

CTP431: Fundamentals of Computer Music

Overview of Music AI: Part 2

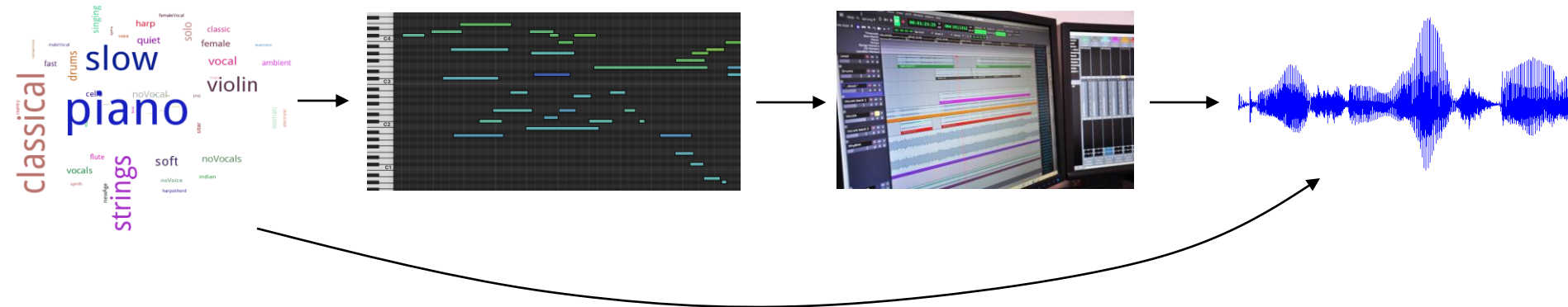


Graduate School of
Culture Technology

Juhan Nam

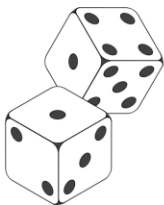
AI for Music Creation

- Symbolic Music Generation
- Sound Synthesis and Digital Audio Effect
- Audio-Level Music Generation



Musical Dice Game (1787)

- Mozart



Erster Theil.
Premiere Partie.

	A	B	C	D	E	F	G	H
2	20	26	32	38	44	50	56	62
3	24	30	36	42	48	54	60	66
4	28	34	40	46	52	58	64	70
5	32	38	44	50	56	62	68	74
6	36	42	48	54	60	66	72	78
7	40	46	52	58	64	70	76	82
8	44	50	56	62	68	74	80	86
9	48	54	60	66	72	78	84	90
10	52	58	64	70	76	82	88	94
11	56	62	68	74	80	86	92	98
12	60	66	72	78	84	90	96	102

Zweiter Theil.
Secunde Partie.

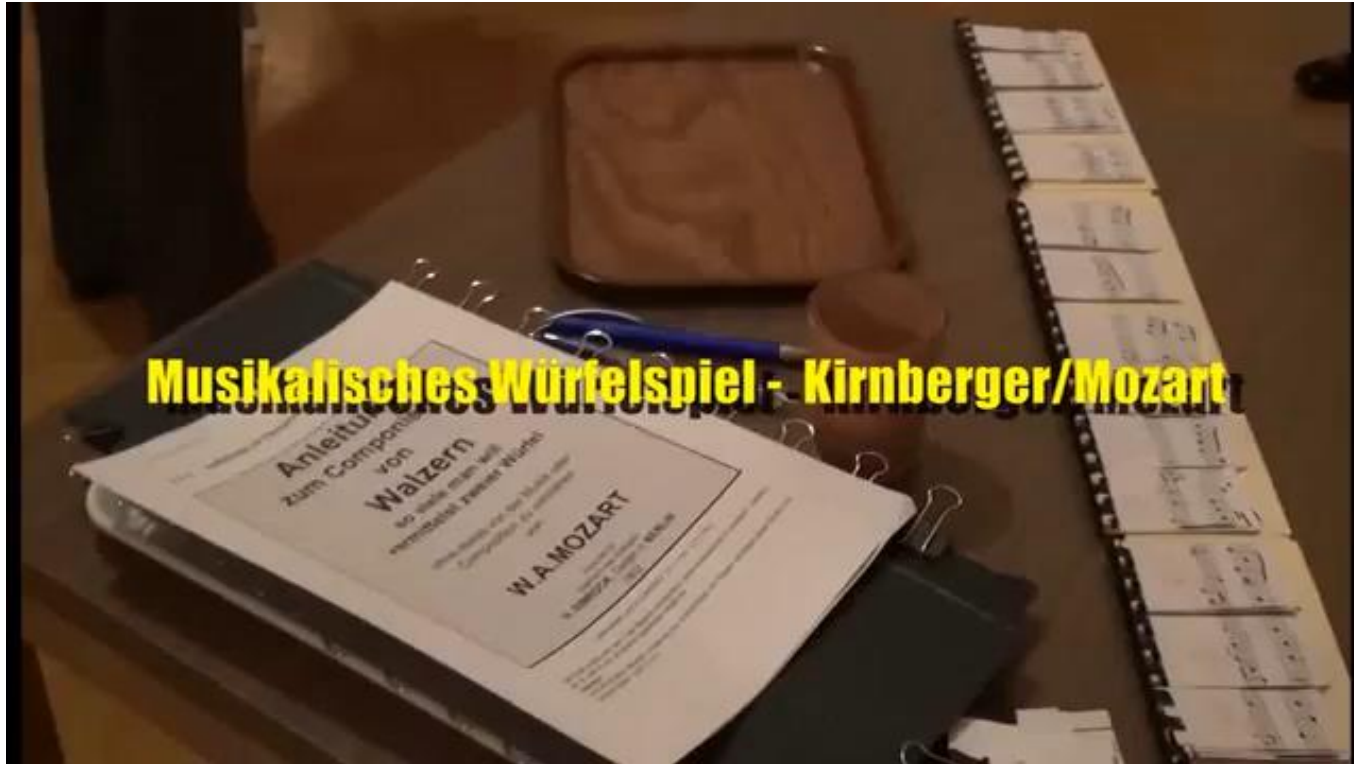
	A	B	C	D	E	F	G	H
2	70	76	82	88	94	100	106	112
3	74	80	86	92	98	104	110	116
4	78	84	90	96	102	108	114	120
5	82	88	94	100	106	112	118	124
6	86	92	98	104	110	116	122	128
7	90	96	102	108	114	120	126	132
8	94	100	106	112	118	124	130	136
9	98	104	110	116	122	128	134	140
10	102	108	114	120	126	132	138	144
11	106	112	118	124	130	136	142	148
12	110	116	122	128	134	140	146	152

