

My 2024 Research & Work Experience Review


Hsiao Wen Yi 蕭文逸



AI Labs.tw
台灣人工智慧實驗室

Research Experience

 > **1K cites** (9 papers, 1 Journal)

 > **1K stars** (from 8 repositories) ID: **wayne391**

5 Yrs Working Exp, 7 Yrs in Music Research

'12

B.Sc.

Computer
Science

@Tsing Hua Uni.



'16

M.Sc.

Computer
Science

@Tsing Hua Uni.



'18

**Research
Assistant**

@Academia Sinica



'19

**Research
Engineer**

@Taiwan Allabs



AI Labs.tw

台灣人工智慧實驗室

'24

**Research
Engineer
(Senior)**

**Master Thesis: Automatic Symbolic Music Generation Based on Convolutional GANs. (2018),
Adviser: Dr. Yi-Hsuan Yang.**



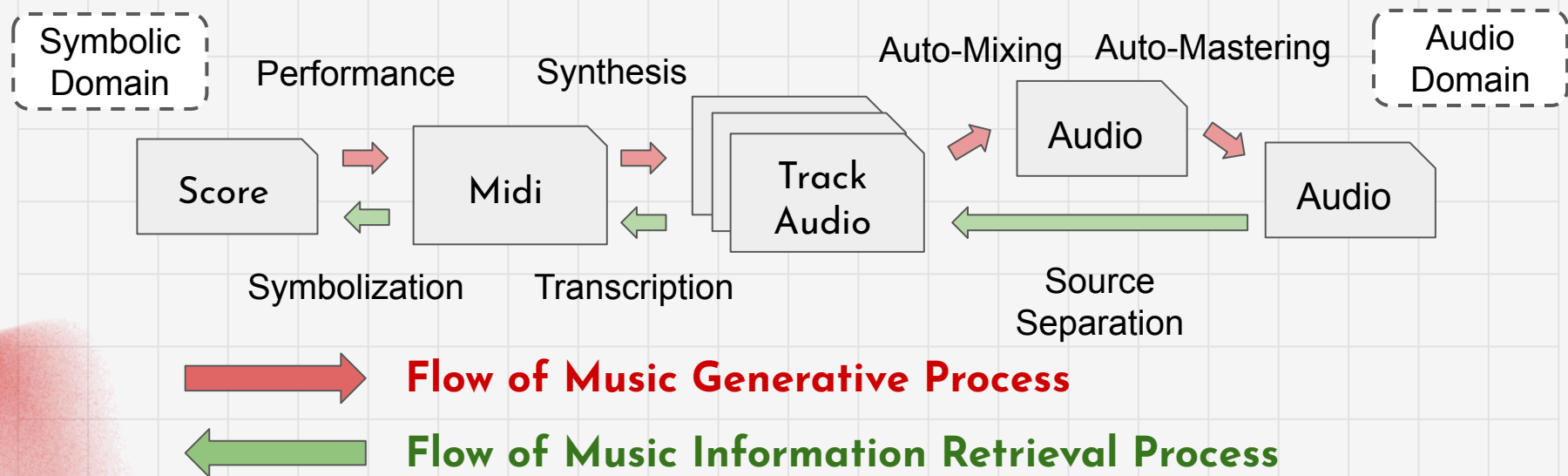
01

Work Overview

7-min Research Introduction

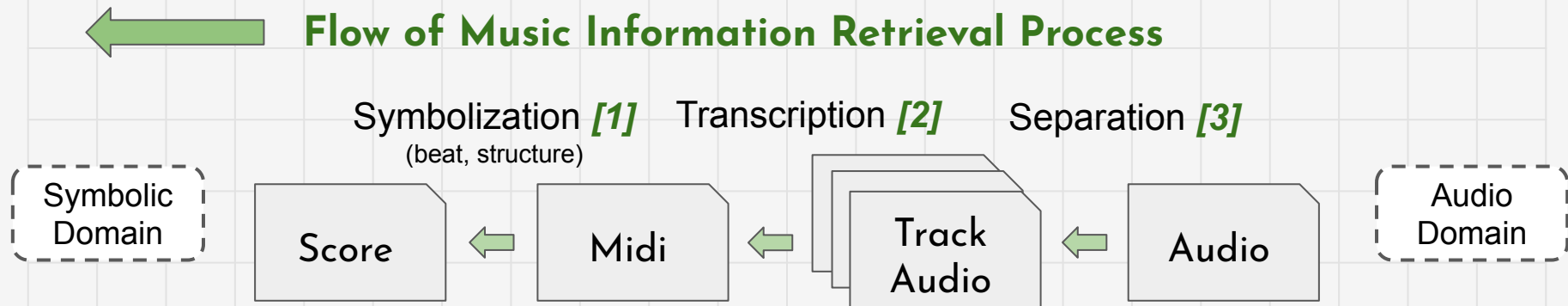
My Music Research Overview

- I have a **comprehensive experience** with the entire **Music Research pipeline**.
- I have strong knowledge of modern **music production industry**.
- I have **cross-domain** (score, text and audio) modeling experience.
- In what following, I will **prove my skill** by **publications** and **github repos**



My Music Research Overview

Flow of Music Information Retrieval Process



(ISMIR'22, 2nd author) Transcription of Polyphonic Electric Guitar Music [2]

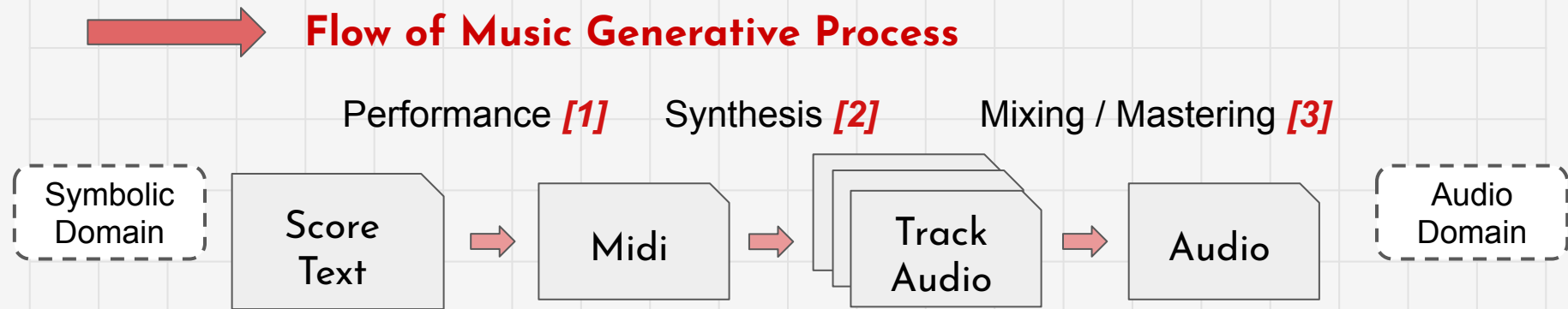
(Eusipco'21, 3rd author) Beat and Downbeat Tracking Enhanced with Source Separation [1]

(MMSP'20, 2nd author) Blind Violin/Piano Source Separation with Mixing-specific Data Augmentation [3]

- MidiToolkit [1] (227 stars) - Popular and Fundamental Tool for MIDI Processing
 - Conversion between absolute & symbolic timing
- SF Segmenter [1] (52 stars) - Structure Analysis with Structural Feature

My Music Research Overview

Flow of Music Generative Process



(ISMIR'24, 2nd author) **MusiConGen** - Text-to-Music Audio Generation with Chord and BPM Control [1,2,3]

(DAFX'24, 2nd author) **Hyper RNN for AFx Modeling** Oral [3]

(ISMIR'22, co-1st author) **DDSP-based Singing Vocoders** - Differentiable DSP Singing Vocoder [2]

(AAAI'21, 1st author) **Compound word transformer** - Symbolic Music Generation for piano [1,2]

(ISMIR'21, 3rd author) **Guitar Tabs by Transformers and Groove Modeling** [1,2]

(AAAI'18, co-1st author) **Musegan** - Symbolic Music Generation Oral [1]

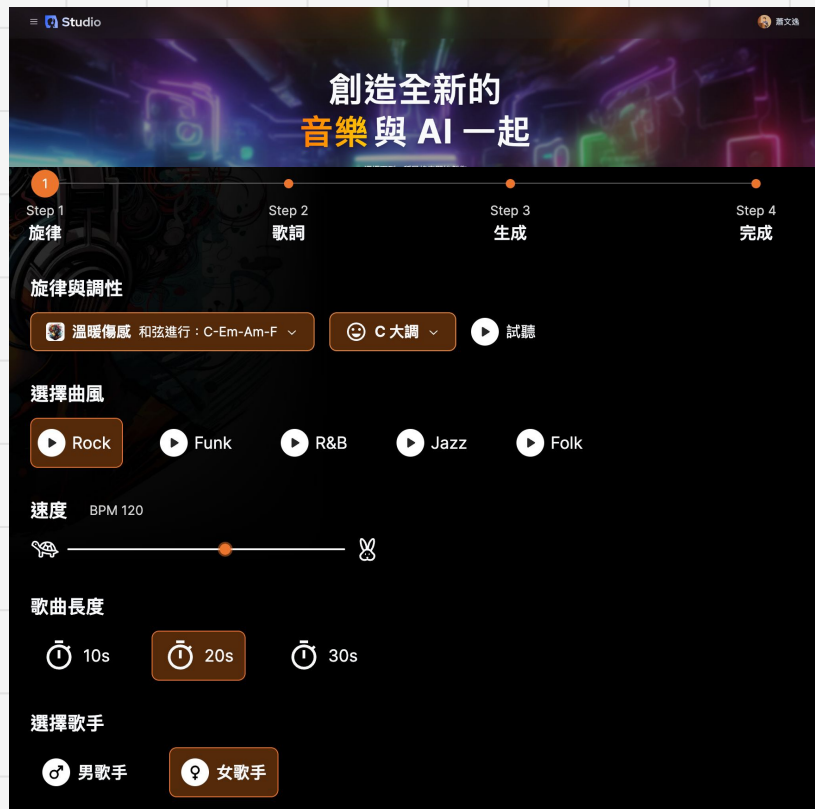
Production

教我如何做你的愛人 - 陳珊妮 AI 模型

Collaboration with a Popular Chinese Singer



AI Singer + AI Song Maker

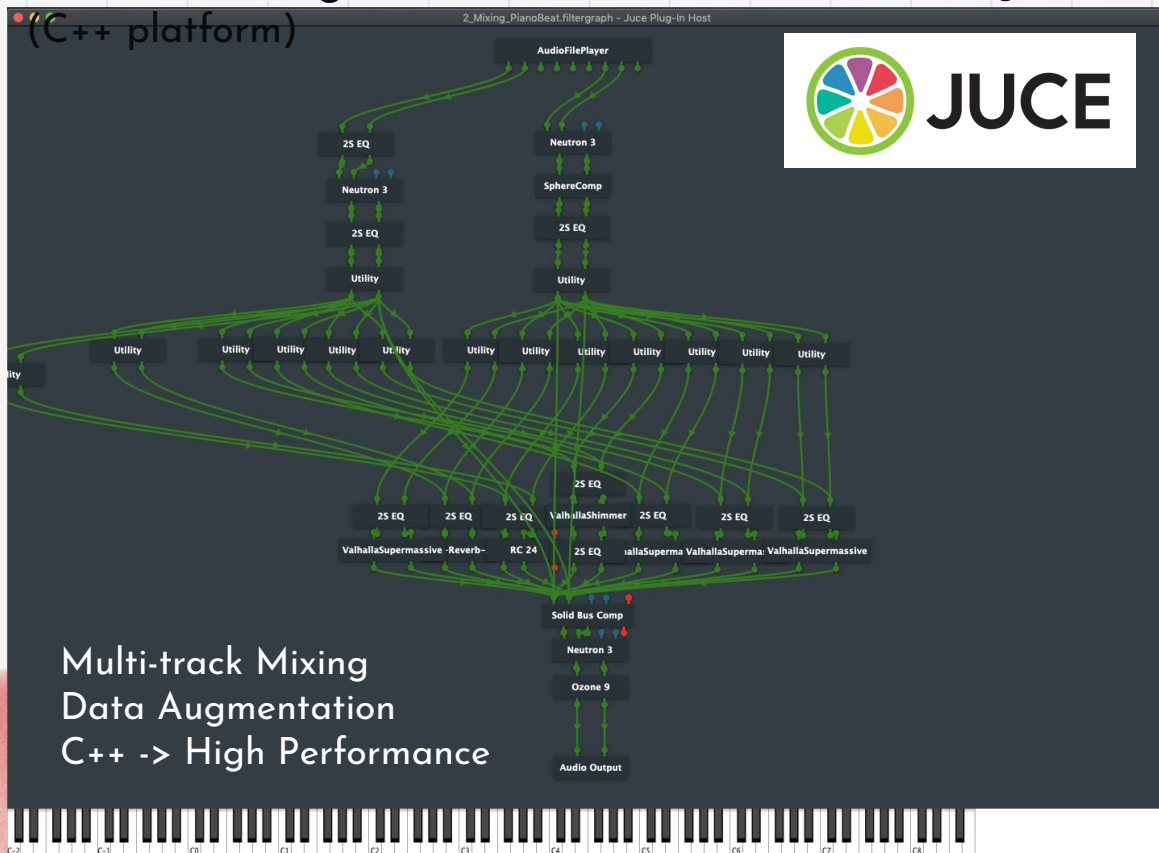


Yating Music - Song Creation Platform

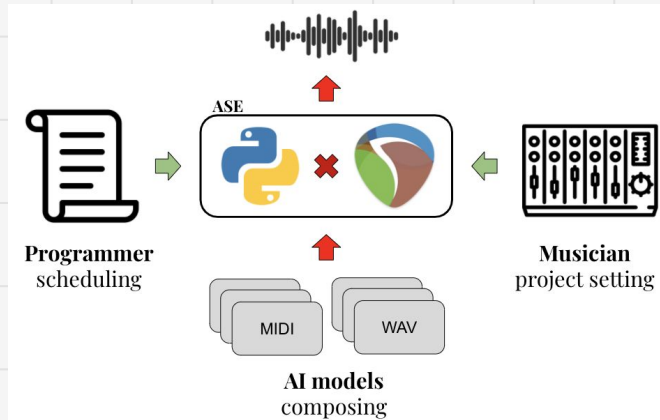
Inhouse Mixing Backend, Based on JUCE C++ Plugin Host

Inhouse Mixing Backend, Based on JUCE C++ Plugin Host

```
(C++ platform)
```



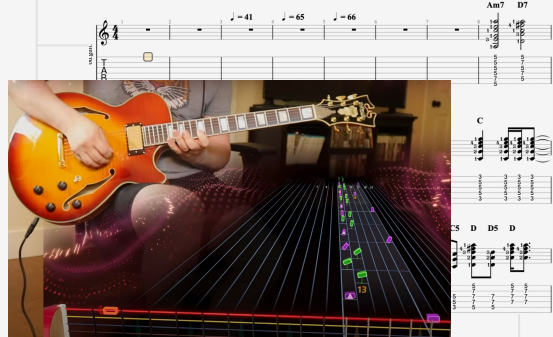
- OpenSource:
 - ReaRender (94 stars)



Dataset Building

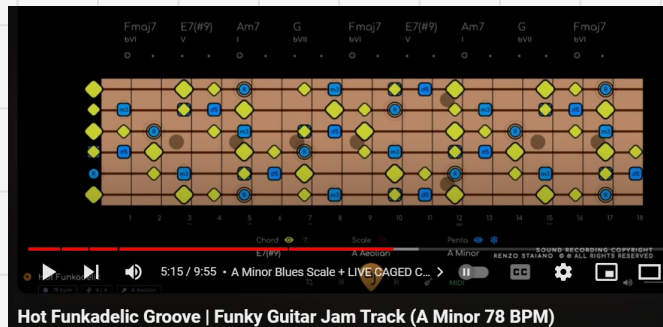
Highlights of My Inhouse Collection:

1. Data from Guitar Gaming Community



- Aligned audio and tab
- Finger position
- Chord label
- Over 1K songs
- Multi-track guitar
- Tab Generation
- Transcription

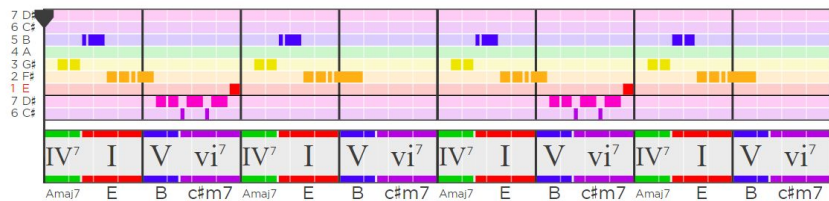
3. Backing Tracks



Skillset:

- Web Crawling, Data Cleaning
- Musicology

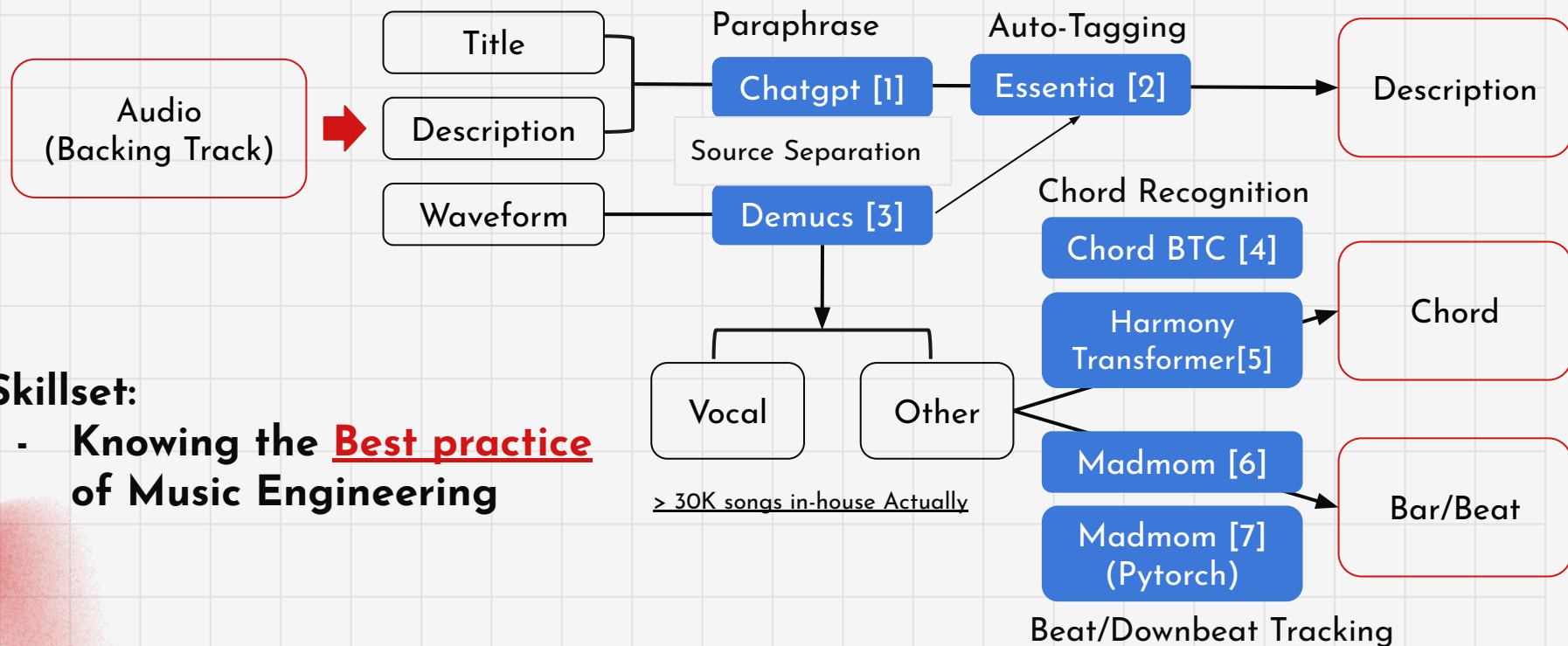
2. Lead Sheet from theorytab (108 stars)



- Over 30k songs
- Our backbone dataset of text2music model
- Description
- Key
- BPM
- Chord Progression
- High Quality after Curation (TODO)
-> Excellent Resources for any task!

Dataset Building

- Pipeline from my work - MusicConGen (ISMIR'24)



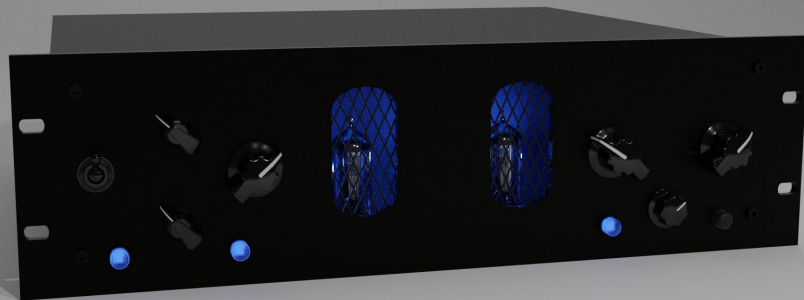
Side Projects

Audio Effect Emulation with AI & Make EQ/Distortion Plugin with JUCE

- TorchLite Demo (5 stars)
- Similar Product:
 - Neural DSP, Positive Grid, ...



