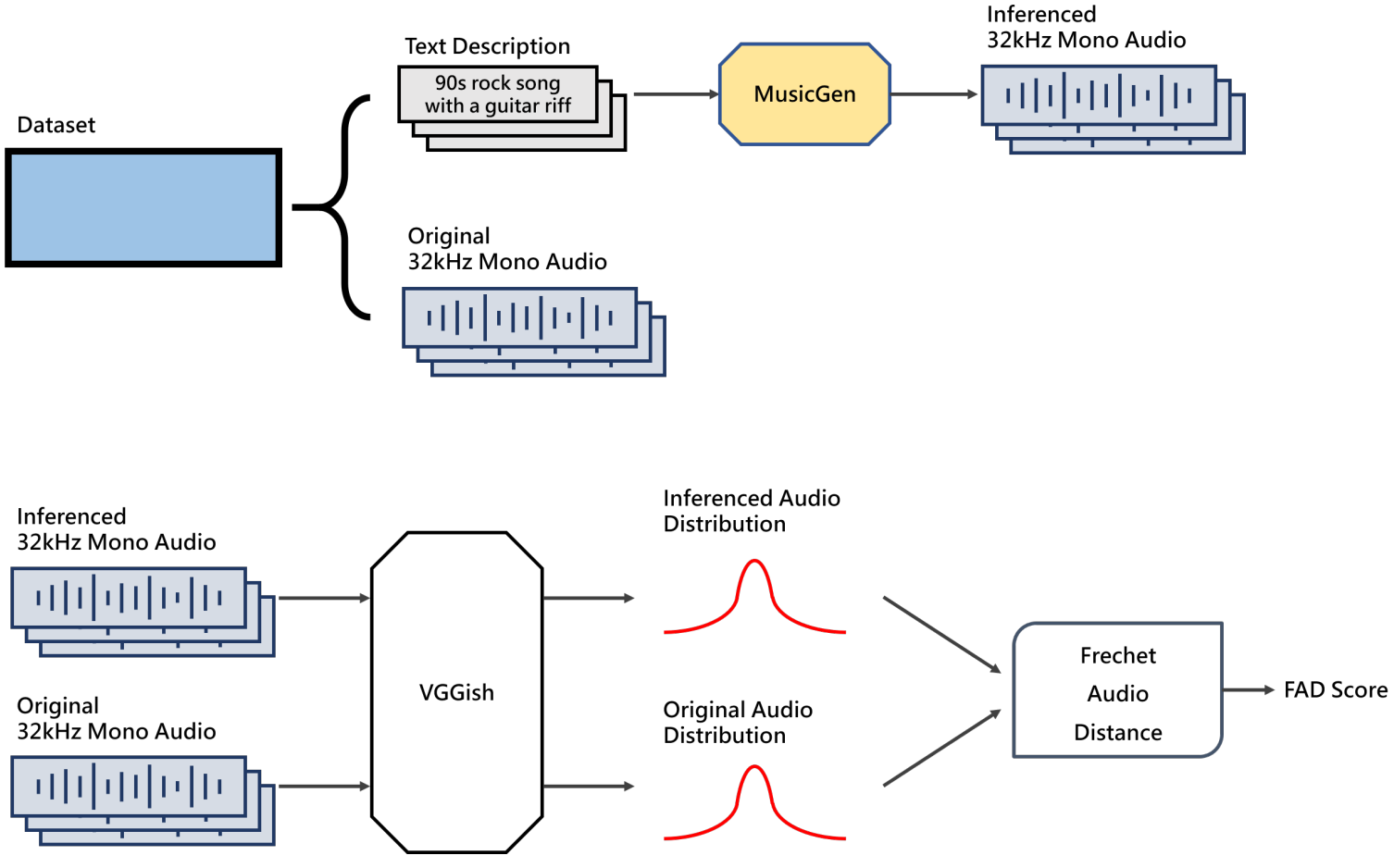


Result

Result - Performance Metric

- FAD score
  - Use VGGish to compute distributions of original and inferenced audio
  - Evaluate frechet-audio-distance between 2 distributions

Result - Performance Metric



Result - Melody Evaluation

MusicGen-1

- Validation Set
  - Non-Filtered Data 300 Data : 7.70155
  - Filtered Data 300 Data : 11.05481
- Test Set
  - Filtered Data 300 Data : 10.82677

Result - Demo Music

- Demo 1 : 

The low quality recording features a ***drum & bass song*** that consists of a repetitive female vocal sound effect, ***growl synth bass, manic percussions, shimmering hi hats, punchy kick and snare hits and echoing synth pad***. It sounds energetic, aggressive and manic - like something you would hear in underground clubs during the 00s.

- Demo 2 : 

This is an advertisement ***jingle music piece***. It is an ***instrumental piece***. The main theme is being played by the piano while ***there is a synth string sound in the melodic background***. There is an emotional, heart-touching atmosphere. ***This piece could be used in the soundtrack of a drama movie during scenes of tragedy***. It could also work well as an advertisement jingle where there is an attempted appeal to emotion.

- Demo 3 : 

The low quality recording features a live performance that consists of bagpipes melodies. ***It sounds soulful, passionate, cultural and the recording is in mono and noisy***.

Result - Demo Music

- Demo 4 : 

  It sounds energetic, aggressive and manic - like something you would hear in underground clubs during the 00s.

- Demo 5 : 

  It sounds soulful, passionate, cultural and the recording is in mono and ***noisy***.