TABLE de MUSIQUE. 5.

A musical score for a piano piece titled 'TABLE de MUSIQUE. 5.' The score is written for two staves (treble and bass clef) and consists of 32 measures. The music is in a 3/4 time signature and features a variety of rhythmic patterns, including eighth and sixteenth notes, and rests. The score is divided into two parts, with the first part ending at measure 16 and the second part starting at measure 17. The notation includes dynamic markings such as 'f' and 'p', and articulation marks like slurs and accents.

Mozart K. 516F

Musical Dice Game (1787)



Illiac Suite (1956)

- Lejaren Hiller and Leonard Isaacson's String Quartet
 - Markov model

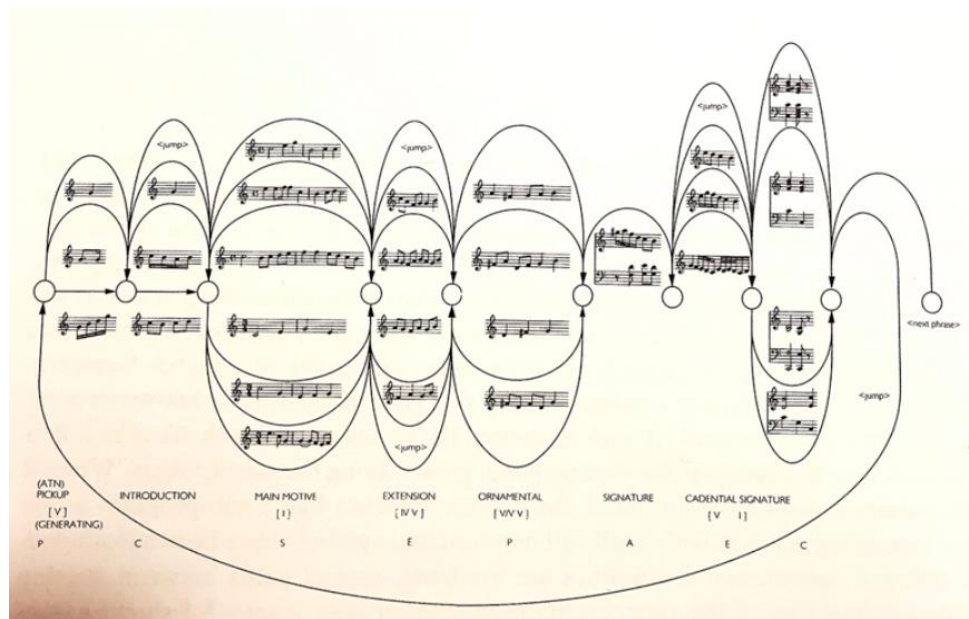
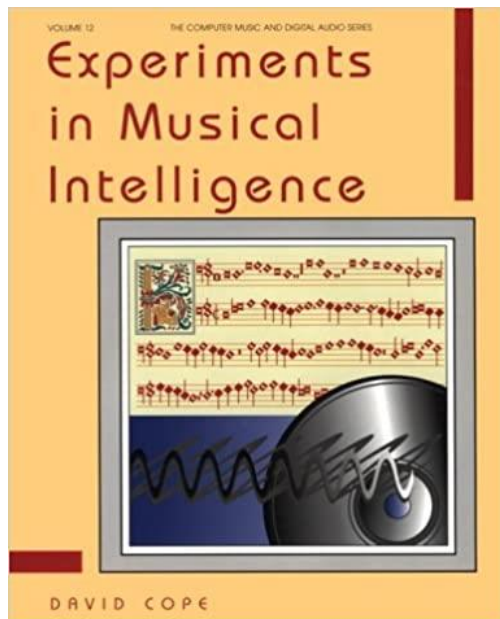