Mixing Gear (vacuum tube) Emulation



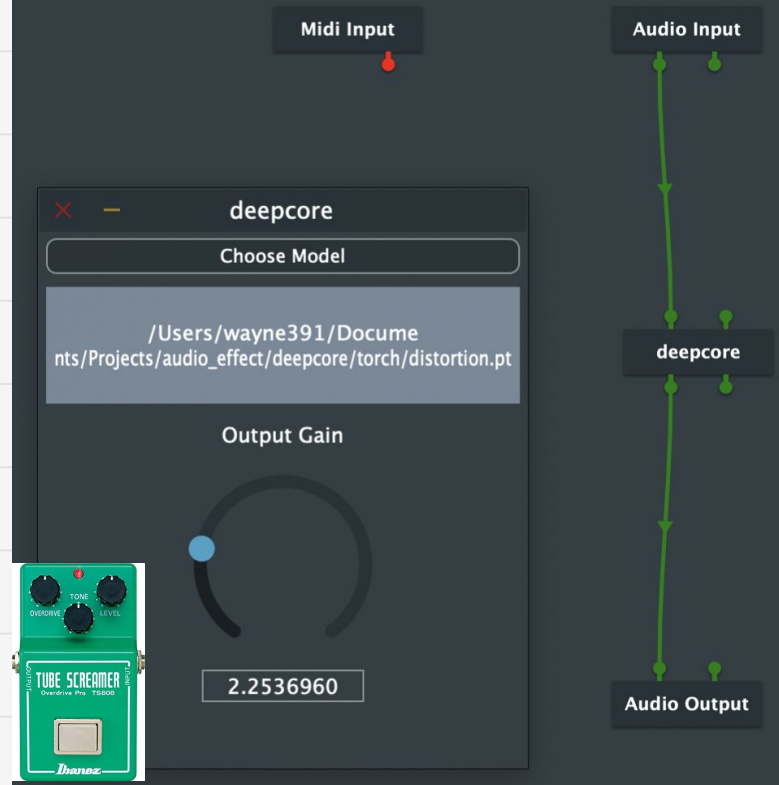
My 3D Modeling Artwork :D

(DAFX'24, 2nd author) Hyper RNN for AFx Modeling

Skillset:

- Train DSP-inspired NN Models
- Deploy with C++ (Libtorch + Eigen)

TS-808 Pedal Real-time Emulation



Visibility

Build Open-source Ecosystem of Our Company



Yating Music, Taiwan AI Labs

A research team working on Music AI technology at the

86 followers

Taipei, Taiwan

<https://ailabs.tw>

Popular repositories

remi

Public

"Pop Music Transformer: Beat-based Modeling and Generation of Expressive Pop Piano Compositions", ACM Multimedia 2020

Python 540 84

ddsp-singing-vocoders

Public

Official implementation of SawSing (ISMIR'22)

Python 249 35

MuseMorphose

Public

PyTorch implementation of MuseMorphose (published at IEEE/ACM TASLP), a Transformer-based model for music style transfer.

Python 170 32

compound-word-transformer

Public

Official implementation of compound word transformer (AAAI'21)

Python 265 43

miditoolkit

Public

<https://pypi.org/project/miditoolkit/>

Python 227 35

ReaRender

Public

A python toolkit for automatic audio/MIDI rendering using REAPER

Python 94 16



Wen-Yi Hsiao

ailabs

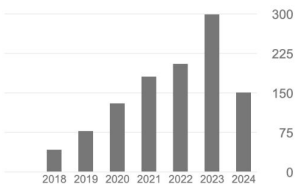
在 ailabs.tw 的電子郵件地址已通過驗證

[Machine Learning](#)

追蹤

引用次數

	全部	自 2019 年
引文	1097	1048
H 指數	10	10
i10 指數	10	10



當AI遇見數位音樂－創作工具的介紹與應用

27 9 月, 2023

當AI
遇見數位音樂：
創作工具
的介紹與應用

2 @ 112A — NYCU
3 跨域系列講座

Product Promotion -
Campus Workshop @NYCU



02

Audio To Symbolic Domain

Music Information Retrieval (MIR)

Audio to Symbolic Domain

What is the Symbolic Domain in Music?

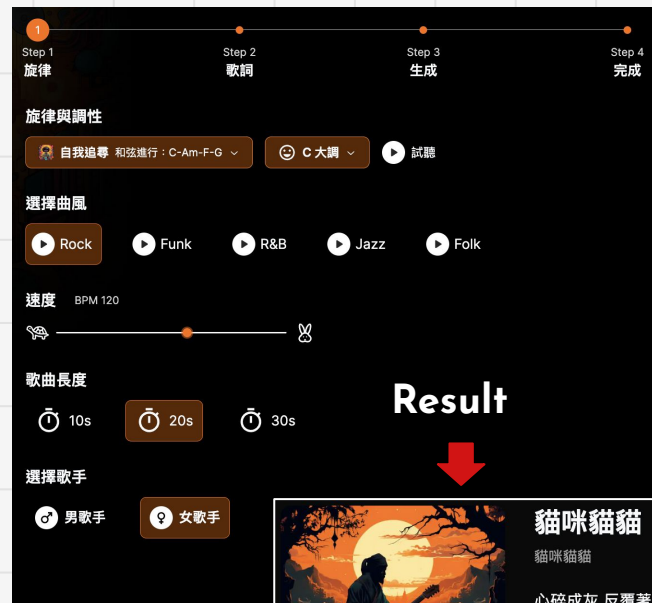
Human understand music with notations and the conceptualized **informations**:

- BPM
- Meter
- Lead Sheet
 - Key
 - Chord
 - Melody
- Arrangement
- Structure
- MIDI
- Sheet Music
 - Staff and Tablature
- Genre
- Description (Autotagging)

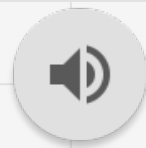
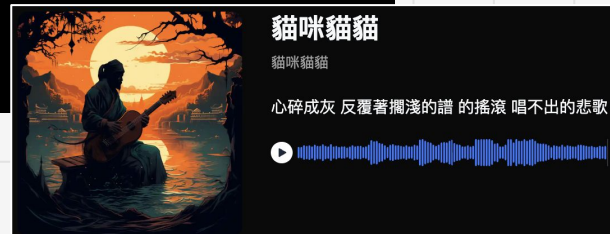
What are the Models to extract theses infos?

Why?

1. For GenAI: Understand then can control
2. Recommendation System
3. Human readable format (Transcription)

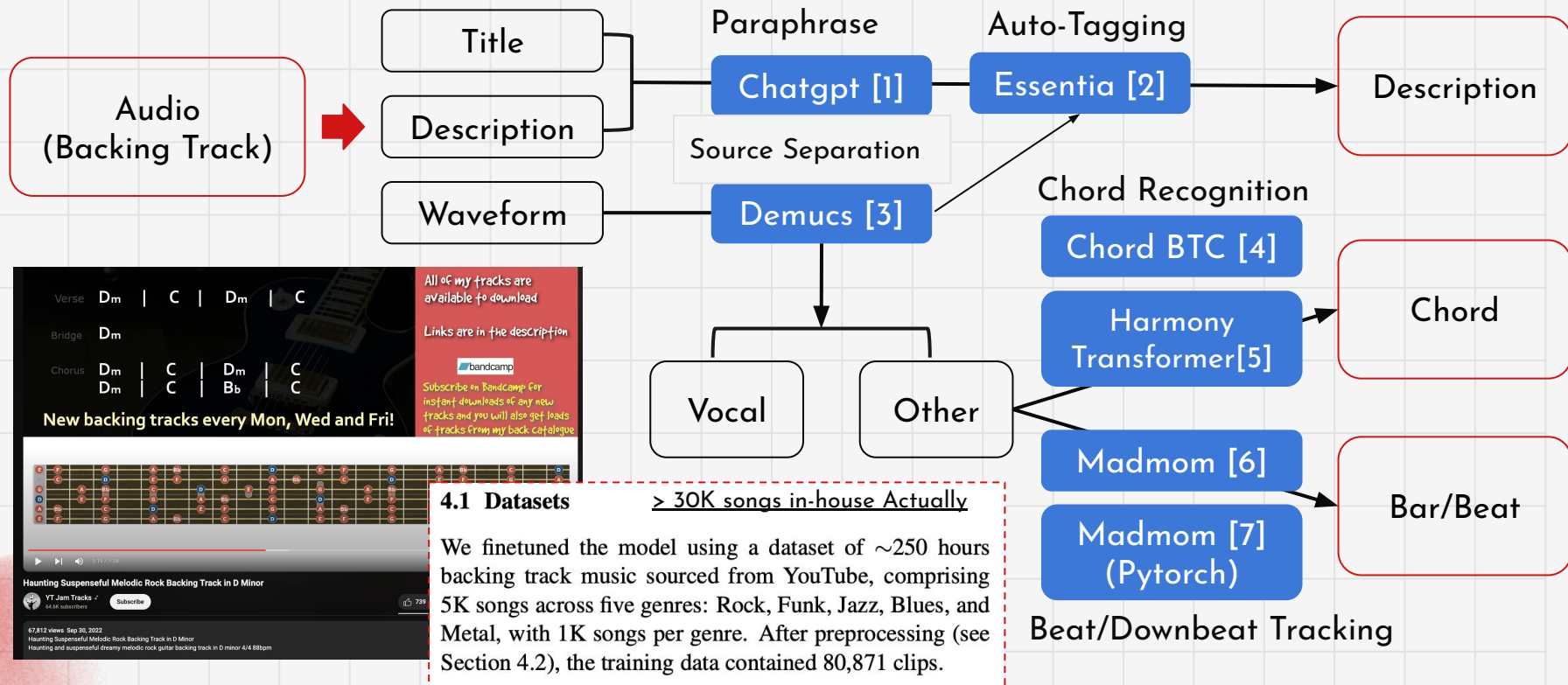


MusicConGen
(ISMIR'24)



Audio to Symbolic Domain - Example I

- MusicConGen (ISMIR'24) - Data preprocessing Pipeline**



Therefore, we have the pair (Non-Vocal Audio, Text, Chord, Beat/Downbeat) as the training data

Audio to Symbolic Domain - Example I

References:

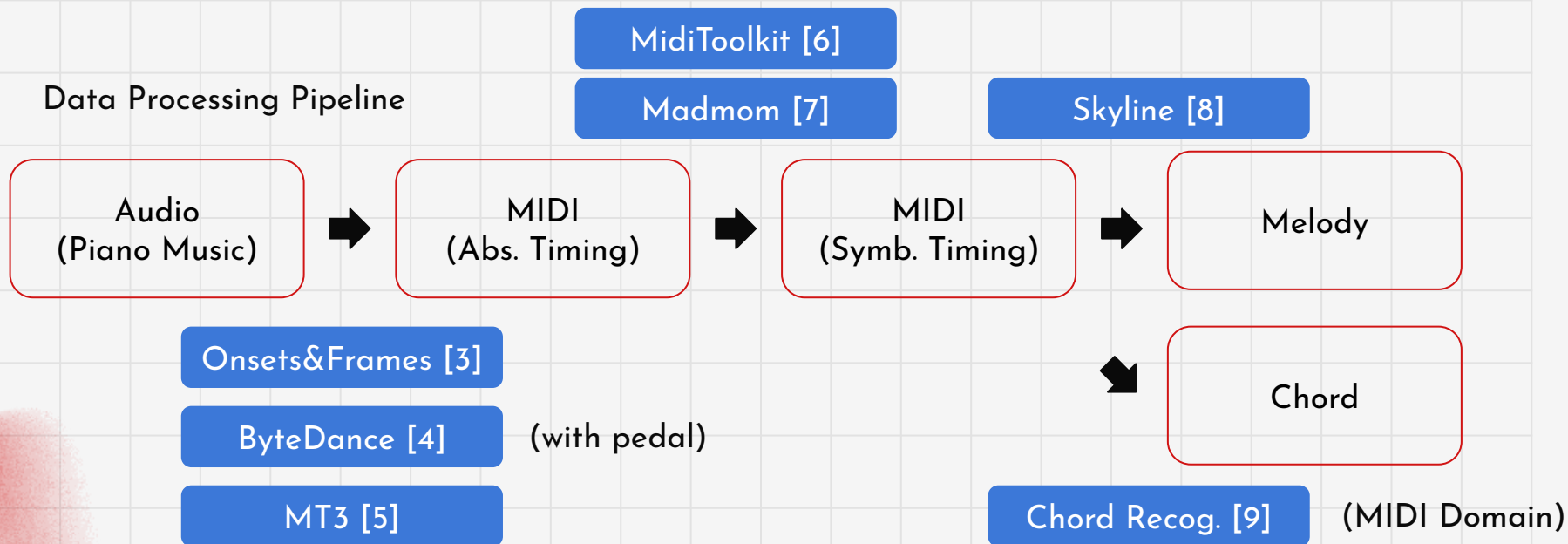
- [1] [ChatGPT API](#)
- [2] [MTG/essentia](#)
- [3] [facebookresearch/demucs](#)
- [4] [jayg996/BTC-ISMIR19](#)
- [5] [Tsung-Ping/Harmony-Transformer-v2](#)
- [6] [CPJKU/madmom](#)
- [7] [ben-hayes/beat-tracking-tcn](#)

Sample File:

```
{
  "key": "G",
  "artist": "",
  "sample_rate": 48000,
  "file_extension": "wav",
  "description": "",
  "keywords": "",
  "duration": 30.0,
  "bpm": 112,
  "genre": "Rock, Gothic Metal, Death Metal, Doom
           Metal, Goth Rock, Melodic Death Metal, Progressive
           Metal, Heavy Metal",
  "title": "",
  "name": "",
  "instrument": [
    "drums",
    "electricguitar",
    "bass",
    "guitar",
    "synthesizer",
    "voice",
    "keyboard"
  ],
  "moods": "epic, dark, melodic, heavy, energetic,
           sad",
  "path": "./_4mH2HwVF-0/13/no_vocal.wav"
}
```

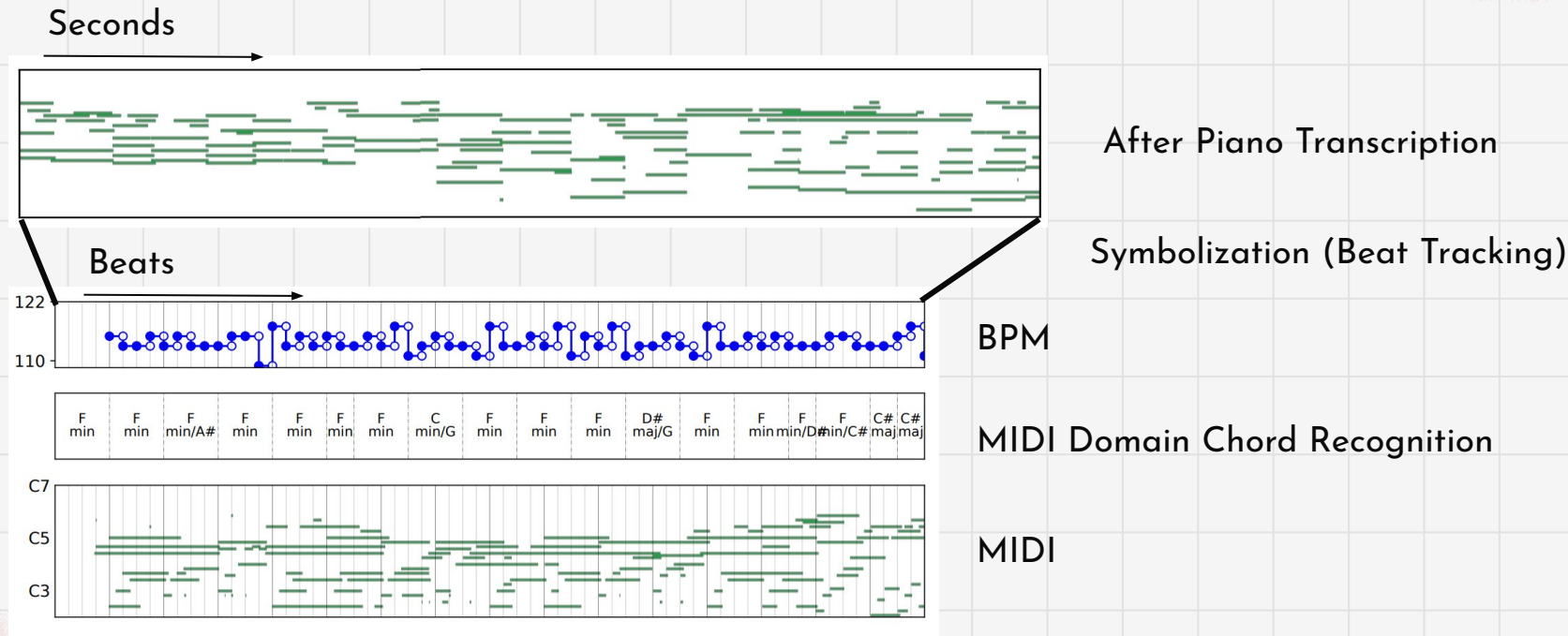
Audio to Symbolic Domain - Example II

1. **Goal:** {Piano MIDI, Lead Sheet} x {Transcription, Generation}
 - a. Compound Word Transformer (AAAI'21) [1]
 - b. REMI (ACMM MM'20) [2]



Audio to Symbolic Domain - Example II

Explain: Timing Symbolization with Miditoolkit [6] and Madmom [7]



MIDI Chord Recognition Toolkit: Choder (91 starts)
developed by me and our former intern

Audio to Symbolic Domain - Example II

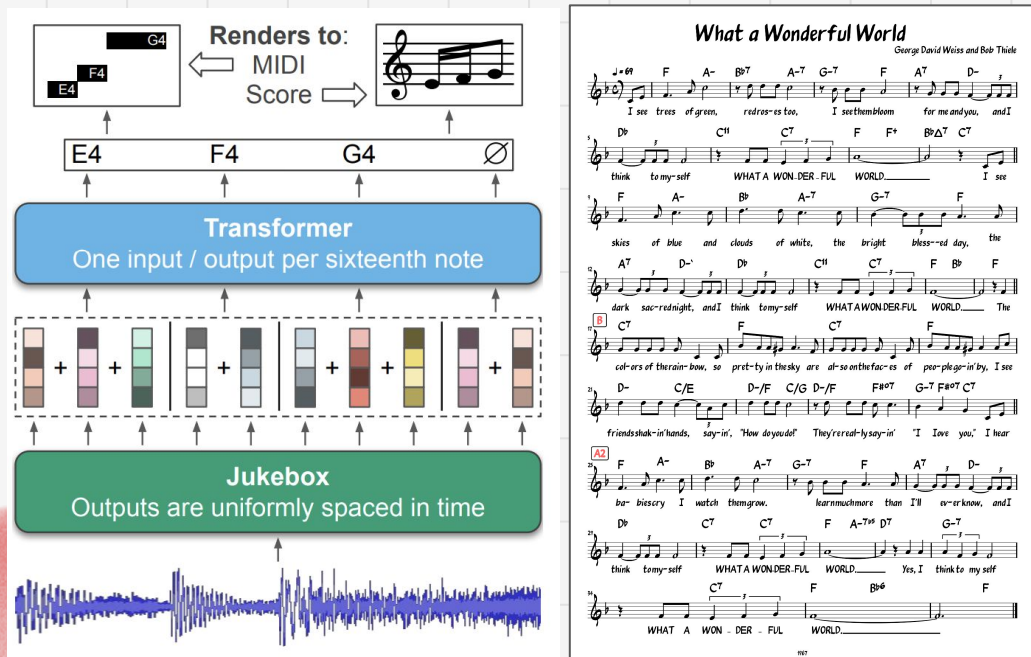
References:

- [1] [YatingMusic/compound-word-transformer](#)
- [2] [YatingMusic/remi](#)
- [3] [jongwook/onsets-and-frames](#)
- [4] [bytedance/piano_transcription](#)
- [5] [magenta/mt3](#)
- [6] [YatingMusic/miditoolkit](#)
- [7] [CPJKU/madmom](#)
- [8] [MIDI-BERT/tree/CP/melody_extraction/skyline](#)
- [9] [joshuachang2311/chorder](#)

Audio to Symbolic Domain - Example III

Goal: Lead Sheet Generation

- SheetSage on 20K in-house curated pop song



Chord is the key to our all service

- **Chord** to Vocal Melody
- **Chord** + Melody to Piano MIDI
- **Chord** + Text to Music

“Chord” can make individually generated tracks sound harmonic

- Sheetsage Problem
 - Extremely Slow
 - Jukebox Pretrained Feats

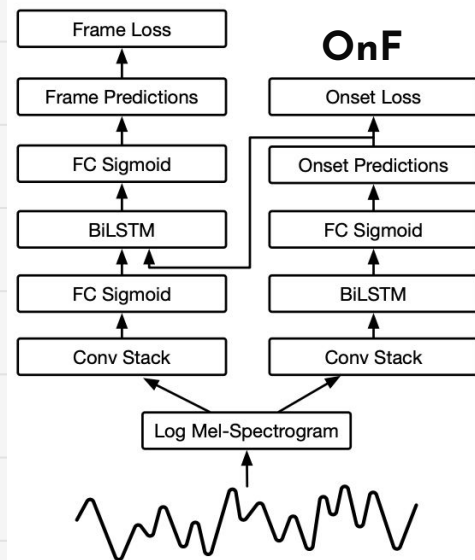
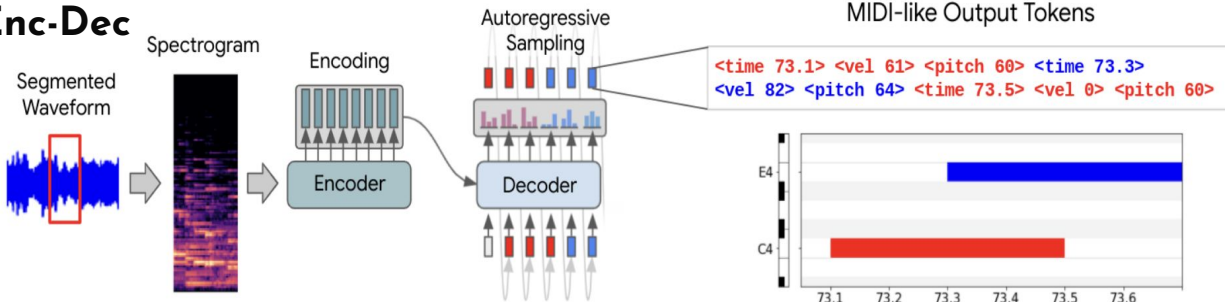
Sheet music for the song "What a Wonderful World" by George David Weiss and Bob Thiele. The music is written for voice and piano, featuring chords and lyrics. The chords are written above the notes, and the lyrics are written below the notes. The music is in 4/4 time and G major. The lyrics are: "I see trees of green, red-roofs too, I see them bloom for me and you, and I think to my self WHAT A WON DER FUL WORLD I see the skies of blue and clouds of white, the bright bless-ed day, the sun, shi-ne, and I think to my self WHAT A WON DER FUL WORLD The colors of the rain-bow, so pret-ty in the sky are al-so on the fac-es of peo-ple-giv-ing, I see friends shak-ing hands, say-in', 'How do you do?' They're real-ly say-in' 'I love you,' I hear bar-bies cry I watch them grow, learn-much more than I'll ev-er know, and I think to my self WHAT A WON DER FUL WORLD Yes, I think to my self WHAT A WON DER FUL WORLD."