Lejaren Hiller - *Illiac Suite for String Quartet* (1956)

First experiment: presto, andante, allegro

Experiment in Music Intelligence (1980s)

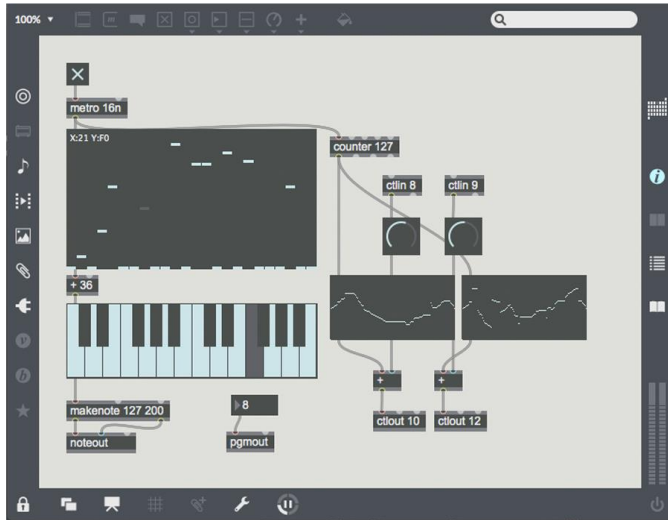
- David Cope
 - A rule-based natural language processing technique



Augmented Transition Network



Algorithmic Composition



MAX

The screenshot shows the SuperCollider software interface. On the left, a code patch is visible:

```
93 freqDevKnob.value = devSpec.unmap(devSpec.default);
94
95 // Frequency Deviation Number
96 freqDevNumber = StaticText.new(40, Rect(560, 210, 80, 255));
97 freqDevNumber.background = defaultFillColor;
98 freqDevNumber.alpha = defaultAlpha;
99 freqDevNumber.align = 'center';
100 freqDevNumber.string = devSpec.default;
101 freqDevNumber.font = defaultFont;
102 freqDevNumber.stringColor = defaultStringColor;
103
104 // Frequency Deviation Label
105 freqDevLabel = StaticText.new(40, Rect(500, 240, 200, 255));
106 freqDevLabel.string = "Frequency Deviation";
107 freqDevLabel.align = 'center';
108 freqDevLabel.font = defaultFont;
109 freqDevLabel.stringColor = defaultStringColor;
110
111 // Volume Slider
112 volumeSlider = EZSlider;
113 parent: sin;
114 bounds: Rect(730, 20, 70, 230);
115 label: 'VOLUME';
116 controlSpec: ControlSpec(-80, 0, 110, 0.1, -40, "dB");
117 action: {self synth.set(amp, $z.value.dbamp)};
118 labelWidth: 80;
119 width: 30;
120 layout: 'vert';
121
122 setColor4;
123 stringColor: defaultStringColor;
124 sliderBackground: Color.grey(0.9);
125 numberColor: Color.black;
126 font = Font("Verdana", 14, bold: true);
127 volumeSlider.numView.align = 'center';
128 volumeSlider.numView.alpha = 'center';
129
130 {
131   SynthDef("freq-mod", {
132     arg carrier = 440, modFreq = 5, freqDev = 20, amp = 0.01;
133     var carrier, modulator;
134     modulator = SinOsc.ar(freq: modFreq, mul: freqDev);
135     carrier = SinOsc.ar(freq: carrier, mul: modulator, mul: amp);
136     Out.ar(2, [carrier, carrier]);
137   });
138
139   s.sync;
140
141   synth = Synth("freq-mod");
142
143   }.fork;
144
145   CmdPeriod.doOnce(4000, {
146     // end of waitForBoot
147   });
148 }
```

On the right, a graphical user interface (GUI) is displayed with three red rotary knobs labeled "Carrier Frequency" (440), "Modulator Frequency" (5), and "Frequency Deviation" (20). A volume slider is also present, labeled "VOLUME" and "db". Below the GUI is a "Freq Analyzer" window showing a frequency spectrum plot.