Audio to Symbolic Domain - Example IV

Transcription - Transcribe Audio into MIDI

1. Onset and Frames (Onf)
(by Curtis Hawthorne, ISMIR'17)
2. Sequence-to-Sequence Piano Transcription with Transformers
(by Curtis Hawthorne, ISMIR'21)
3. MT3: Multi-Task Multitrack Music Transcription
(by Josh Gardner, ICLR'22)

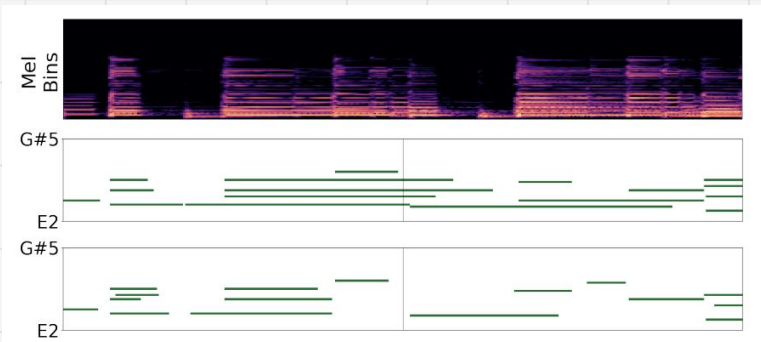
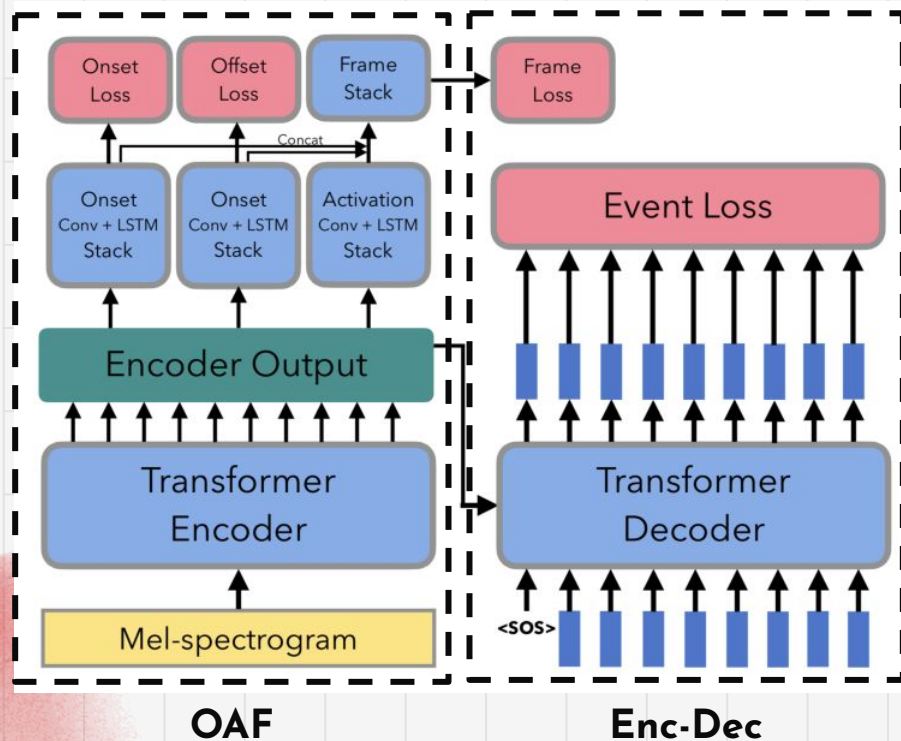
Enc-Dec



Audio to Symbolic Domain - Example IV

Inspired by (1), (2) and (3)

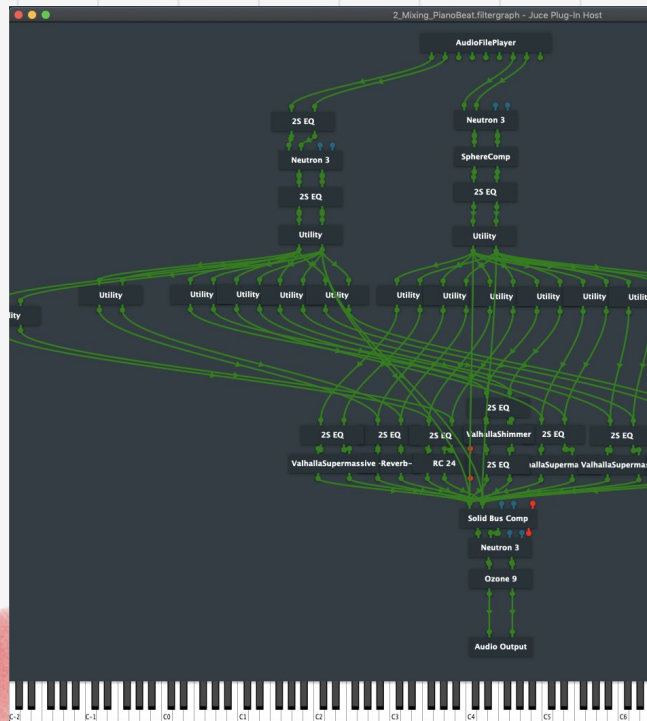
We proposed a Novel Guitar Transcription Model (ICASSP'22)



Model	(Encoder output)	
	Onset F1	Frame F1
Onset and Frame (OAF) [14]	0.591	0.583
CE-only Transformer [17]	0.543	0.523
	0.554	0.524
	0.568	0.537
Proposed multi-loss Transformer	0.598	0.579
	0.604	0.573
	0.613	0.582

Audio to Symbolic Domain - Example IV

... and a new Guitar Dataset - EGDB



DI (Direct Input)
Clean Signal



Guitar Rig Plugin

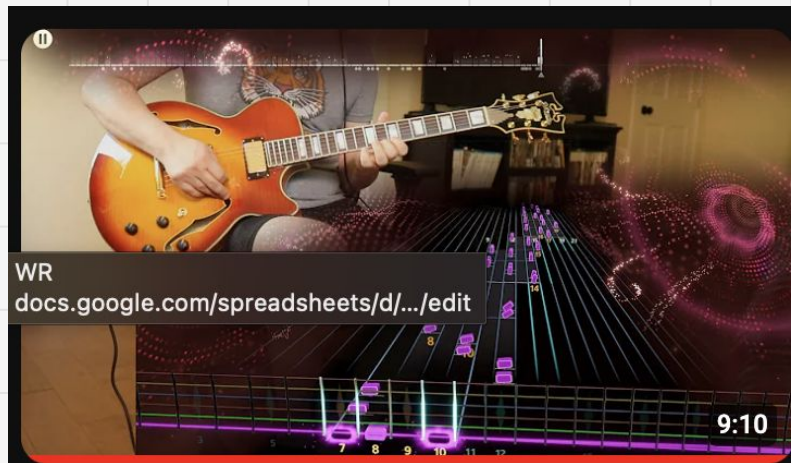
Colored Singal

- The DI (input signal) is recorded by musician
 - Given a Tab
 - Sight Reading Performance
 - w/ a special pickup
 - Post-processing
 - Human Curation
 - Rendered by JUCE
 - w/ different tones
- We have (tab, DI, color) pairs
 - Audio of individual string



Audio to Symbolic Domain - Example IV

Current Plan on Guitar, a Larger Dataset



"Master of Puppets" Metallica - Lead Rocksmith+

觀看次數：54萬次 · 1 年前



Riff Repeater

Available Worldwide! not only have we lived to see moar muse in rb, but there's officially Metallica in **Rocksmith**. Albeit in a...

4K

INTRO D7

Am7

Am7 D7

41 65 66

6 7 8 9 10 11 12

13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200

201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300

301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400

401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500

501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600

601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700

701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800

801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900

901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000

1001 1002 1003 1004 1005 1006 1007 1008 1009 1010 1011 1012 1013 1014 1015 1016 1017 1018 1019 1020 1021 1022 1023 1024 1025 1026

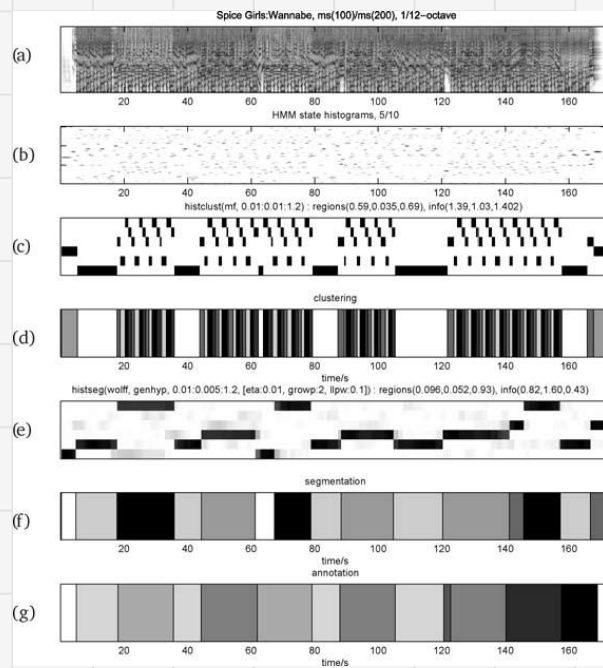
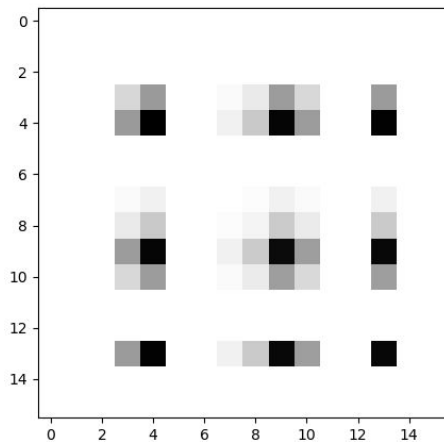
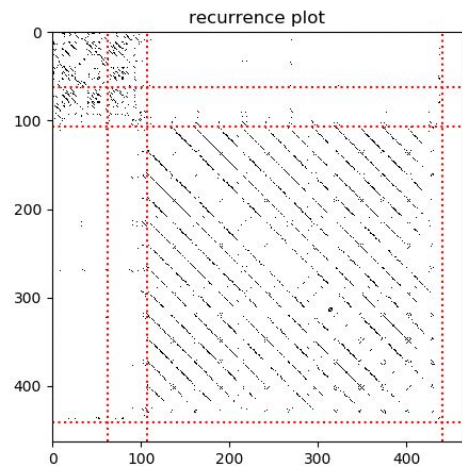
Our Vision:

- Aligned audio and tab
- Tab, not only MIDI (Position & Fingering)
- Chord label
- Over 1K songs
- Multi-track guitar
- Transcription

Audio to Symbolic Domain - Example V

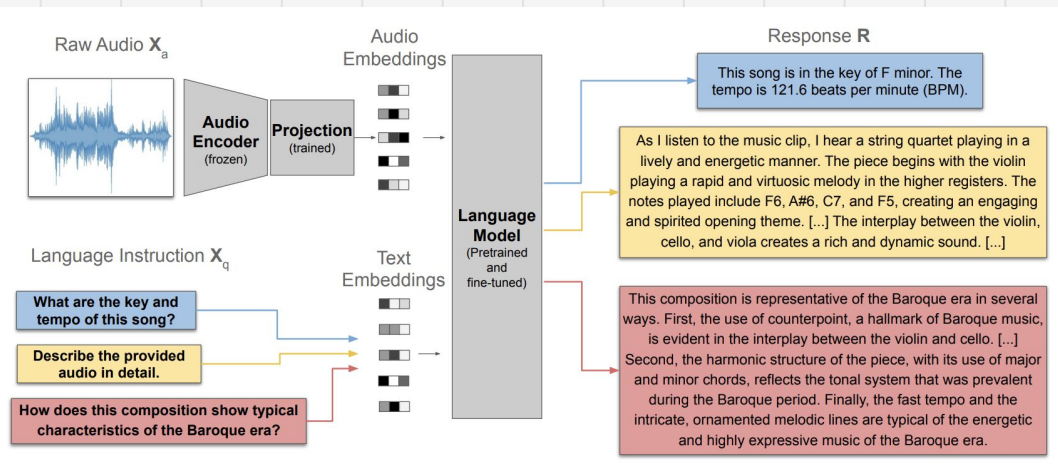
Structure = Boundary + Section Labeling

- MSAF Toolkit, by Oriol Nieto, ISMIR'16
- Unsupervised Music Structure Annotation w/ Structure Features (SF)
 - Joan Serrà, AAAI'16, IEEE MM'17
- SF Segmenter (by me, 52 stars), works on **MIDI & Audio**



Audio to Symbolic Domain - Example VI

- LLark (from spotify)



- MERT

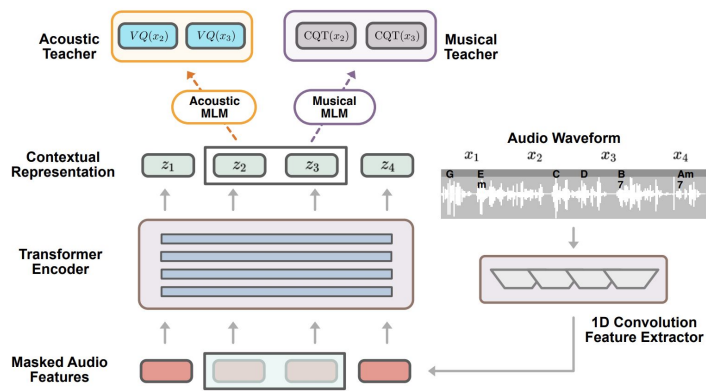
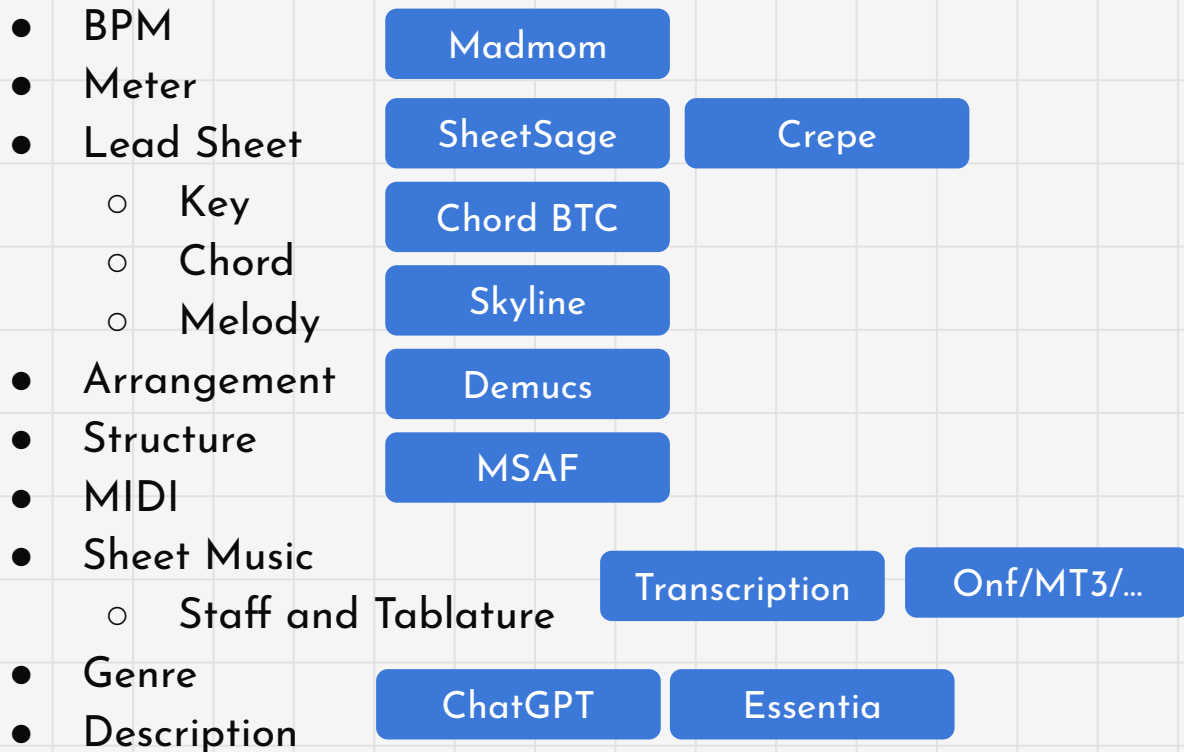


Figure 1: Illustration of the MERT Pre-training Framework.

- Not enough resources (especially GPU) in my current company : (
- But... Rethinking the necessity?
- If there are enough resources, I can do scaling with my expertise :)

Audio to Symbolic Domain

Quick Review of My MIR Tech Stack



Baseline of All - **Essentia from MTG**, an old but **fast** universal Auto-Tagging model



03

Symbolic To Audio Domain

Generative AI Music

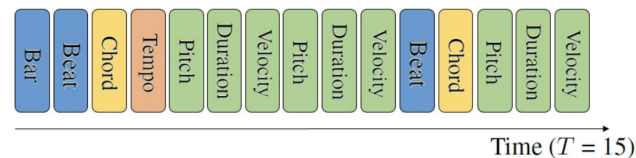
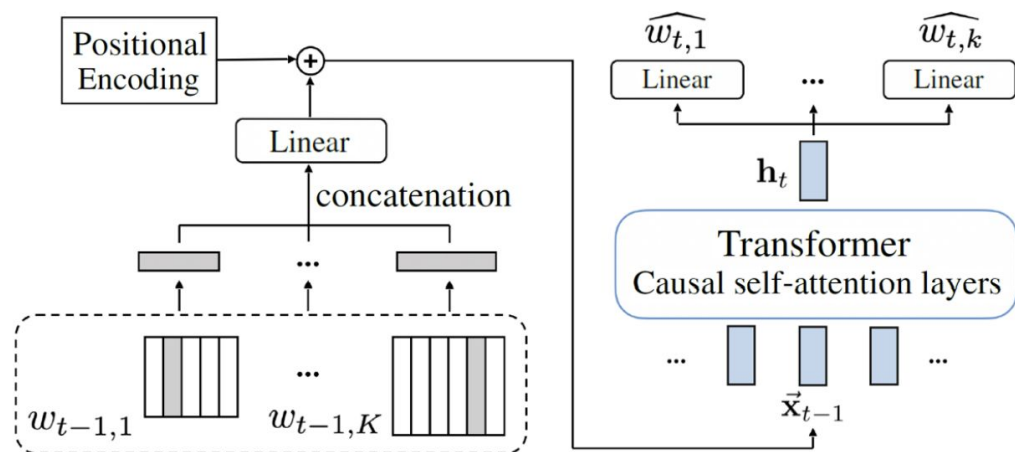
Symbolic to Audio to Domain - Example I

Goal: Generate Piano MIDI (Symbolic Domain Generation)

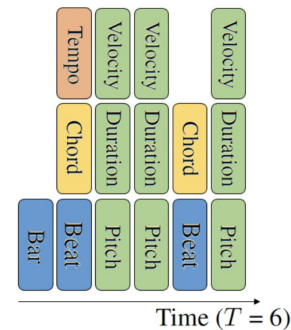
- Compound Word Transformer** (AAAI'21) [1] | DEMO
- REMI** (ACM MM'20) [2]

MIDI Note = Pitch + Duration + Velocity

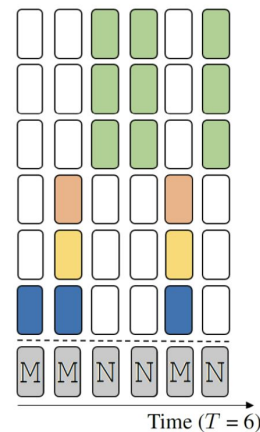
MIDI Meta Events: BPM, Time Signature, ...