On the far right, a browser window displays the "Disintegrator" class page, which includes a description and class methods:

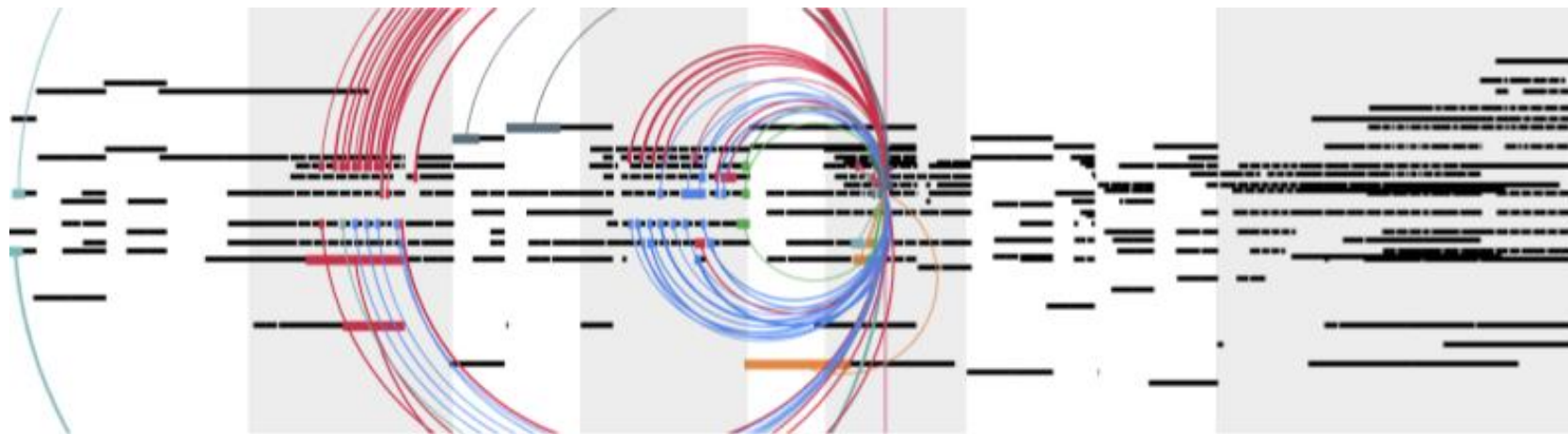
Description
Amplifies random half-cycles of its input by multiplier. Set multiplier to 0 and very probability for a wam-like effect, or set multiplier to 1 and probability to 0.5 to turn pitched sounds into noise.

Class Methods
*ar (in, probability: 0.5, multiplier: 0, mul: 1, add: 0)
Arguments:
in The input signal.
probability multiplier
mul Output will be multiplied by this value.
add This value will be added to the output.

SuperCollider

Google Music Transformer (2018)

- Music Language Model



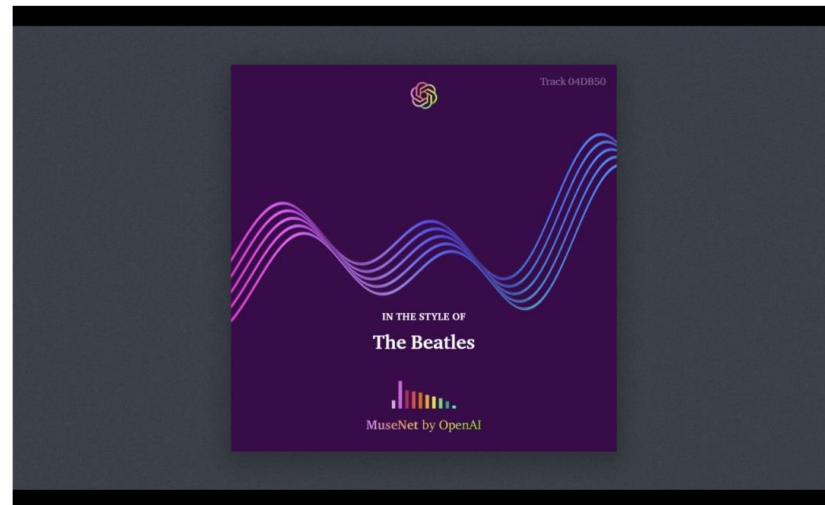
OpenAI MuseNet (2019)

MuseNet

We've created MuseNet, a deep neural network that can generate 4-minute musical compositions with 10 different instruments, and can combine styles from country to Mozart to the Beatles.



The screenshot shows the OpenAI MuseNet web interface. On the left is a dark sidebar with a list of styles: CHOPIN, MOZART, RACHMANINOFF, LADY GAGA, COUNTRY, DISNEY, JAZZ, BACH, BEETHOVEN, JOURNEY, THE BEATLES, VIDEO GAMES, BROADWAY, FRANK SINATRA, BLUEGRASS, and TCHAIKOVSKY. The main panel has a light gray background. At the top, 'STYLE' is set to 'CHOPIN' and 'INTRO' is set to 'BEETHOVEN'S FÜR ELISE'. Below that, 'INSTRUMENTS' are listed as PIANO, STRINGS, WINDS, DRUMS, HARP, GUITAR, and BASS. A 'NUMBER OF TOKENS' slider is set to 225. A 'HIDE ADVANCED SETTINGS' link is visible. The central part of the interface features a piano roll visualization with blue horizontal bars on a black background. At the bottom, there are controls: a play button, 'STOP PLAYBACK', 'DOWNLOAD', 'TWEET', and 'RESET'.