(a) REMI representation



(b) Tokens grouped

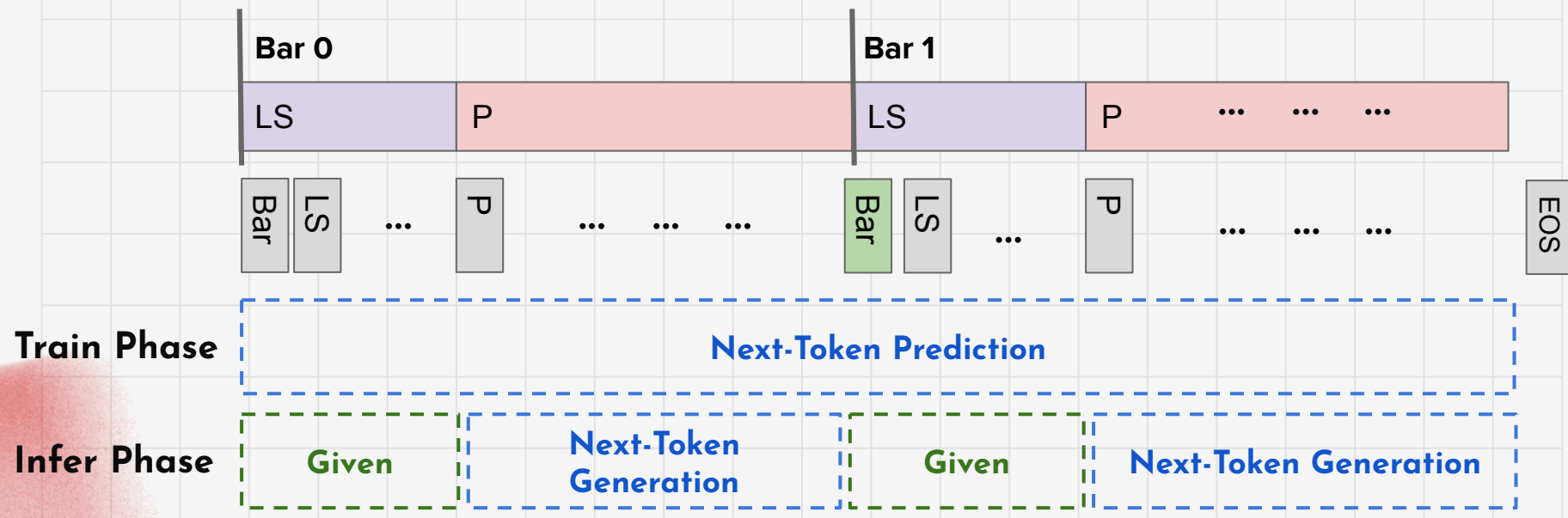


(c) Compound words

Symbolic to Audio to Domain - Example I

Conditional Generation, with decoder only (GPT-like) transformer

- Condition: Lead Sheet (L)
- Generation: Piano MIDI (P) - can be generalized to multi-track
- **T5 Prefix-LM** Mechanism



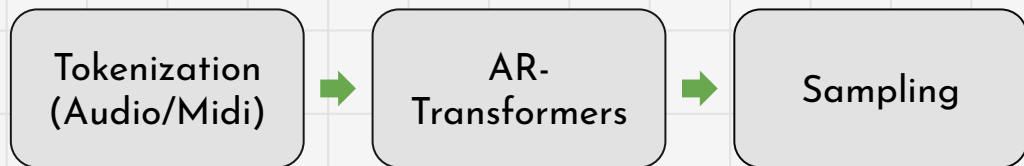
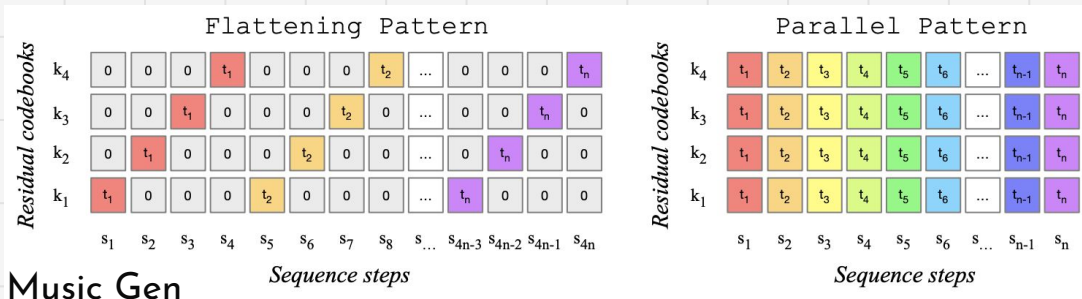
Symbolic to Audio to Domain - Example I

● Design Principle

- Token Length (Tokenization)
 - Length Compression
- Memory Complexity of Transformers
 - $O(N^2)$, N is seq len
 - Transformer-XL
 - Linear Transformer
- Sampling Policy
 - beam-search
 - Top-k, w/ temp
 - **Top-p**

CP Transformer

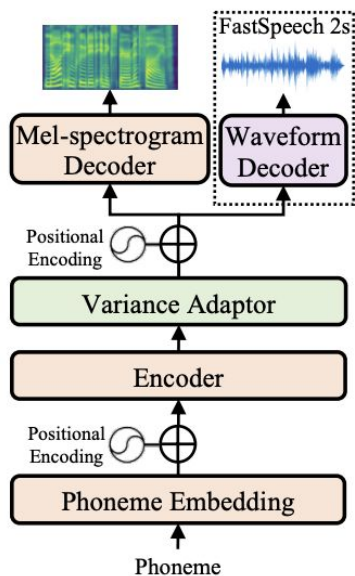
Task	Repre.	#words (T)	
		mean (\pm std)	max
Conditional	REMI	6,432 (\pm 1,689)	10,240
	CP	3,142 (\pm 821)	5,120
Unconditional	REMI	4,873 (\pm 1,311)	7,680
	CP	2,053 (\pm 580)	3,584



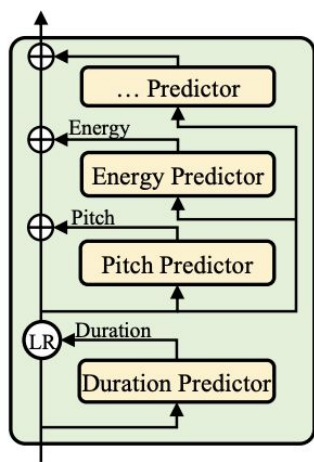
Symbolic to Audio to Domain - Example II

Singing Voice Synthesis = FastSpeech2 (modified) + Singing Vocoder

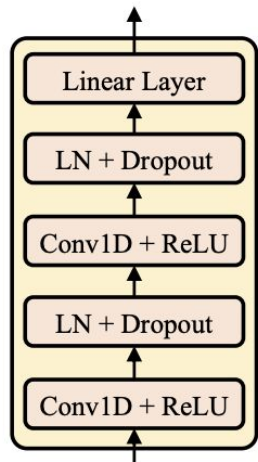
Original FastSpeech2



(a) FastSpeech 2

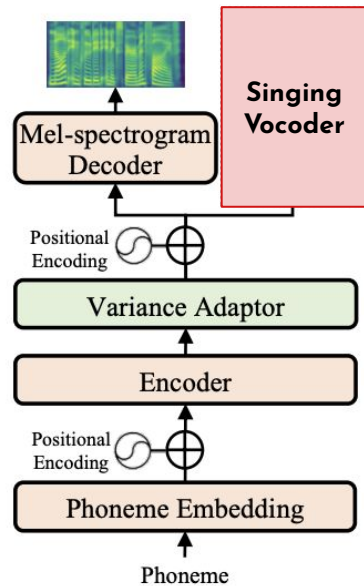


(b) Variance adaptor

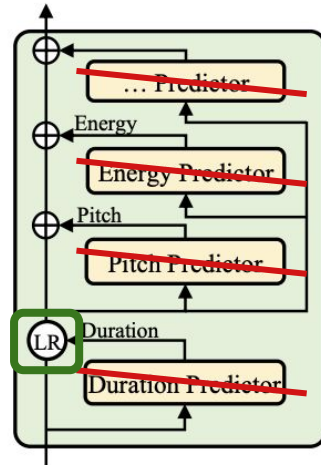


(c)
Duration/pitch/energy
predictor

Modified FastSpeech2 for Singing
Given: Duration/Pitch/Velocity(Energy)



(a) FastSpeech 2

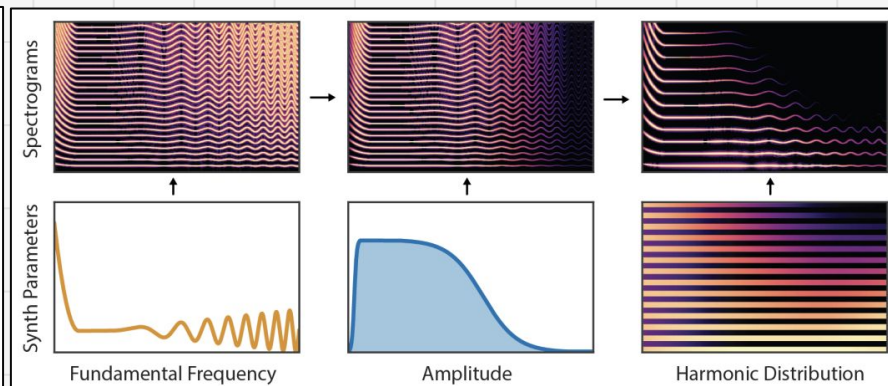
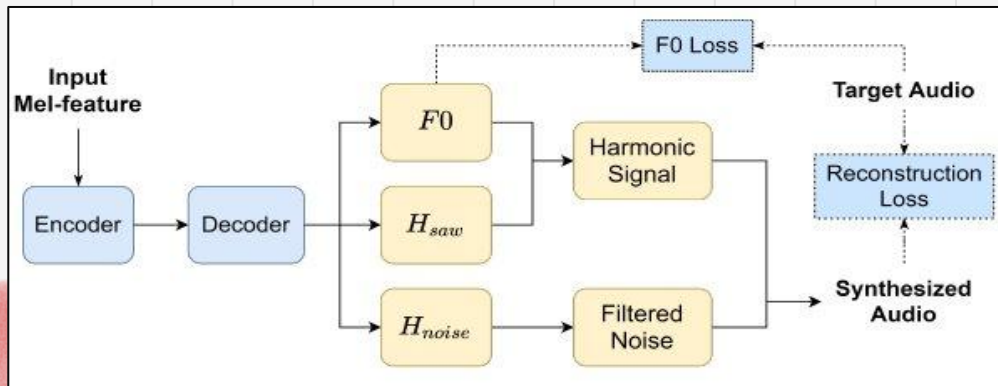
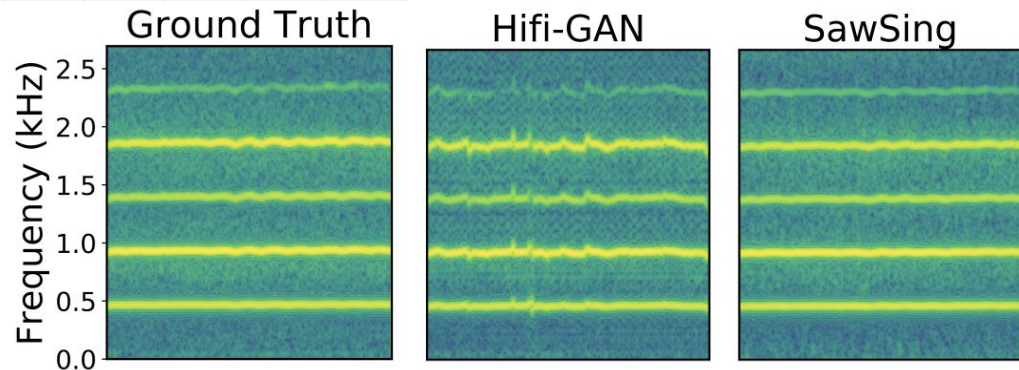


(b) Variance adaptor

Symbolic to Audio to Domain - Example II

DDSP Singing Vocoder (ISMIR'22)

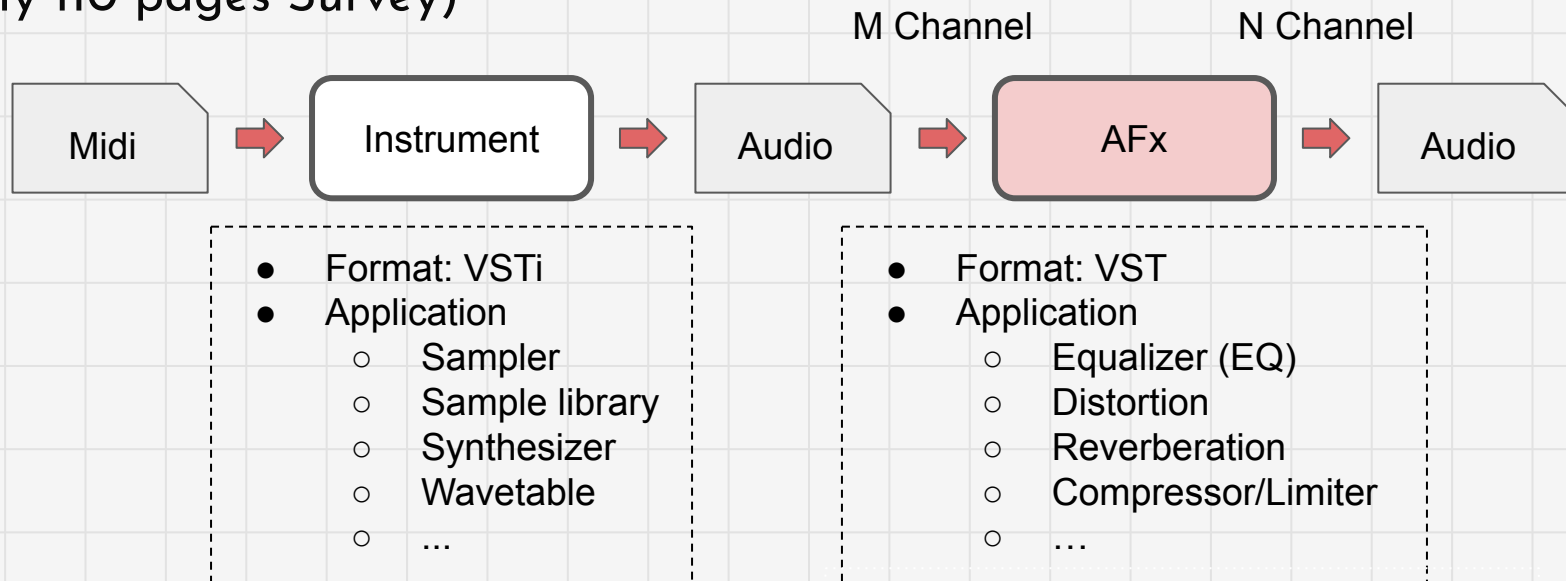
- DEMO
- NN-based Vocoder: slow
- No source signal input:
 - glitch in long utterance



Symbolic to Audio to Domain - Example III

Neural Audio Effect Modeling

(My 110 pages Survey)



x : input signal, M channel

y : output signal, N channel

c_g : gloabl condition

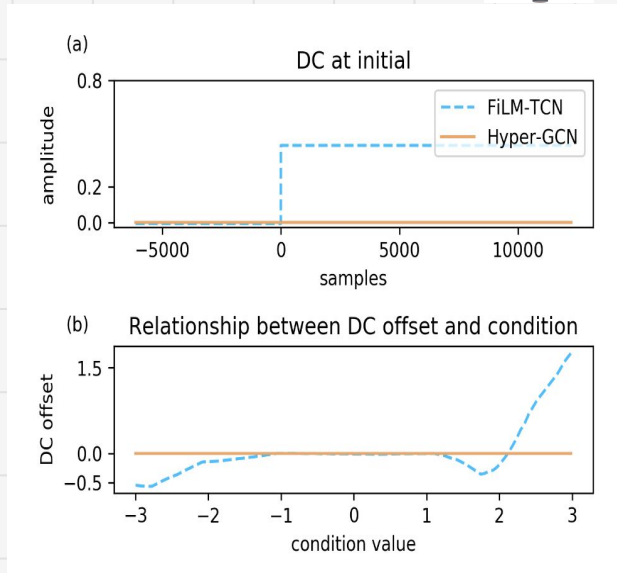
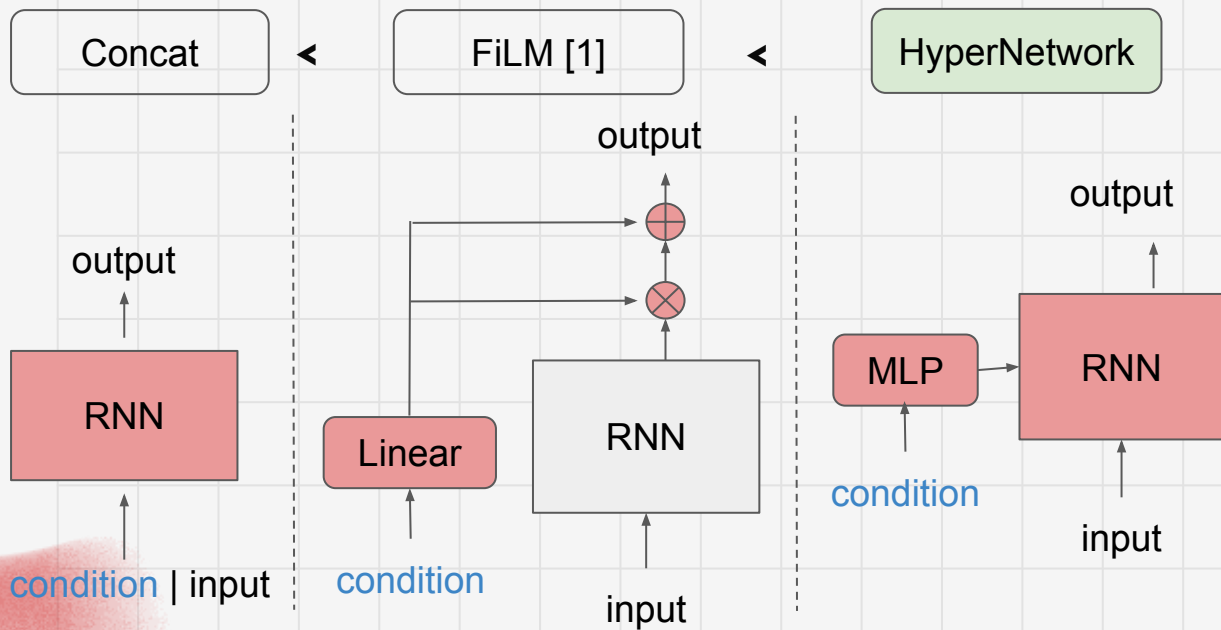
c_t : local condition

$$y = f(x, c_t, c_g)$$

Symbolic to Audio to Domain - Example III

Neural Audio Effect Modeling - DAFX'24 Oral

- Improve Quality and Solve DC Bias Issue

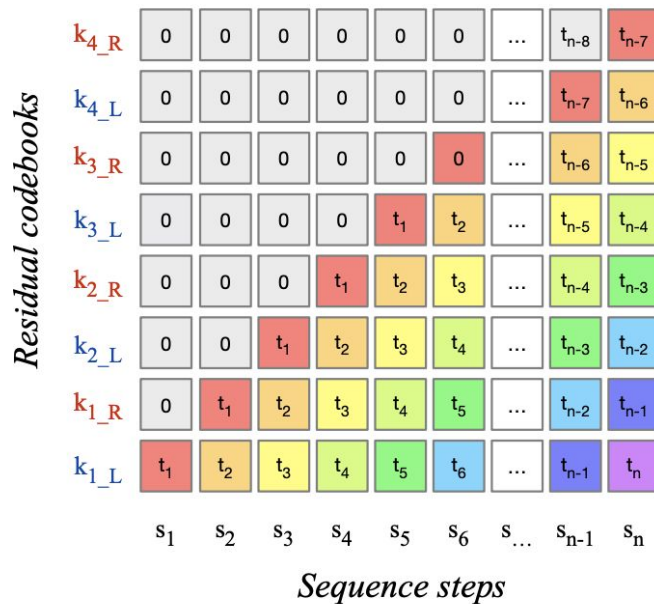


Symbolic to Audio to Domain - Example IV

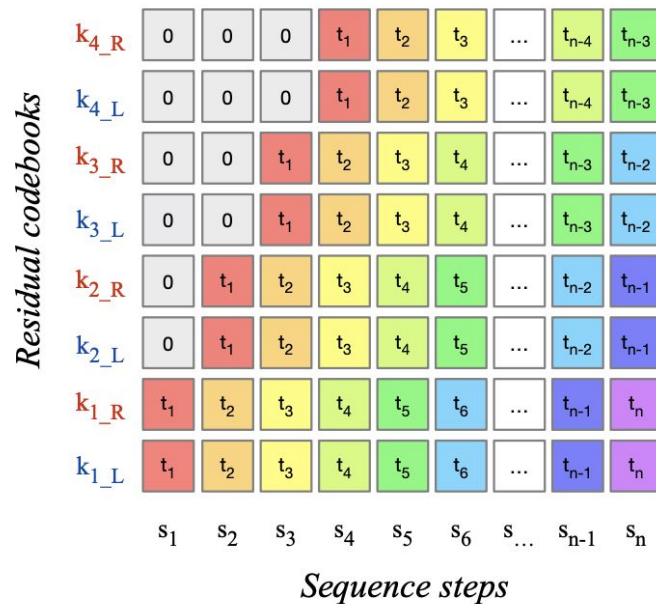
Text2Music with Temporal Controllation - MusicConGen (ISMIR'24)

- Fine-tune MusicGen (w/ melody) to control tempo & chord

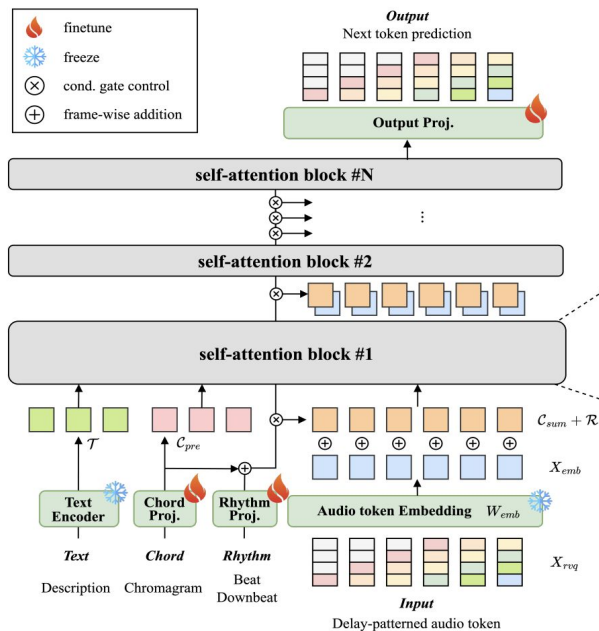
Stereo Delay Pattern



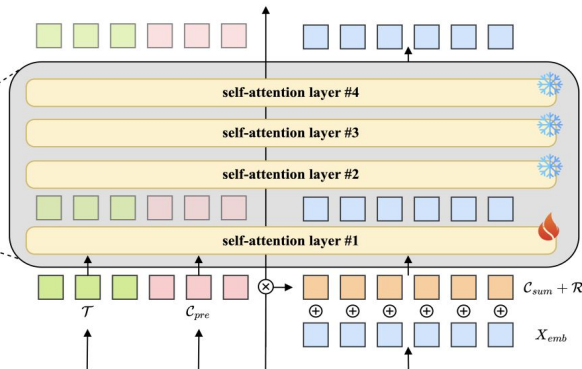
Stereo Partial Delay Pattern



Symbolic to Audio to Domain - Example IV



(a) MusiConGen model structure



(b) self-attention block

- MusicGen [1] =
 - RVQ [2] +
 - Flash-Attn [3]
- MusicConGen =
 - Module Reuse
 - Fine-tuning
- Fine-tune on
 - Single RTX3090
 - Inhouse Data

• DEMO

Reference Chords																																			
Generated Sample's Chords																																			
	<div>0.410.821.221.632.032.442.833.253.664.064.464.875.265.686.076.496.97.317.78.128.528.939.319.7310.1410.5510.9411.3611.7512.1712.5612.9813.3913.7914.1914.615.015.4115.816.2216.6317.0417.4317.8518.2418.6519.0419.4719.8820.2820.6821.0921.4921.922.2922.7123.1223.5323.9224.3424.7725.1425.5325.9526.3626.7727.1627.5828.028.3928.7729.1629.6</div>																																		
description	A laid-back blues shuffle with a relaxed tempo, warm guitar tones, and a comfortable groove, perfect for a slow dance or a night in. Instruments: electric guitar, bass, drums.							A smooth acid jazz track with a laid-back groove, silky electric piano, and a cool bass, providing a modern take on jazz. Instruments: electric piano, bass, drums.							A classic rock n' roll tune with catchy guitar riffs, driving drums, and a pulsating bass line, reminiscent of the golden era of rock. Instruments: electric guitar, bass, drums.							A high-energy funk tune with slap bass, rhythmic guitar riffs, and a tight horn section, guaranteed to get you grooving. Instruments: bass, guitar, trumpet, saxophone, drums.							A heavy metal onslaught with double kick drum madness, aggressive guitar riffs, and an unrelenting bass, embodying the spirit of metal. Instruments: electric guitar, bass guitar, drums.						
Sample 001	<div>▶0:00 / 0:30</div> <div><div></div><div></div><div></div><div></div></div>							<div>▶0:00 / 0:30</div> <div><div></div><div></div><div></div><div></div></div>							<div>▶0:00 / 0:30</div> <div><div></div><div></div><div></div><div></div></div>							<div>▶0:00 / 0:30</div> <div><div></div><div></div><div></div><div></div></div>							<div>▶0:00 / 0:30</div> <div><div></div><div></div><div></div><div></div></div>						

- [1] Simple and Controllable Music Generation (Neurips'23)
- [2] High-Fidelity Audio Compression with Improved RVQGAN (Neurips'23)
- [3] FlashAttention

Thank you

My Paper Reading Notes



01

LLM & FM

Large Language Model (LLM) and
Foundation Model (FM)

LLM & FM – Definition

- What is **Language Model (LM)**?

$$\begin{aligned} P_{(w_1, w_2, \dots, w_n)} &= p(w_1)p(w_2|w_1)p(w_3|w_1, w_2)\dots p(w_n|w_1, w_2, \dots, w_{n-1}) \\ &= \prod_{i=1}^n p(w_i|w_1, \dots, w_{i-1}) \end{aligned} \quad (1)$$

- What is **Large Language Model (LLM)**?
 - It's LM trained on large corpus with large amount of parameters (Billion/Trillion).
 - ChaptGPT, LLama
- What is **Foundation Model (FM)**?
 - It's a broader concept including LLMs
 - Multimodal data, including images, audio, video, and text.
 - In a Paradigm like {Petrained, Fine-Tuning}
 - GPT, CLIP, CLAP, BERT

LLM & FM – Examples

Three families:

- **BERT-like (Transformer Encoder)**
- **GPT-like (Transformer Decoder)**
- **CLIP-like (Contrastive Learning)**

Two Topics:

- **How to Fine-Tuning?**
- **Hallucination**

BERT-like

- **Training Goals**

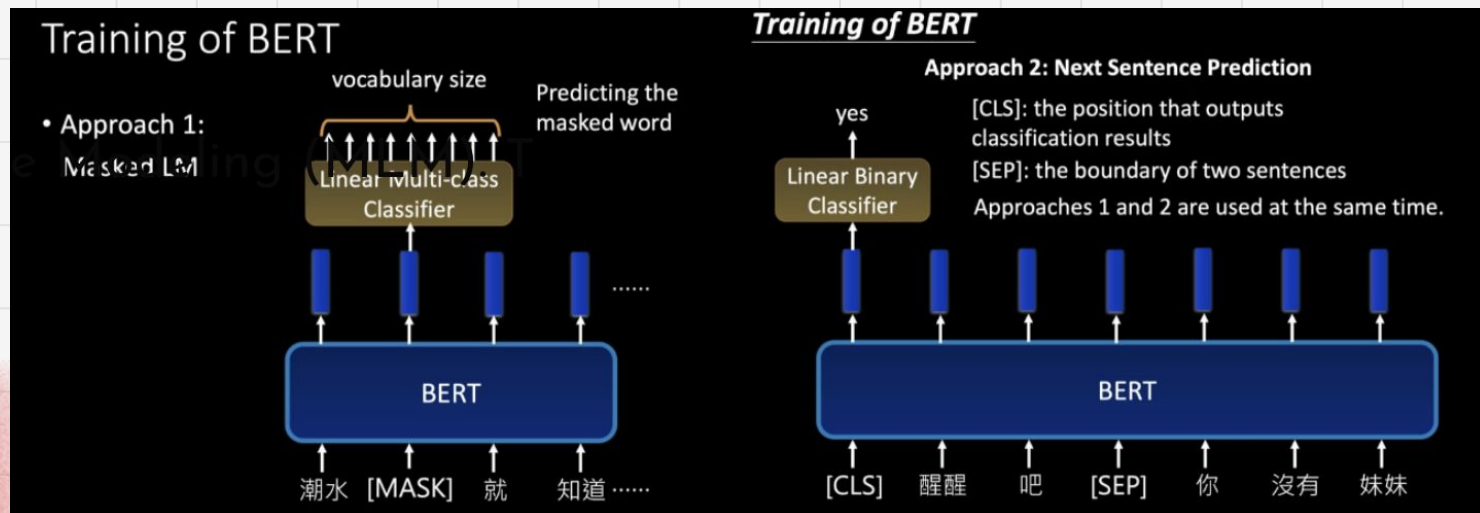
- **Masked Language Modeling (MLM).**
- **Next Sentence Prediction (NSP).**

- **Applications**

- Feats for Downstream task

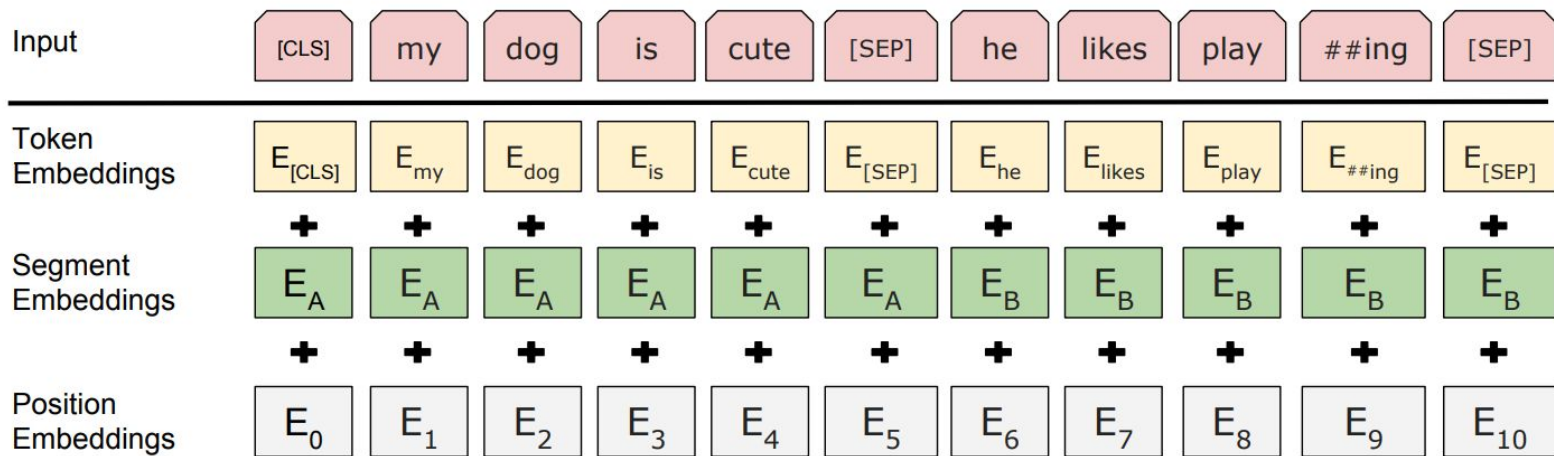
- **Difficulties for other domain**

- Transformers work on discrete tokens
- How to discretize continuous feats?
 - Spectrogram, Imagem ...



BERT-like

- **Backbone Model - Transformer Encoder**
 - No causal mask
 - Bidirectional
 - Non-autoregressive model



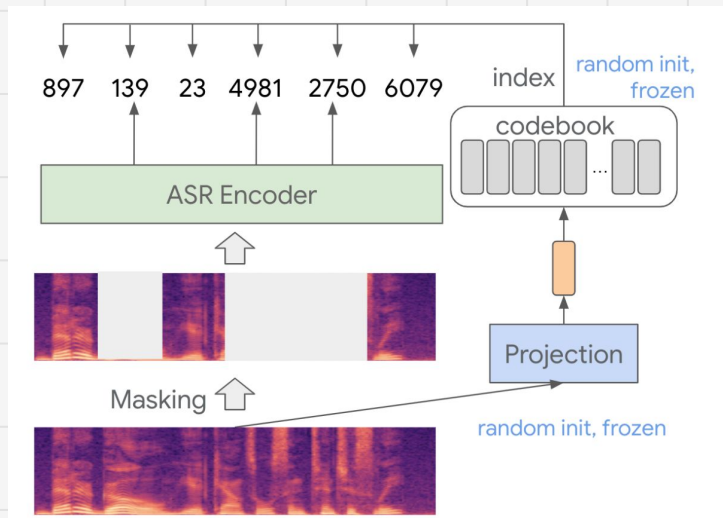
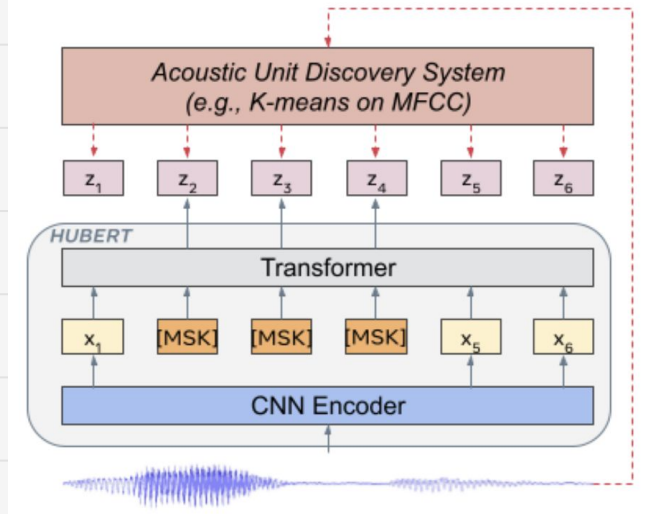
BERT-like

- **Difficulties for other domain**

- Transformers work on discrete tokens
- How to discretize continuous feats?
 - Spectrogram, Image, ...

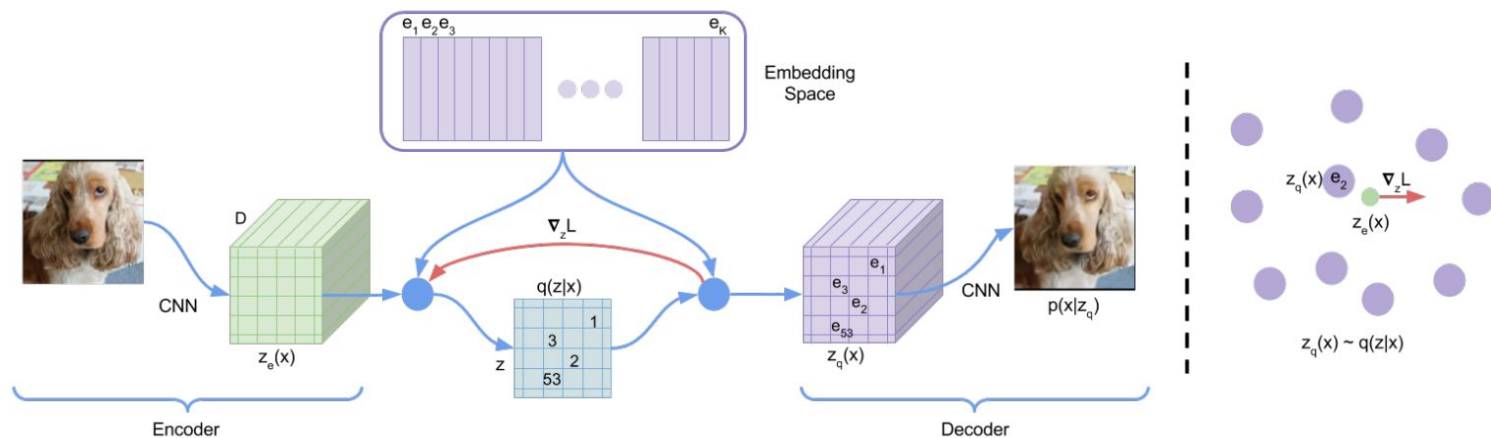
- **Examples on Audio**

- Wav2Vec, HuBERT, Best RQ.
- How to discretize audio waveform?
 - Hubert: K-means Clustering
 - Best RQ: Vector Quantizer



BERT-like

- Discretization and Quantization
 - VQVAE (for image)



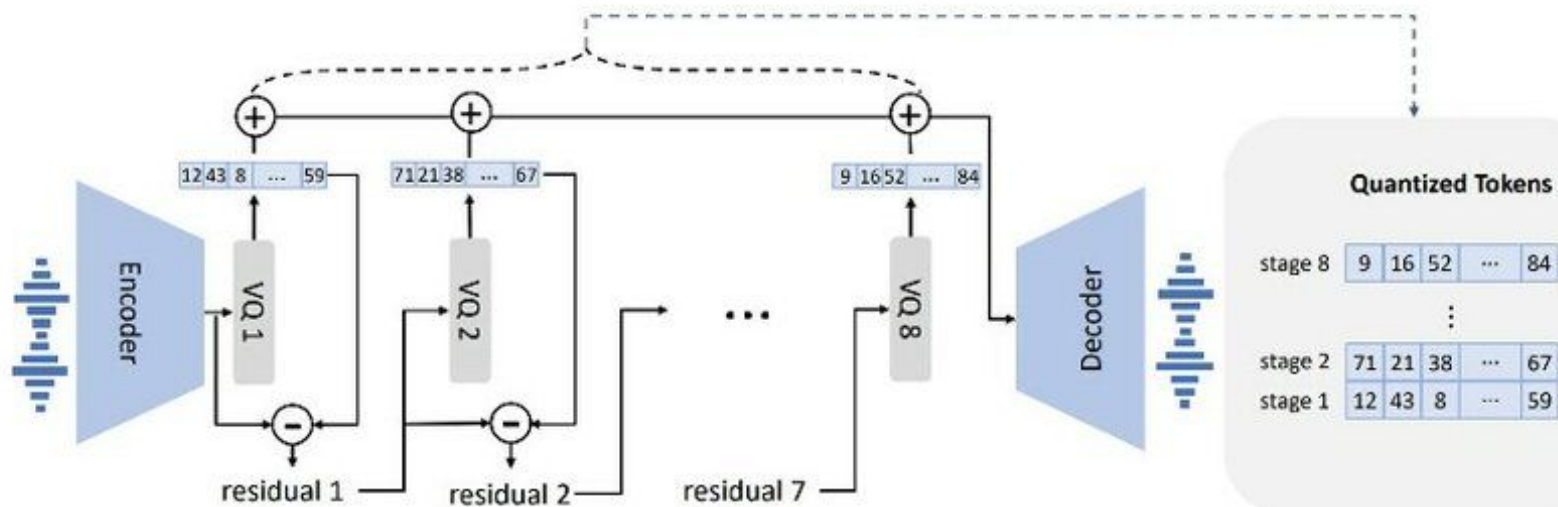
$$L = \log p(x|z_q(x)) + \|\text{sg}[z_e(x)] - e\|_2^2 + \beta \|z_e(x) - \text{sg}[e]\|_2^2, \quad (3)$$

where sg stands for the stopgradient operator that is defined as identity at forward computation time and has zero partial derivatives, thus effectively constraining its operand to be a non-updated constant.

The decoder optimises the first loss term only, the encoder optimises the first and the last loss terms, and the embeddings are optimised by the middle loss term. We found the resulting algorithm to be

BERT-like

- **Improved Version - RVQ**
 - SoundStream (from google)
 - Encodec (from Meta)



GPT-like

- **Training Goals**
 - Next Token Prediction
- **Backbone Model:**
 - **Transformer Decoder**
 - Training
 - Causal Masked
 - Inference
 - Auto-regressive
 - Sampling
- **Applications**
 - "LMs are Few-Shot Learners"
 - Prompt Interaction

The three settings we explore for in-context learning

Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

```
1 Translate English to French: ← task description
2 cheese => ..... ← prompt
```

One-shot

In addition to the task description, the model sees a single example of the task. No gradient updates are performed.

```
1 Translate English to French: ← task description
2 sea otter => loutre de mer ← example
3 cheese => ..... ← prompt
```

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

```
1 Translate English to French: ← task description
2 sea otter => loutre de mer ← examples
3 peppermint => menthe poivrée ←
4 plush girafe => girafe peluche ←
5 cheese => ..... ← prompt
```

Traditional fine-tuning (not used for GPT-3)

Fine-tuning

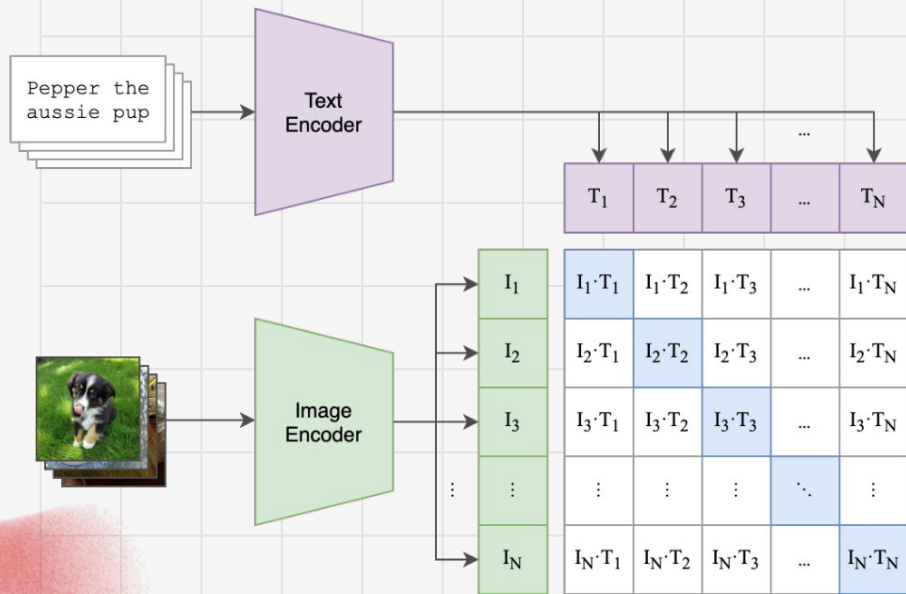
The model is trained via repeated gradient updates using large corpus of example tasks.