The screenshot shows a generated album cover for a track titled 'Track 04DB50'. The cover has a dark purple background with a glowing blue and purple wave pattern. At the top center is the OpenAI logo. Below the wave, the text reads 'IN THE STYLE OF The Beatles'. At the bottom, there is a small bar chart and the text 'MuseNet by OpenAI'.

MuseNet played an experimental concert on April 25th, 2019, livestreamed on OpenAI's [Twitch channel](#), in which no human (including us) had heard the pieces before.

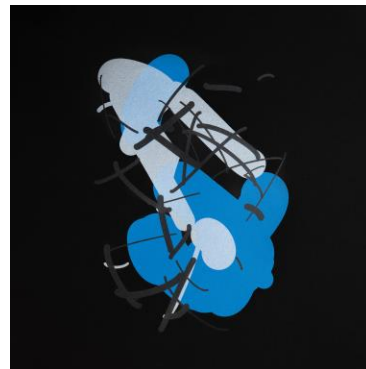


Magenta + YACHT (2019)

WHAT IS MAGENTA?

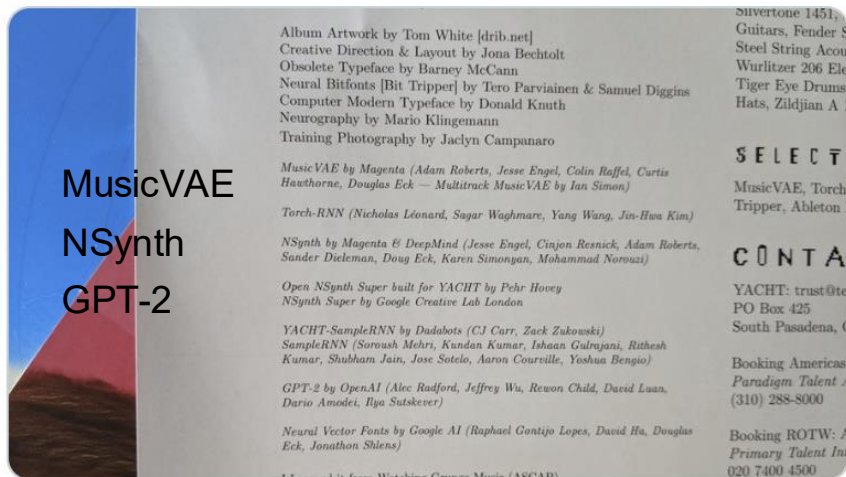
An open source research project exploring the role of machine learning as a tool in the creative process.

Chain Tripping



Adam Roberts @ada_rob · Sep 5

More album liner notes should include paper citations.



MusicVAE
NSynth
GPT-2



<https://yacht.bandcamp.com/album/chain-tripping>

Grammy-Nominated

AI Song Contest (2020 – 2022)



The AI Song Contest is an international competition exploring the use of AI in the songwriting process. Teams consisting of musicians, researchers, data scientists, developers and anyone else working on or interested in the combination of music and AI create a song using AI as part of their songwriting process.

2021 organizing team

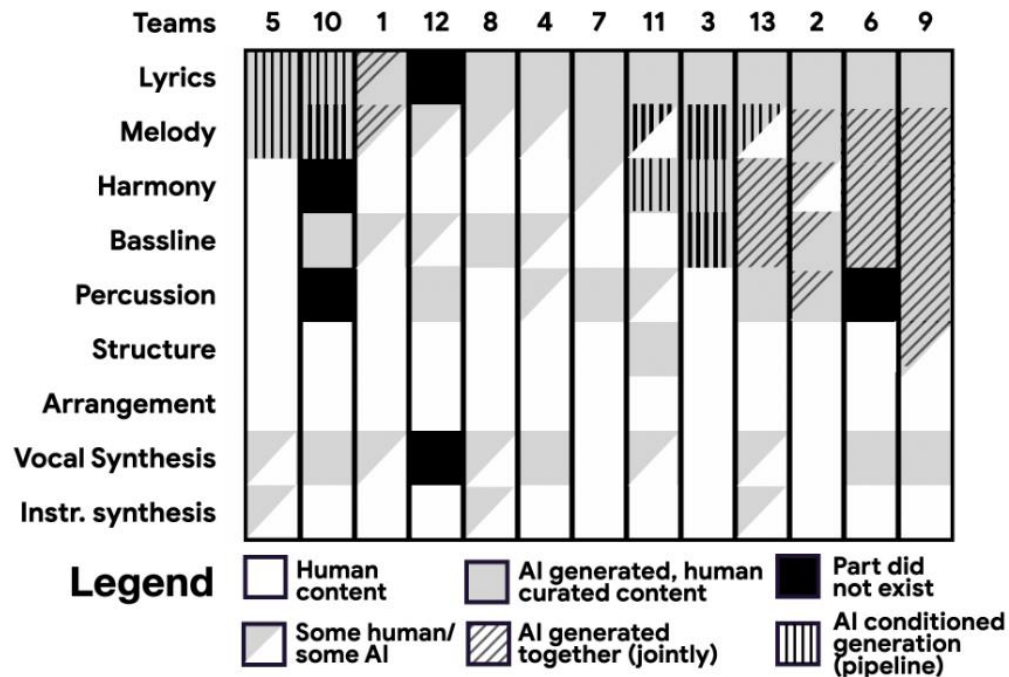
The 2021 contest was organised by Karen van Dijk, Rebecca Leger, Anna Huang, Ryan Groves, Rujing “Stacy” Huang, Hendrik Vincent Koops, Ashley Burgoyne, and Carol E. Reiley.