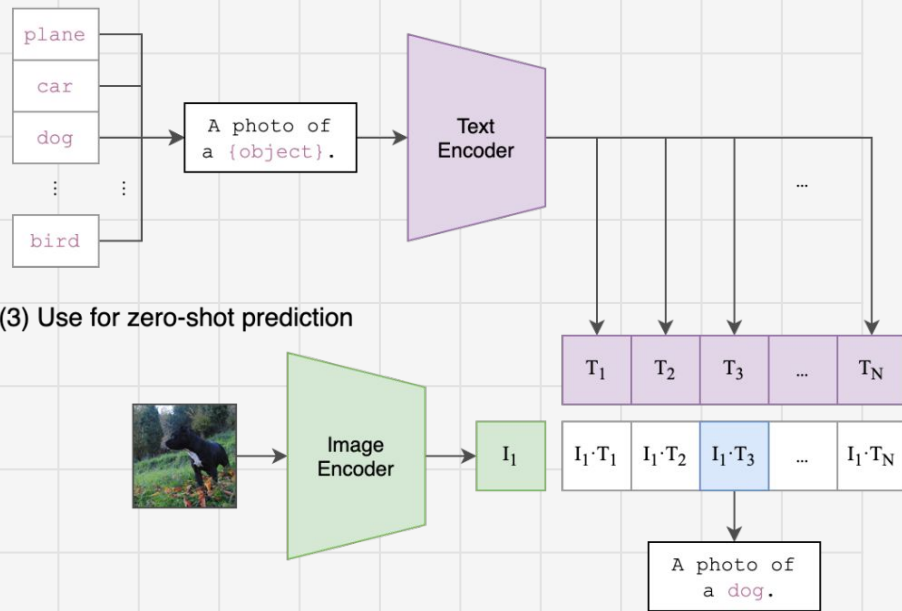
CLIP-like

CLAP: Image x Text

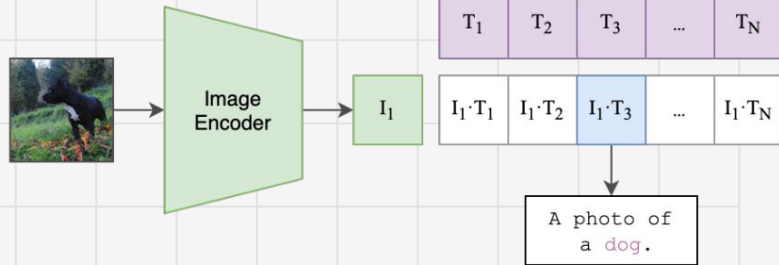
(1) Contrastive pre-training



(2) Create dataset classifier from label text

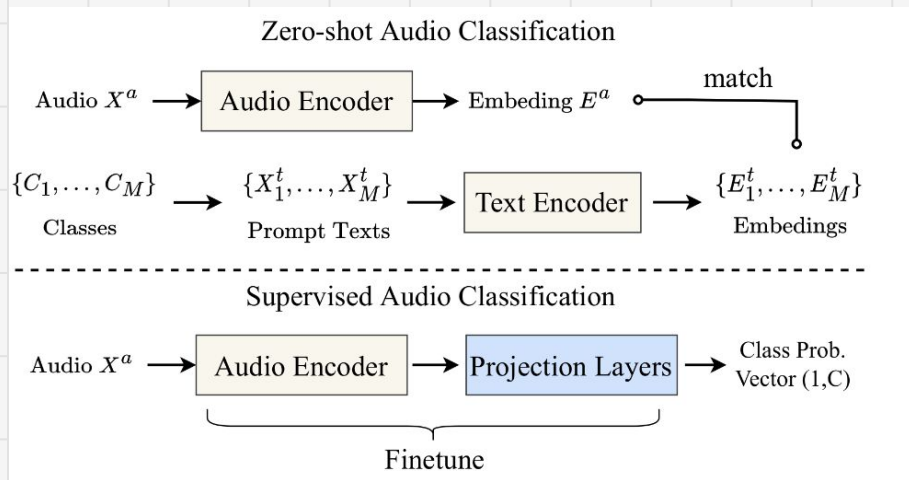
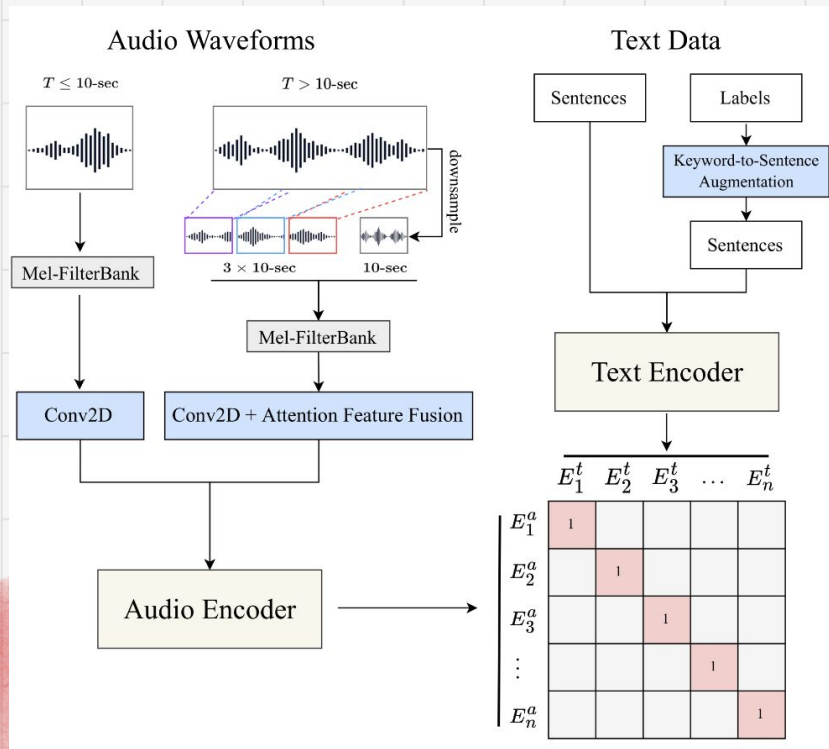


(3) Use for zero-shot prediction



CLIP-like

CLAP: Audio x Audio



How to Fine-tuning

- **Supervised Fine-Tuning (SFT)**
 - Use small and clean high quality data
 - Freeze part of trainable models, small learning rate
 - **Cons:** Gradient Update required
- **Reinforcement Learning with Human Feedback (RLHF)**
 - Similar to SFT, different rewarding policy
 - **Cons:** human annotation -> resource-intensive
- **Prompt Engineering**
 - **Zero-shot, One-shot, Few-shot**
 - **Pros:** no Gradient update

Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

```
1 Translate English to French: ← task description
2 cheese => ..... ← prompt
```

One-shot

In addition to the task description, the model sees a single example of the task. No gradient updates are performed.

```
1 Translate English to French: ← task description
2 sea otter => loutre de mer ← example
3 cheese => ..... ← prompt
```

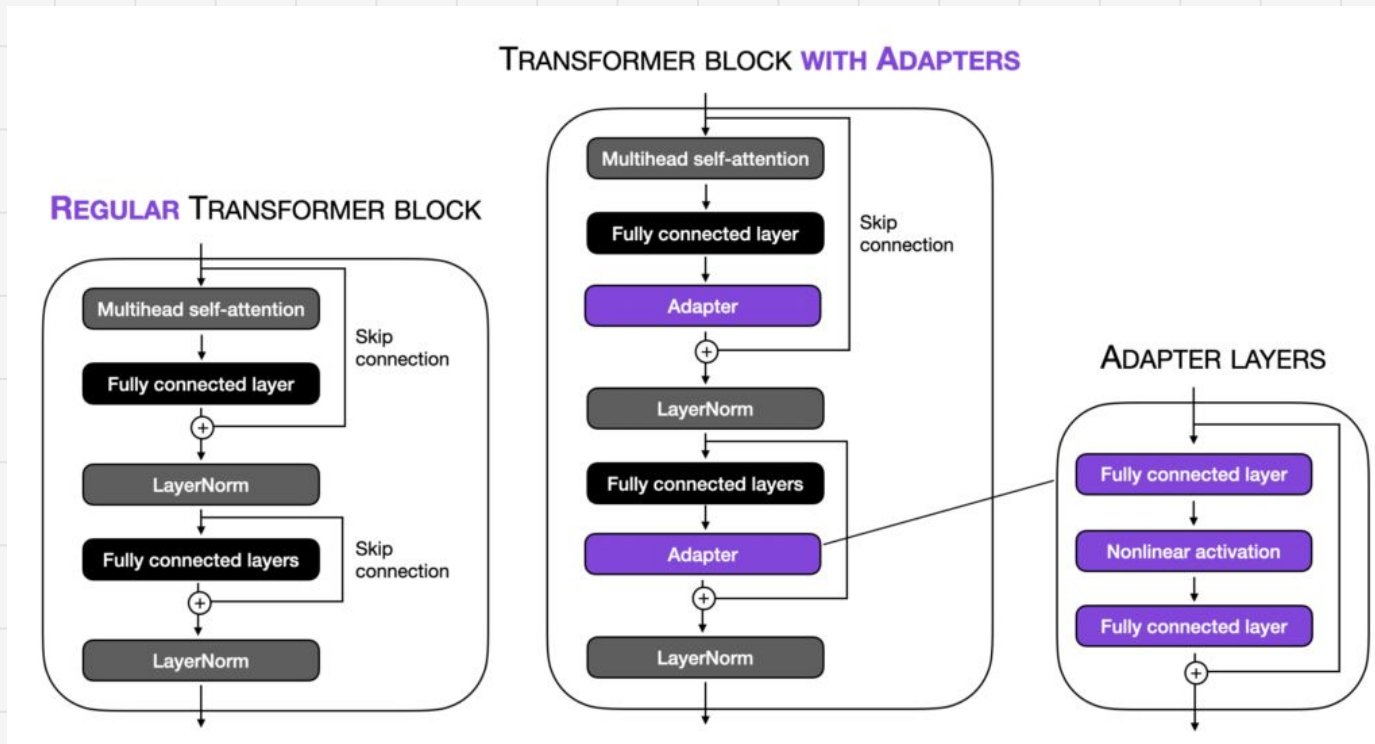
Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

```
1 Translate English to French: ← task description
2 sea otter => loutre de mer ← examples
3 peppermint => menthe poivrée ←
4 plush girafe => girafe peluche ←
5 cheese => ..... ← prompt
```

How to Fine-tuning

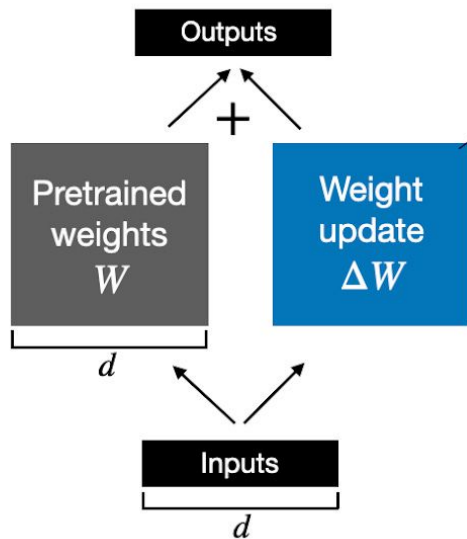
- **Adapter Layers (LLaMA-Adapte)**



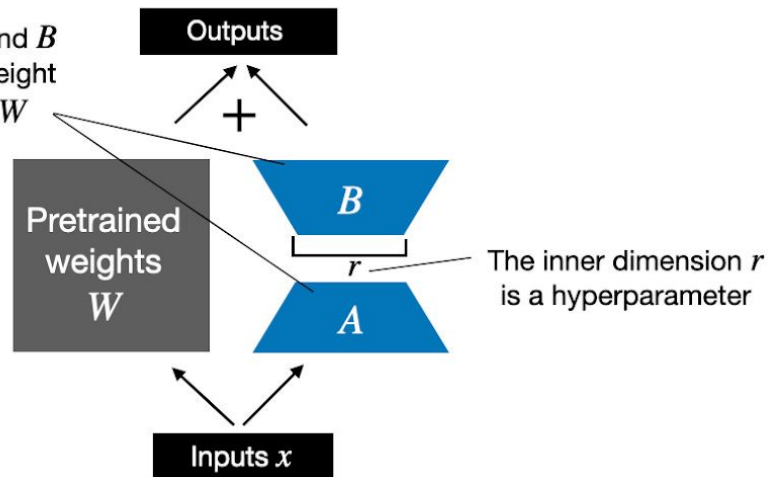
How to Fine-tuning

- LoRA

Weight update in **regular finetuning**



Weight update in **LoRA**



Hallucination

The model generates **fake or fabricated information** but is **delivered confidently**.

The generated content is **not coherent to reality**.

- **Why**

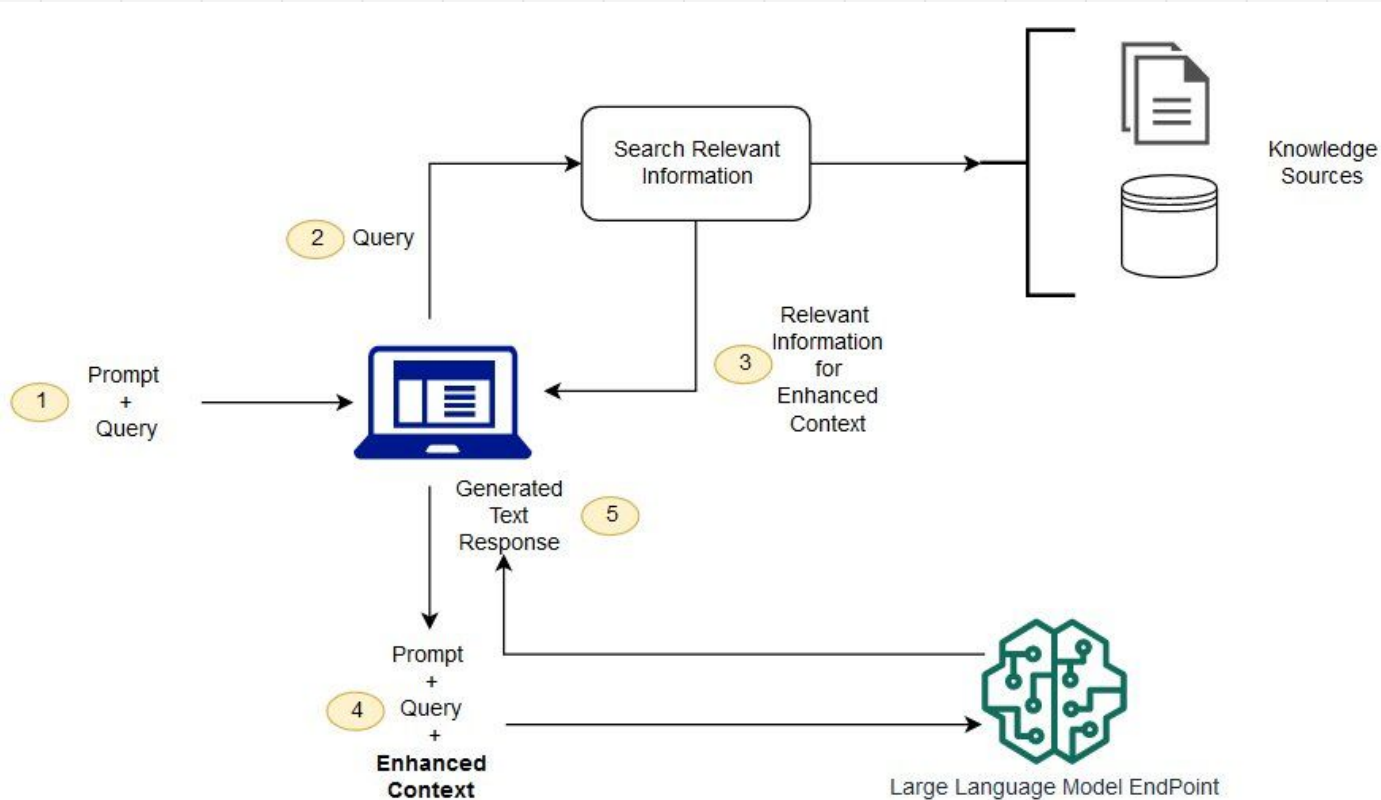
- Training on Noisy/Biased/Inaccurate/Outdated Data
- Training Objectives
 - Modern ML is more like a Pattern Recognition/probabilistic model.
 - It's not based on reasoning and not interact with real-world
- Context Length
 - While training, the sequence length of training samples is limited
 - While generating long content, the model tends to forget the past

- **Solution**

- Prompt Engineering
- Fine-Tuning
- **Integrate with external data - RAG**

Hallucination - RAG

RAG (Retriever-Augmented Generation)





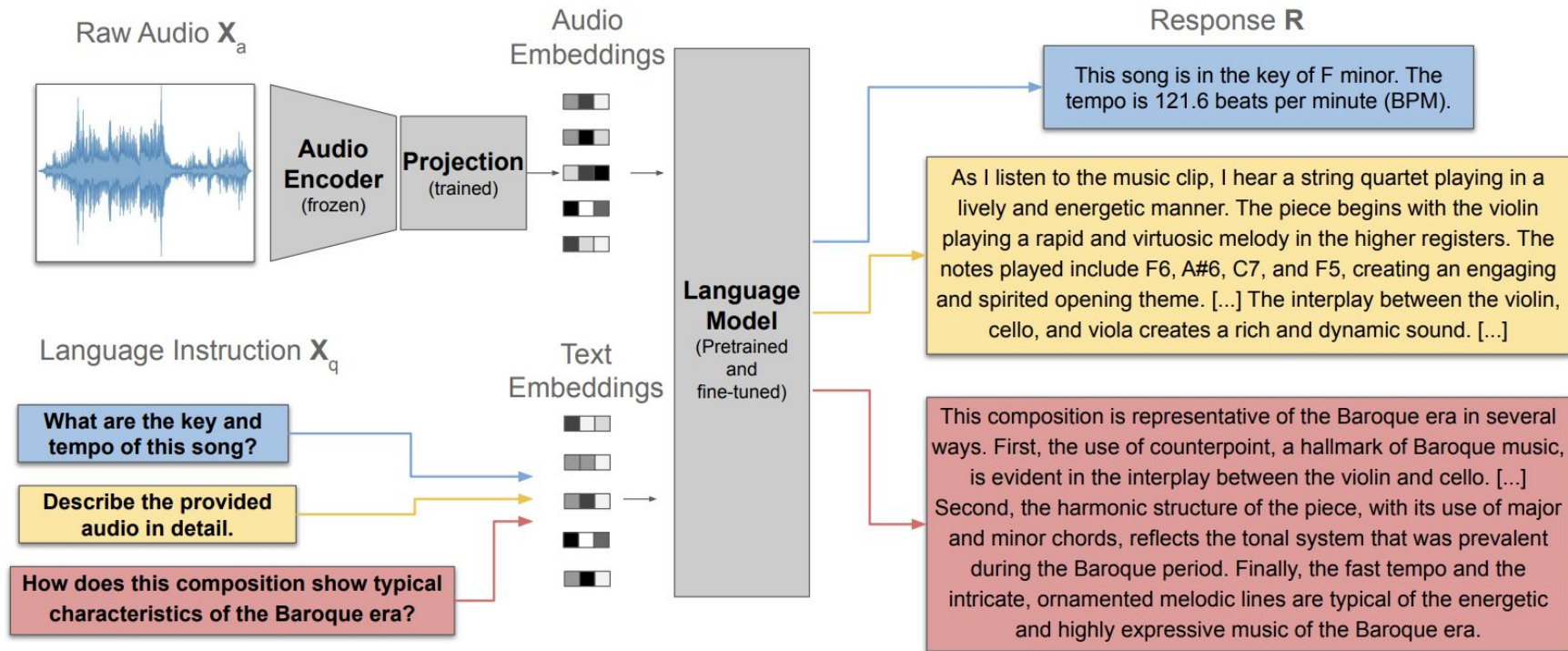
02

LLM & FM on Music

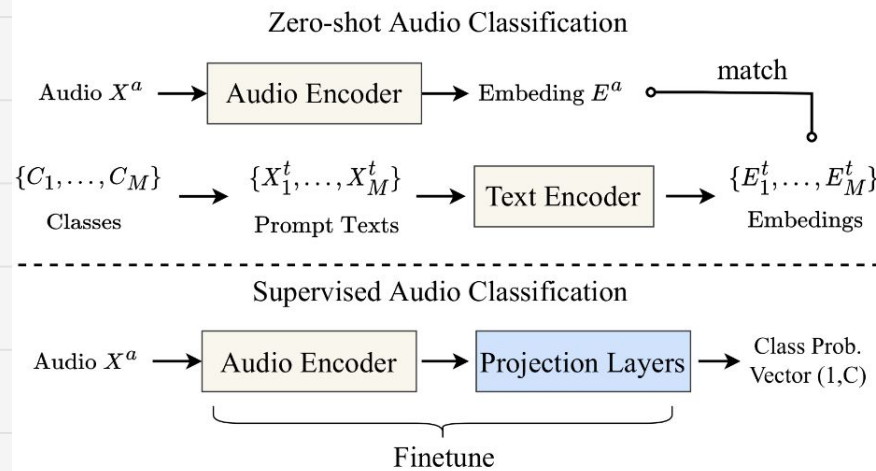
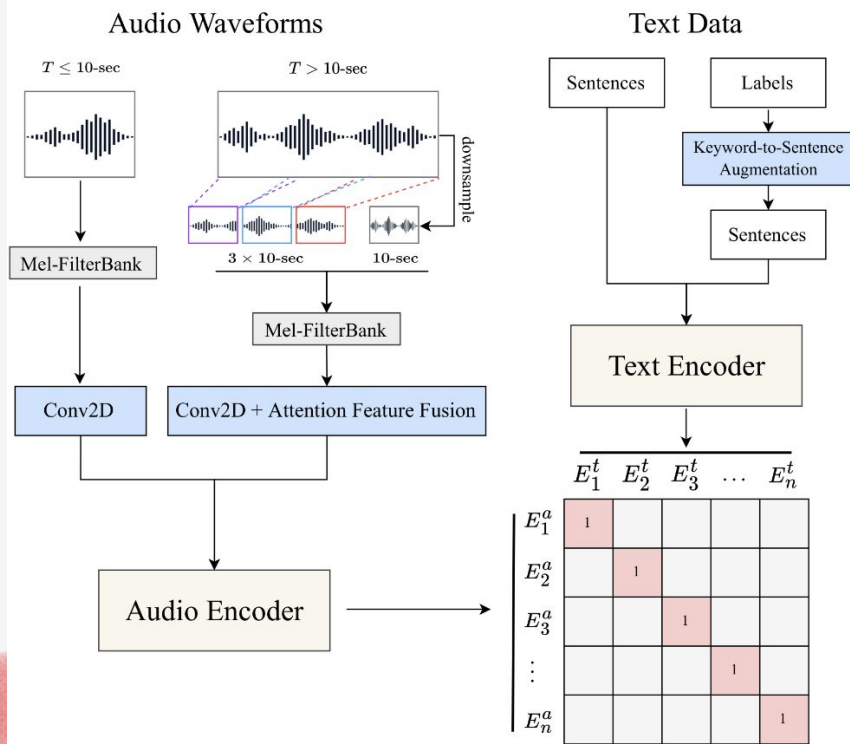
LLM on Music

- **LLark**
- **CLAP**
- **MERT**
- **MusicGen**
- **MT3**
- **Foundation Model Survey**

LLark (from Spotify)



CLAP (from LAION-AI)



$$\text{CLIPScore}(I, C) = \max(100 * \cos(E_I, E_C), 0)$$

Contrastive Learning

```
# image_encoder - ResNet or Vision Transformer
# text_encoder - CBOW or Text Transformer
# I[n, h, w, c] - minibatch of aligned images
# T[n, l] - minibatch of aligned texts
# W_i[d_i, d_e] - learned proj of image to embed
# W_t[d_t, d_e] - learned proj of text to embed
# t - learned temperature parameter

# extract feature representations of each modality
I_f = image_encoder(I) #[n, d_i]
T_f = text_encoder(T) #[n, d_t]

# joint multimodal embedding [n, d_e]
I_e = l2_normalize(np.dot(I_f, W_i), axis=1)
T_e = l2_normalize(np.dot(T_f, W_t), axis=1)

# scaled pairwise cosine similarities [n, n]
logits = np.dot(I_e, T_e.T) * np.exp(t)

# symmetric loss function
labels = np.arange(n)
loss_i = cross_entropy_loss(logits, labels, axis=0)
loss_t = cross_entropy_loss(logits, labels, axis=1)
loss = (loss_i + loss_t)/2
```

```
import torch
import torch.nn as nn
import torch.nn.functional as F
import numpy as np

from simple_clip.utils import get_feature_size

def contrastive_loss(logits):
    targets = torch.arange(logits.size(0)).to(logits.device)
    loss_images = F.cross_entropy(logits, targets)
    loss_texts = F.cross_entropy(logits.t(), targets)
    return (loss_images + loss_texts) / 2

def siglip_loss(logits):
    n = logits.size(0)
    # -1 for off-diagonals and 1 for diagonals
    labels = 2 * torch.eye(n, device=logits.device) - 1
    # pairwise sigmoid loss
    return -torch.sum(F.logsigmoid(labels * logits)) / n

class CLIP(torch.nn.Module):
    def __init__(self,
                 image_encoder,
                 text_encoder,
                 image_mlp_dim=False,
                 text_mlp_dim=768,
                 proj_dim=256,
                 init_tau=np.log(1.0),
                 init_b=0):
        super(CLIP, self).__init__()

        if not image_mlp_dim:
            image_mlp_dim = get_feature_size(image_encoder)

        self.image_encoder = image_encoder
        self.text_encoder = text_encoder

        self.image_projection = torch.nn.Sequential(
            torch.nn.Linear(image_mlp_dim, image_mlp_dim, bias=False),
            torch.nn.ReLU(),
            torch.nn.Linear(image_mlp_dim, proj_dim, bias=False))

        self.text_projection = torch.nn.Sequential(
            torch.nn.Linear(text_mlp_dim, text_mlp_dim, bias=False),
            torch.nn.ReLU(),
            torch.nn.Linear(text_mlp_dim, proj_dim, bias=False))

        self.t_prime = nn.Parameter(torch.ones(1) * init_tau)
        self.b = nn.Parameter(torch.ones(1) * init_b)

    def forward(self, image, input_ids, attention_mask):
        image_features = self.extract_image_features(image)
        text_features = self.extract_text_features(input_ids, attention_mask)
        image_features = F.normalize(image_features, p=2, dim=-1)
        text_features = F.normalize(text_features, p=2, dim=-1)
        return image_features @ text_features.t() * self.t_prime.exp() + self.b

    def extract_image_features(self, images):
        image_features = self.image_encoder(images)
        return self.image_projection(image_features)

    def extract_text_features(self, input_ids, attention_mask):
        text_features = self.text_encoder(input_ids, attention_mask)
        return self.text_projection(text_features)
```

MERT (from LAION-AI)

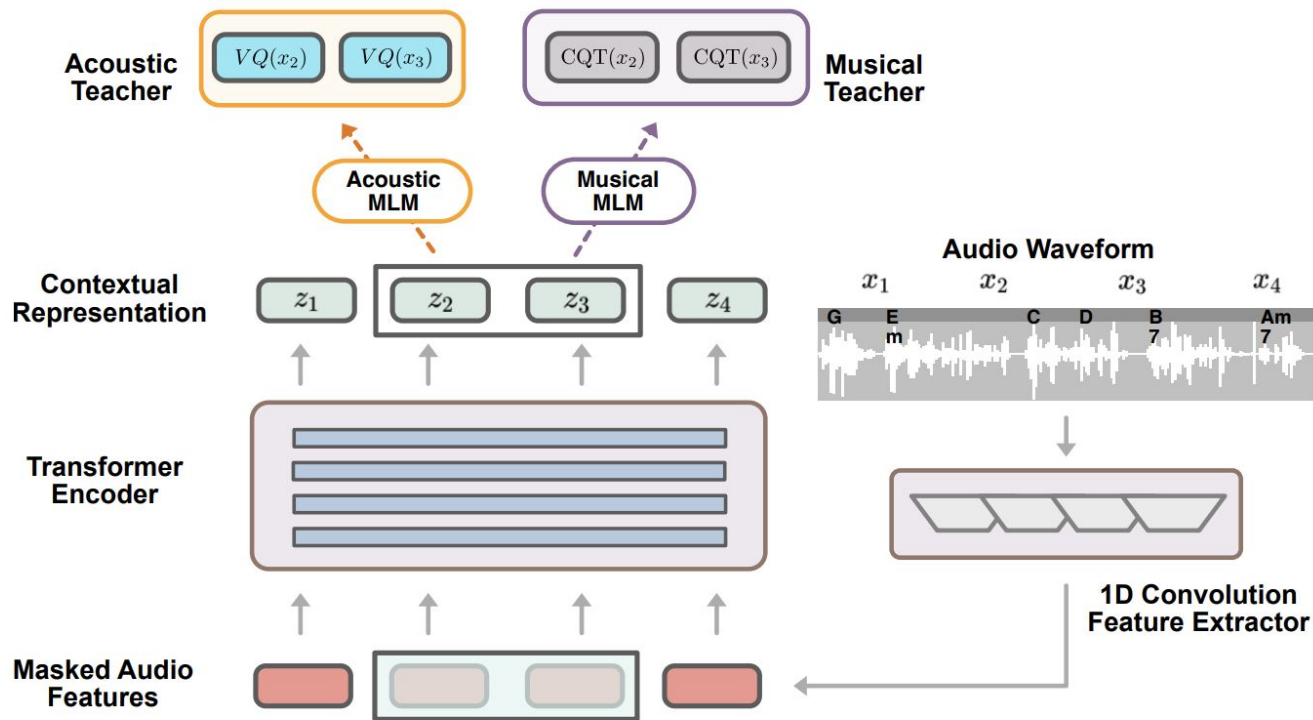
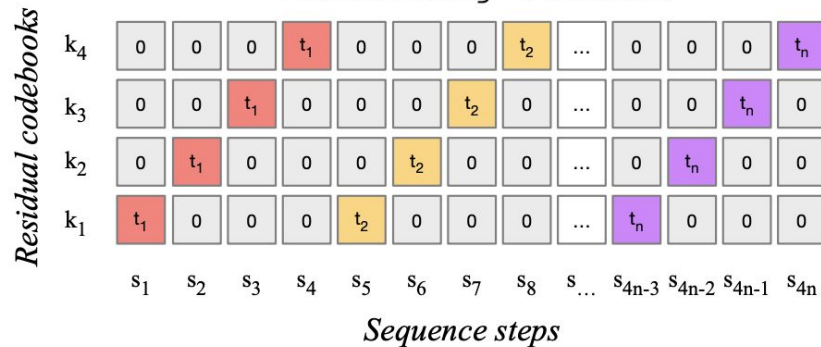


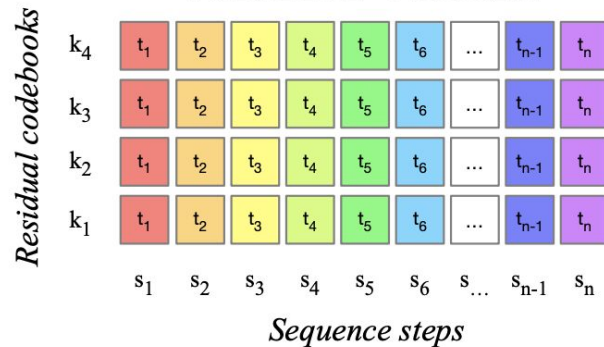
Figure 1: Illustration of the MERT Pre-training Framework.

MusicGen (from Meta)

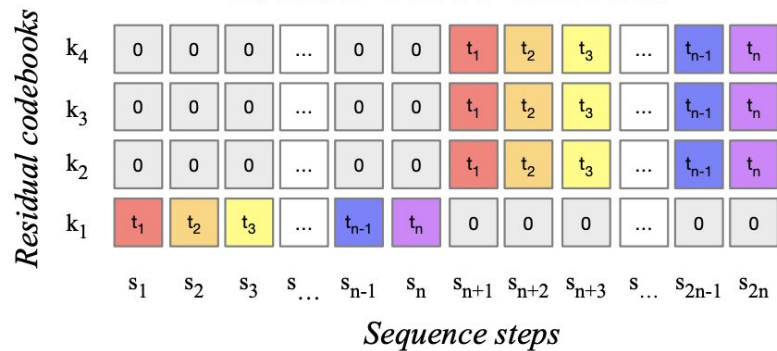
Flattening Pattern



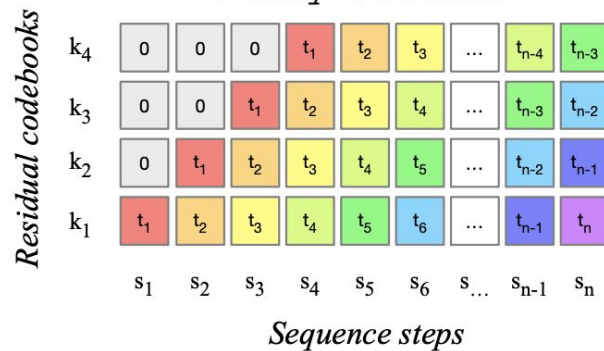
Parallel Pattern



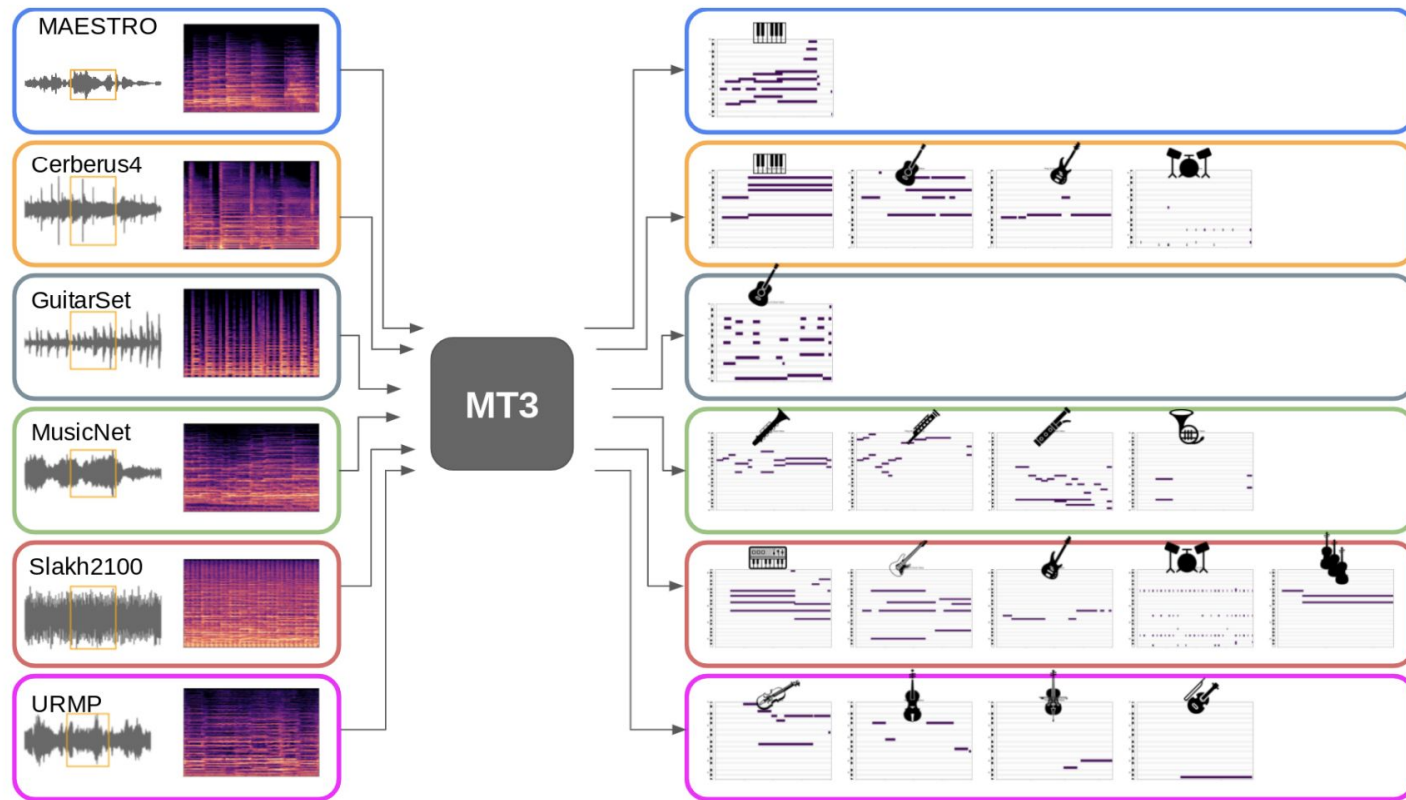
Coarse First Pattern



Delay Pattern

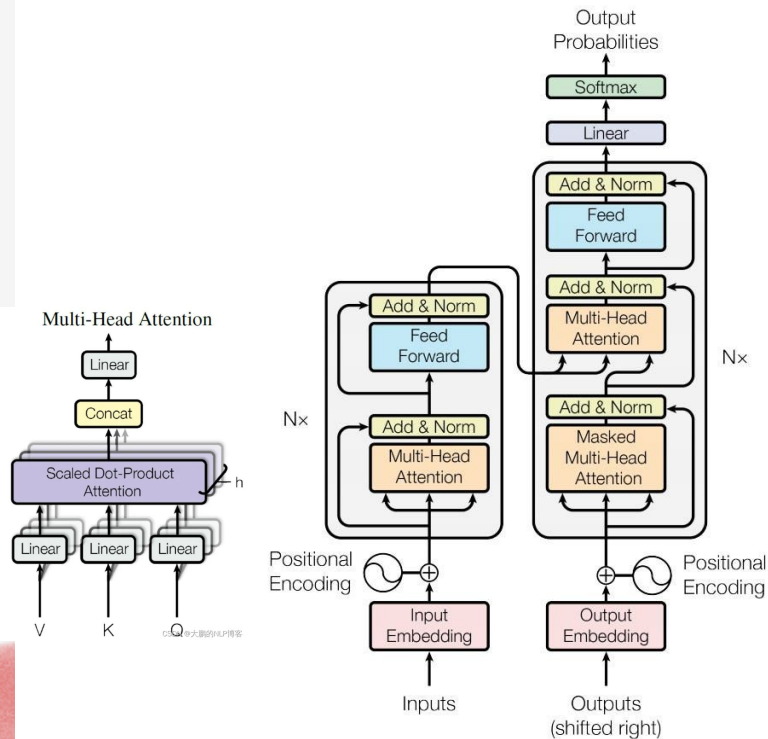


MT3 (from Google)



Idea: audio + prompt (instrument) -> corresponding score (midi, tab)

Transformer



Docoder Only: Masking Policy

[illegible]

Transformer

$$f_{t:t \in \{q,k,v\}}(\mathbf{x}_i, i) := \mathbf{W}_{t:t \in \{q,k,v\}}(\mathbf{x}_i + \mathbf{p}_i),$$

$$\begin{cases} \mathbf{p}_{i,2t} &= \sin(k/10000^{2t/d}) \\ \mathbf{p}_{i,2t+1} &= \cos(k/10000^{2t/d}) \end{cases}$$

Absolute

$$f_q(\mathbf{x}_m) := \mathbf{W}_q \mathbf{x}_m$$

$$\begin{aligned} f_k(\mathbf{x}_n, n) &:= \mathbf{W}_k(\mathbf{x}_n - \tilde{\mathbf{p}}_r^k) \\ f_v(\mathbf{x}_n, n) &:= \mathbf{W}_v(\mathbf{x}_n - \tilde{\mathbf{p}}_r^v) \end{aligned}$$

Relative

trainable

Positional Embedding

- Absolute
- Relative (pairwise)
- Rope = Abs + Rel
- Alibi = Rope + Extrapolatio

absolute

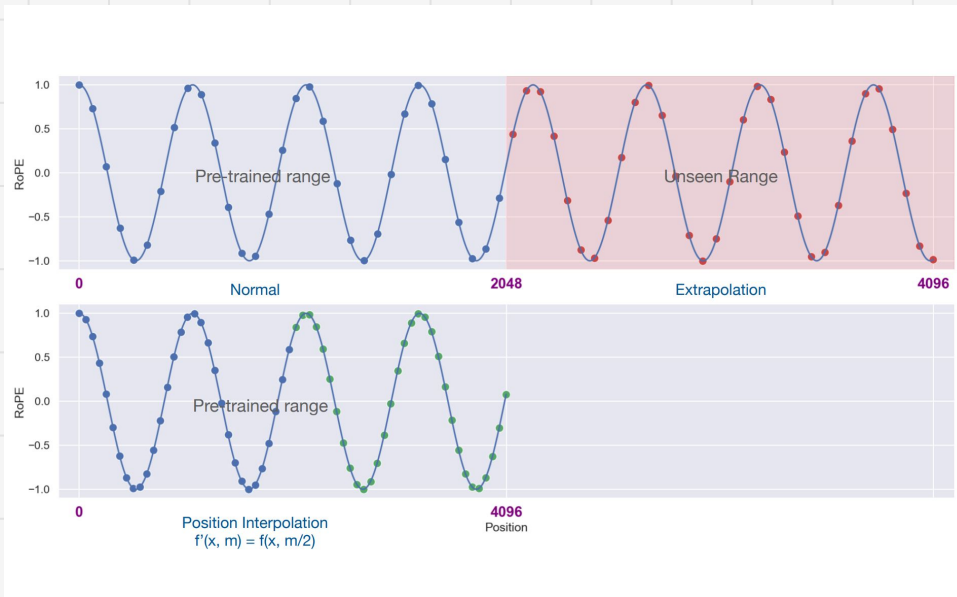
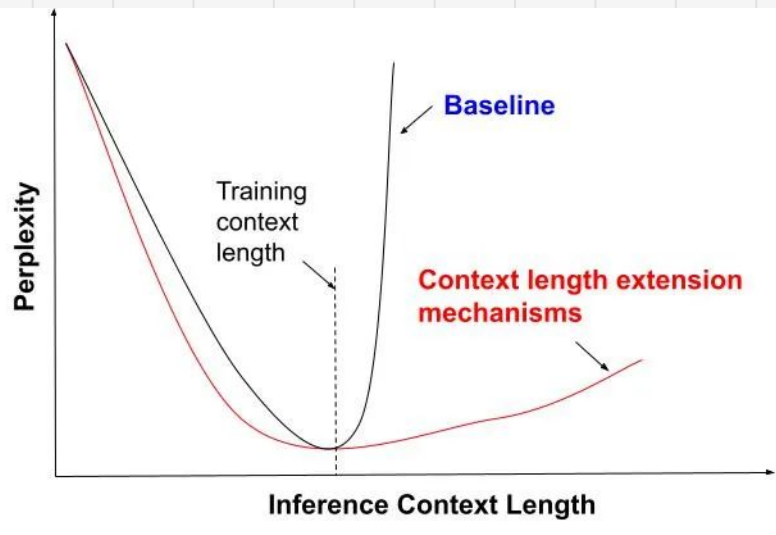
(ex: Vanilla Transformer)

for pair-wise

(ex: Transformer-XL)