AI Song Contest 2020

Music building blocks	Models & techniques
Lyrics	GPT2, LSTM, Transformer
Melody	CharRNN, SampleRNN, LSTM + CNN, WaveNet + LSTM, GAN, Markov model
Harmony	LSTM, RNN autoencoder, GAN, Markov model
Bassline	LSTM + CNN, WaveNet + LSTM, GAN
Drums	DrumRNN, Neural Drum Machine, SampleRNN, Markov model
Multi-part	MusicVAE trio (melody, bass, drums), MiniVAE trio, Coconet/Coucou (4-part counterpoint), MusicAutobot (melody, accompaniment), Transformer (full arrangement)
Structure	Markov model
Vocal synthesis	WaveNet, SampleRNN, Vocaloid, Sinsy, Mellotron, Emvoice, Vocaloid, custom vocal assistant
Instrument synthesis	SampleRNN, WaveGAN, DDSP

“AI Song Contest: Human-AI Co-Creation in Songwriting”, Cheng-Zhi Anna Huang et al., ISMIR 2020

AI Song Contest (2020-2022)



“AI Song Contest: Human-AI Co-Creation in Songwriting”, Cheng-Zhi Anna Huang et al., ISMIR 2020

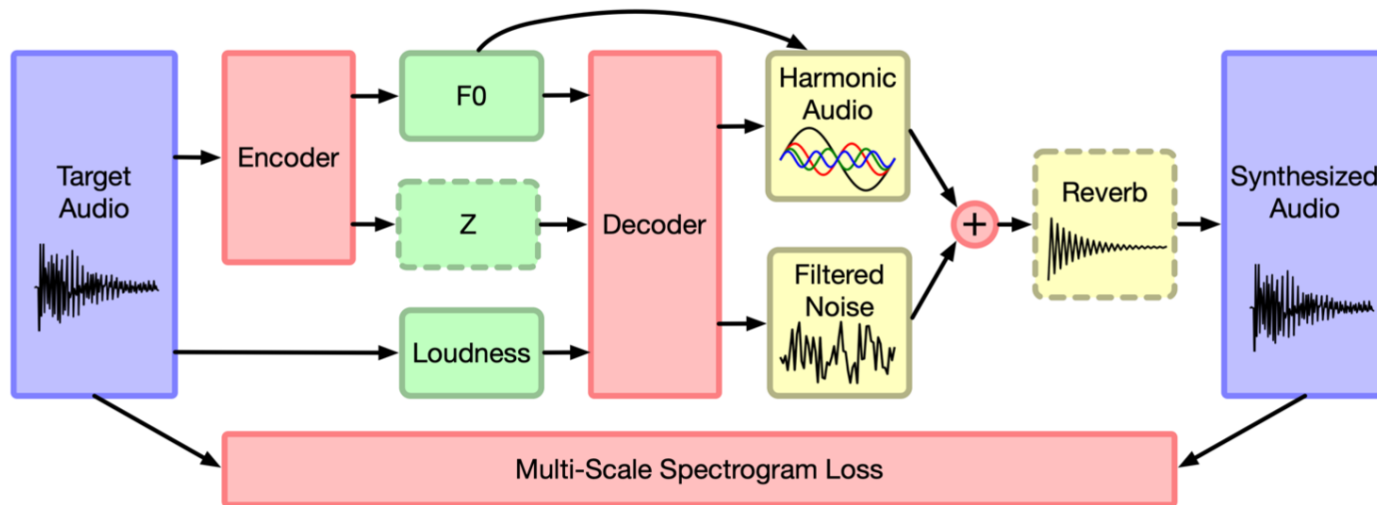
Tone Transfer

- Change timbre of the input source while preserving the expressions



Differentiable Digital Signal Processing (DDSP)