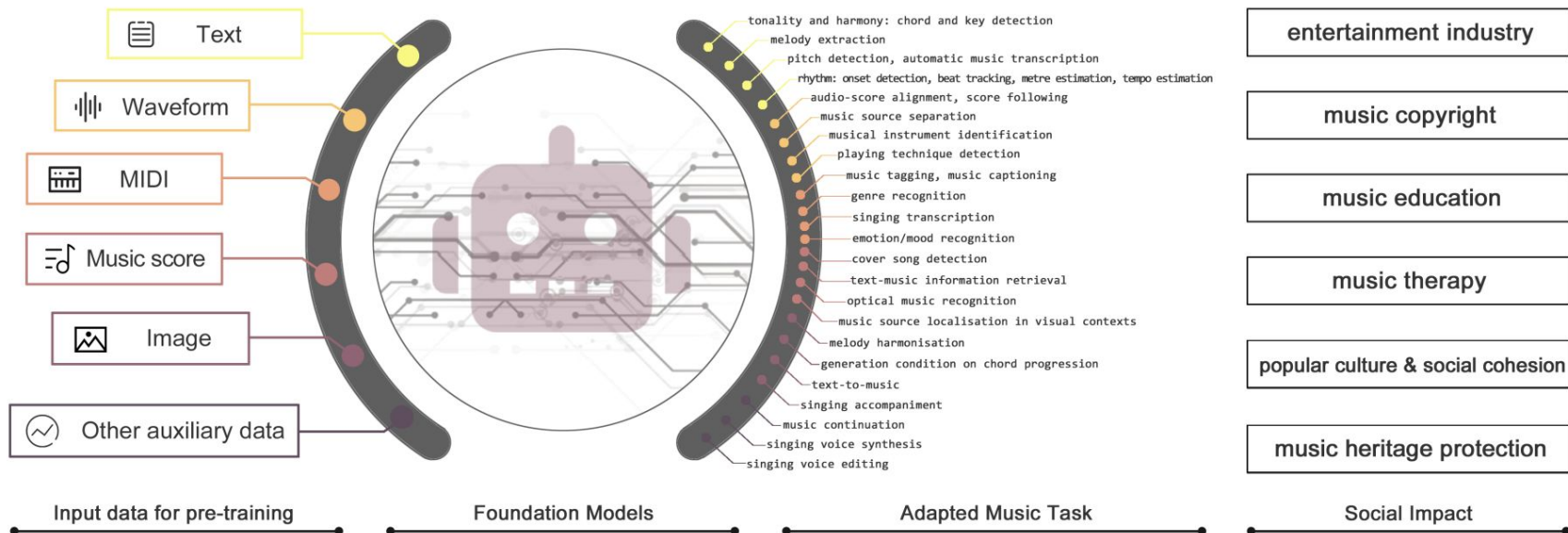
for pair-wise and absolute (ex: LLama, ChatGLM)

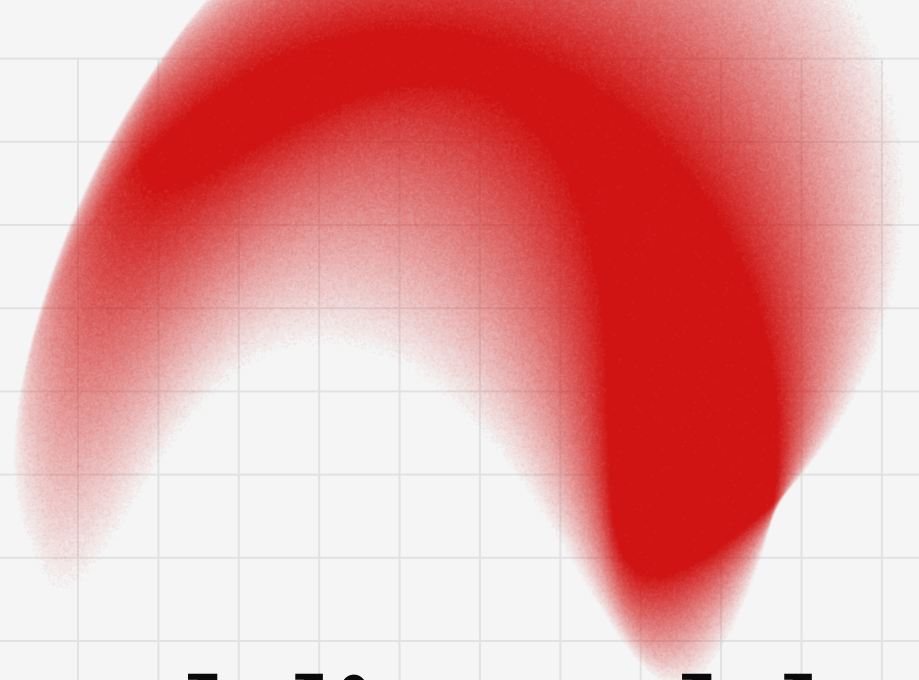
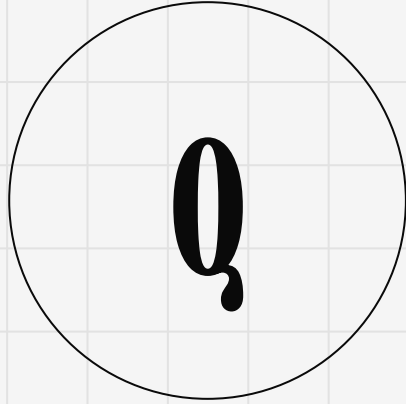
for context extending



Foundation Model Survey (latest)

[Link](#) - [arxiv.2408](#)





Design Multi-Modality Model

SonyCSL

Q: Designing Multi-Modal Models Combining **Audio** and **Symbolic** Representations

Goal : Propose a framework for a generative model that simultaneously learns from both **audio and symbolic** representations of music to **improve generation quality**.

A: Similar to my work - MusicConGen (ISIMIR'2024)

What are the qualities we want to improve with symbolic data?

- Controllability

What is the Symbolic Description?

Human understand music with notations and the conceptualized **informations**:

- BPM
- Meter
- Lead Sheet
 - Key
 - Chord
 - Melody
- Arrangement
- Structure
- MIDI
- Sheet Music
 - Staff and Tablature
- Genre
- Description (Autotagging)

Metrics for Evaluation

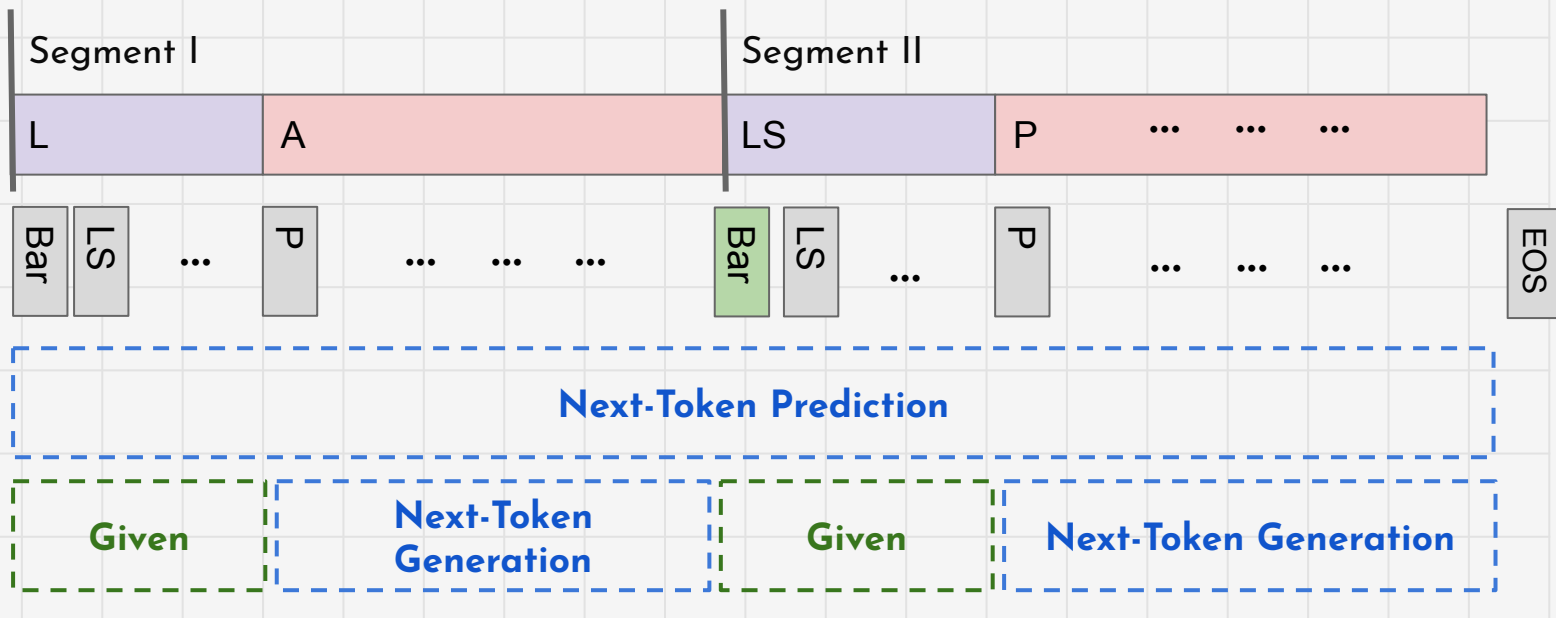
MIR eval Toolkit

- F-measure for BPM and Meter
- Chords
- FAD for audio quality

Combination of Different Dataset

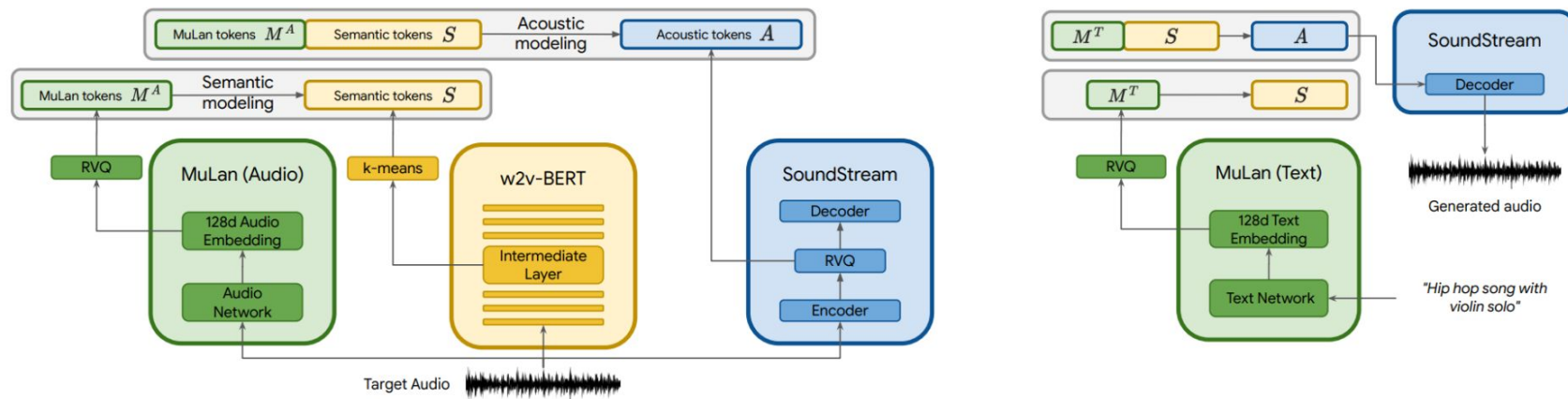
Method I:

- **Conditional Generation**, with decoder only (GPT-like) transformer
- **T5 Prefix-LM Mechanism**
 - Condition: Lead Sheet (L)
 - Generation: Acoustic Token (A) - can be generalized to multi-track



Method II: Hierarchical LLM

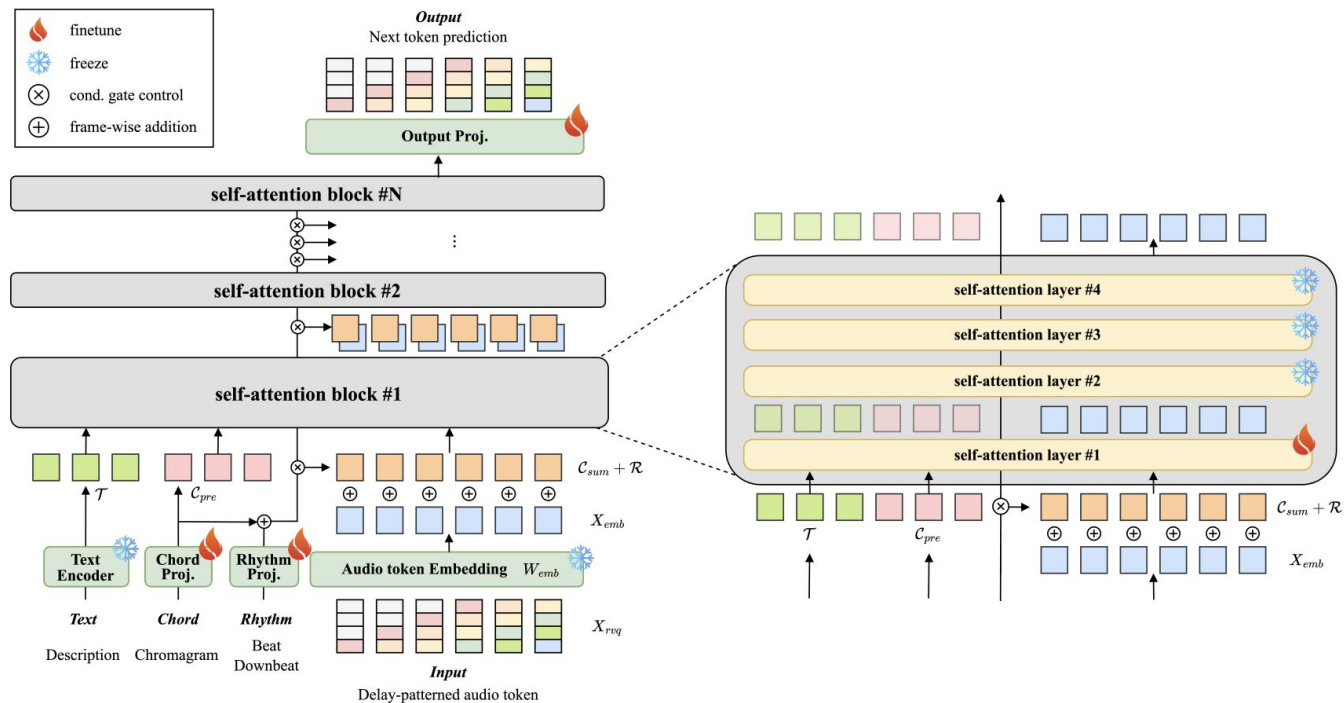
MusicLM: Generating Music From Text



Lead sheet -> Multi-track MIDI

Method III:

- Replacement of Positional/Sentence Embedding

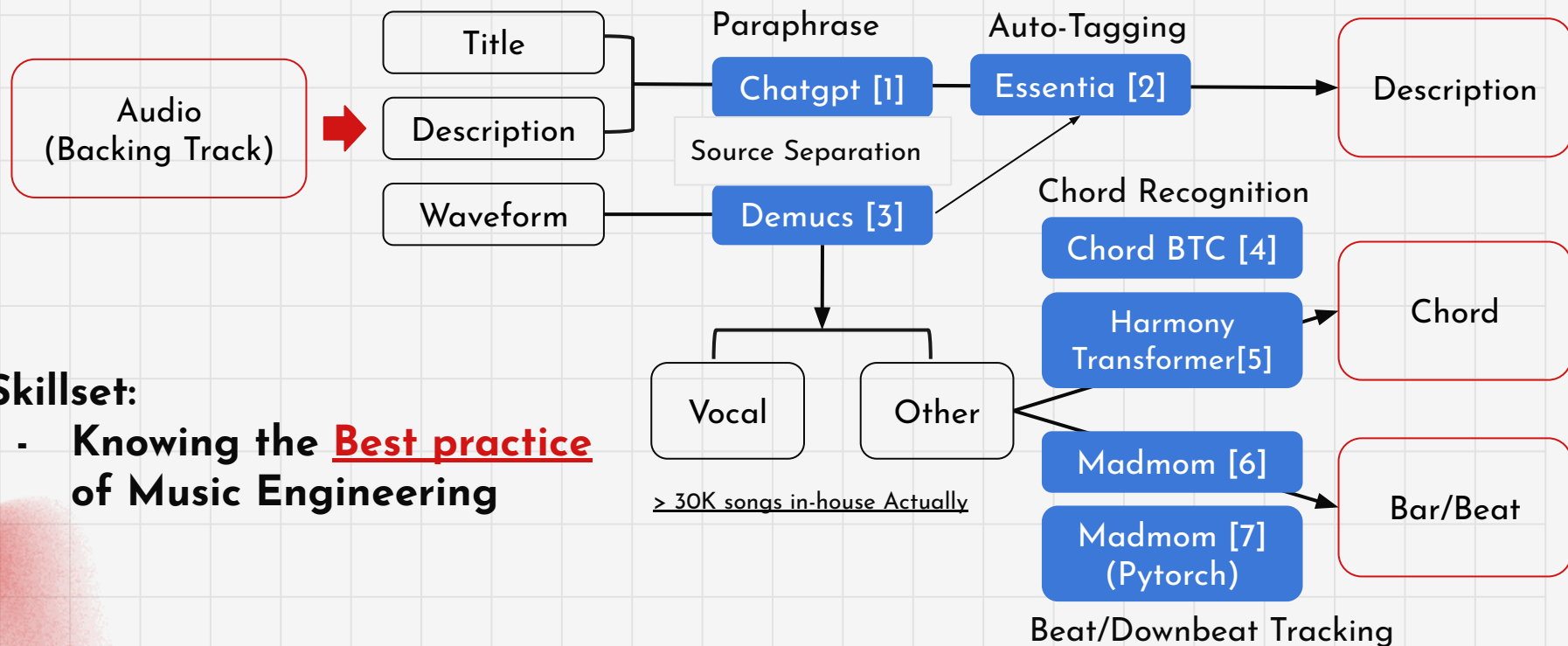


(a) MusiConGen model structure

(b) self-attention block

Dataset Building

- Pipeline from my work - MusicConGen (ISMIR'24)



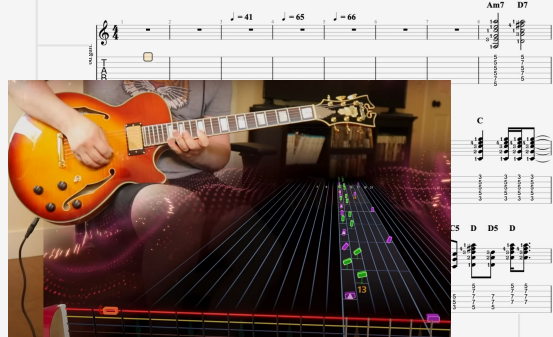
Skillset:

- Knowing the Best practice of Music Engineering

Dataset Building

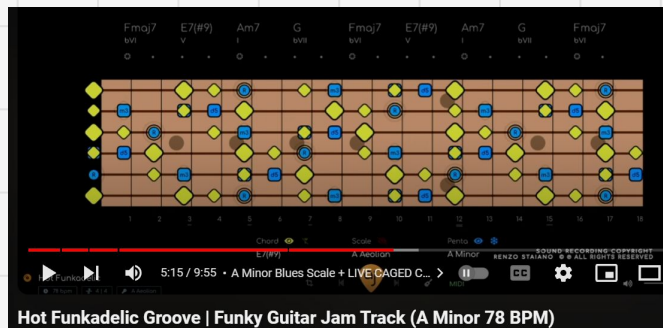
Highlights of My Inhouse Collection:

1. Data from Guitar Gaming Community



- Aligned audio and tab
- Finger position
- Chord label
- Over 1K songs
- Multi-track guitar
- Tab Generation
- Transcription

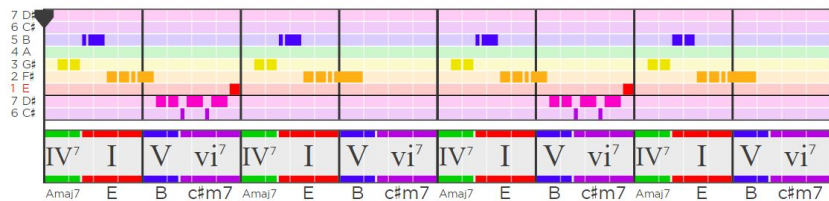
3. Backing Tracks



Skillset:

- Web Crawling, Data Cleaning
- Musicology

2. Lead Sheet from theorytab (108 stars)



- Over 30k songs
- Our backbone dataset of text2music model
- Description
- Key
- BPM
- Chord Progression
- High Quality after Curation (TODO)
 - > Excellent Resources for any task!