- A hybrid model of neural network and spectral modeling synthesis



DDSP-VST

DDSP captures the small nuances of your playing

<https://magenta.tensorflow.org/ddsp-vst>

AI Song Contest 2022 Winner

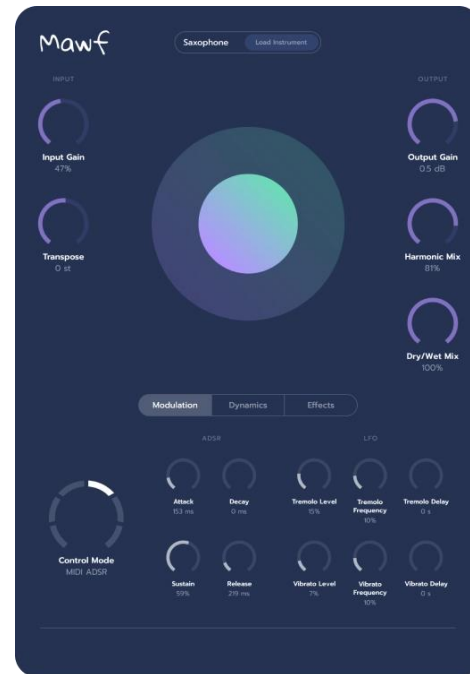
- Lamtharn “Hanoi” Hantrakul



Original Phi

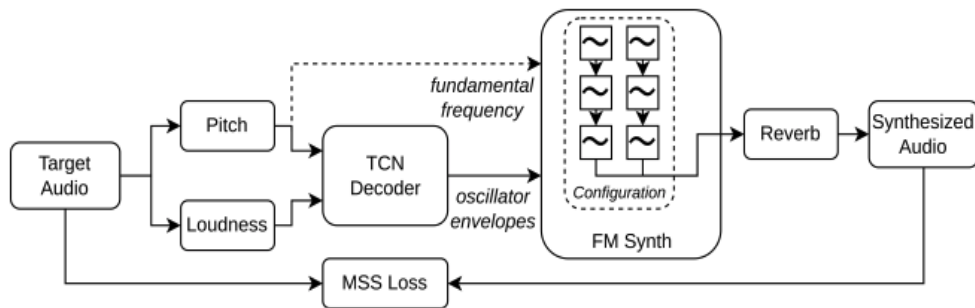


มาสังเคราะห์เป็นเสียงดนตรีใหม่ๆ
ที่ออกนอกกรอบเสียงดนตรีไทย



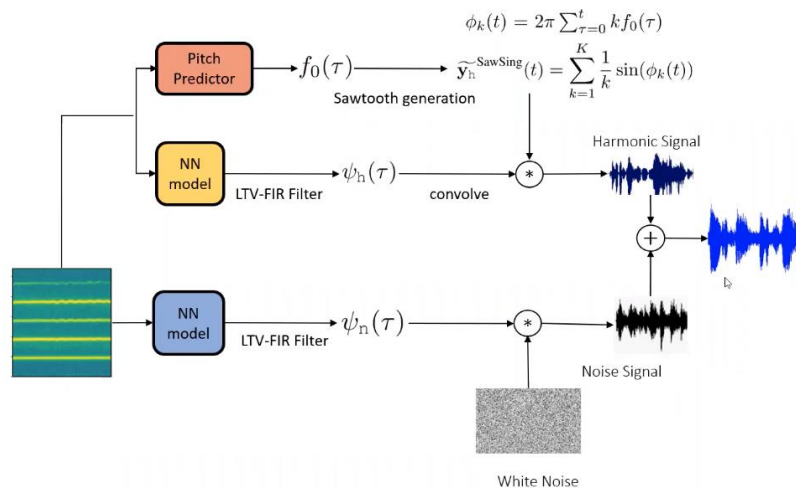
ByteDance Mawf

Neural Versions of Traditional Sound Synthesis Techniques



DDX7: Differential FM Synthesis

<https://fcaspe.github.io/ddx7/>



SawSing: Differential Subtractive Synthesis

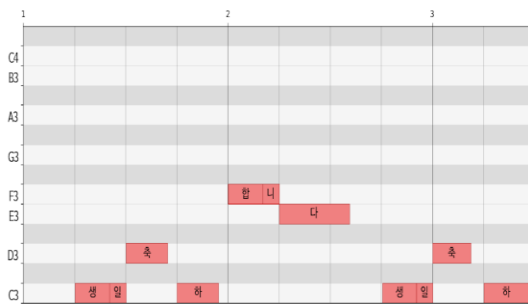
<https://ddspvocoder.github.io/ismir-demo/>

Differentiable Wavetable Synthesis
Neural Granular Sound Synthesis
DDS-Piano

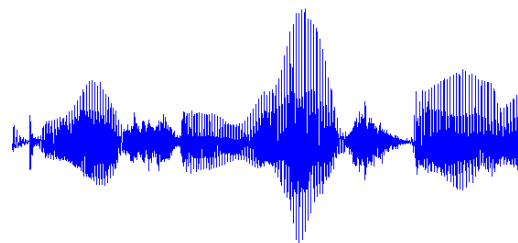
...

Singing Voice Synthesis (SVS)

- Generate singing voice audio from melody (MIDI) and lyrics (text)
 - Huge advances in neural SVS models over the past few years
 - Challenge: the lack of precisely annotated MIDI data to audio



Melody and Lyrics



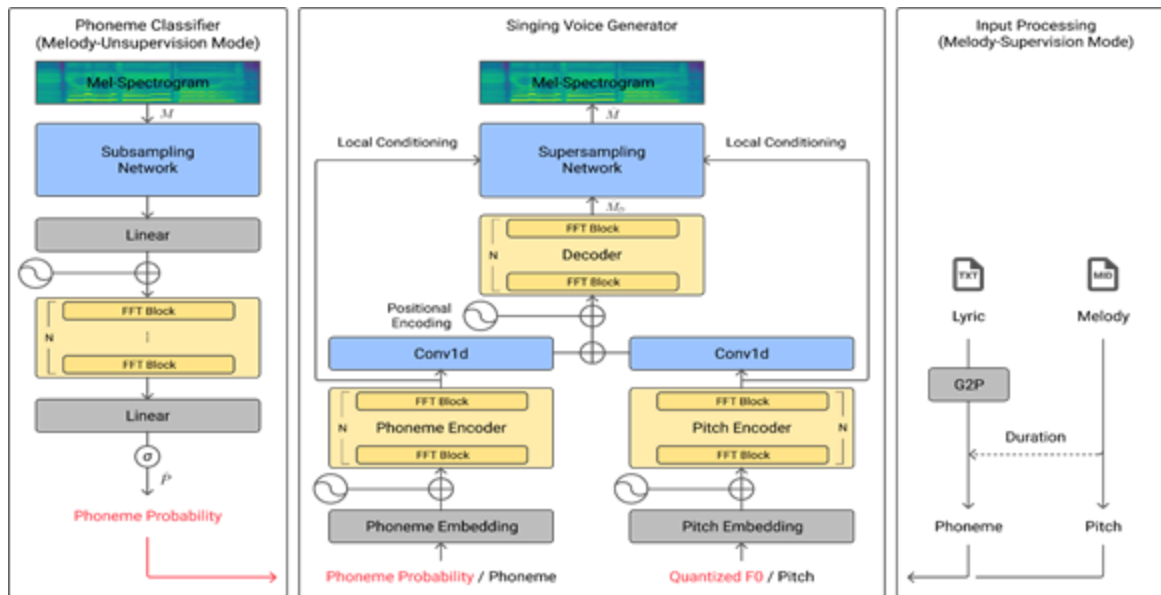
Audio

Singing Voice Synthesis



Singing Voice Synthesis Without MIDI Labels

- Train the model only with non-aligned text labels (like speech synthesis)
 - A phoneme classifier with the CTC loss: time-aligned phonemes (text input)
 - A pitch estimator: F0 (MIDI input)

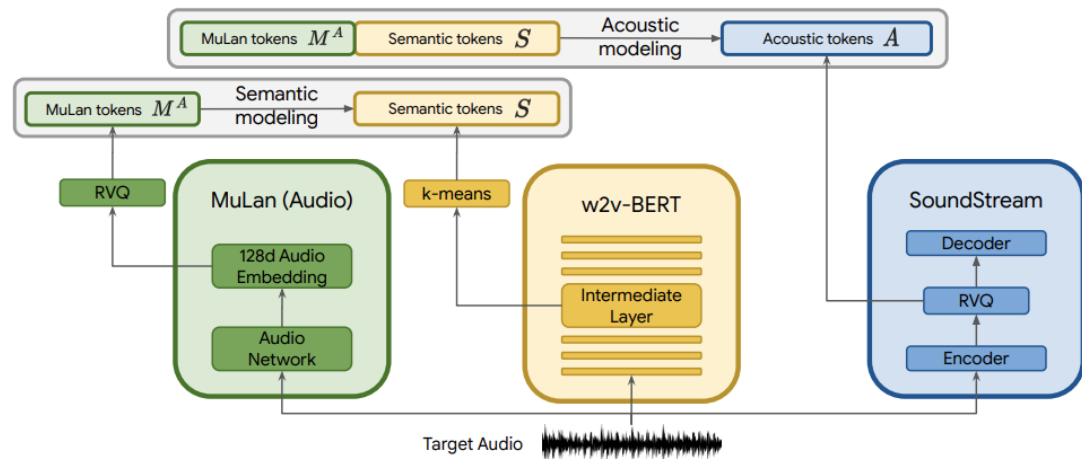


Singing Voice Synthesis Services

- Vox Factory: <https://voxfactory.app/>

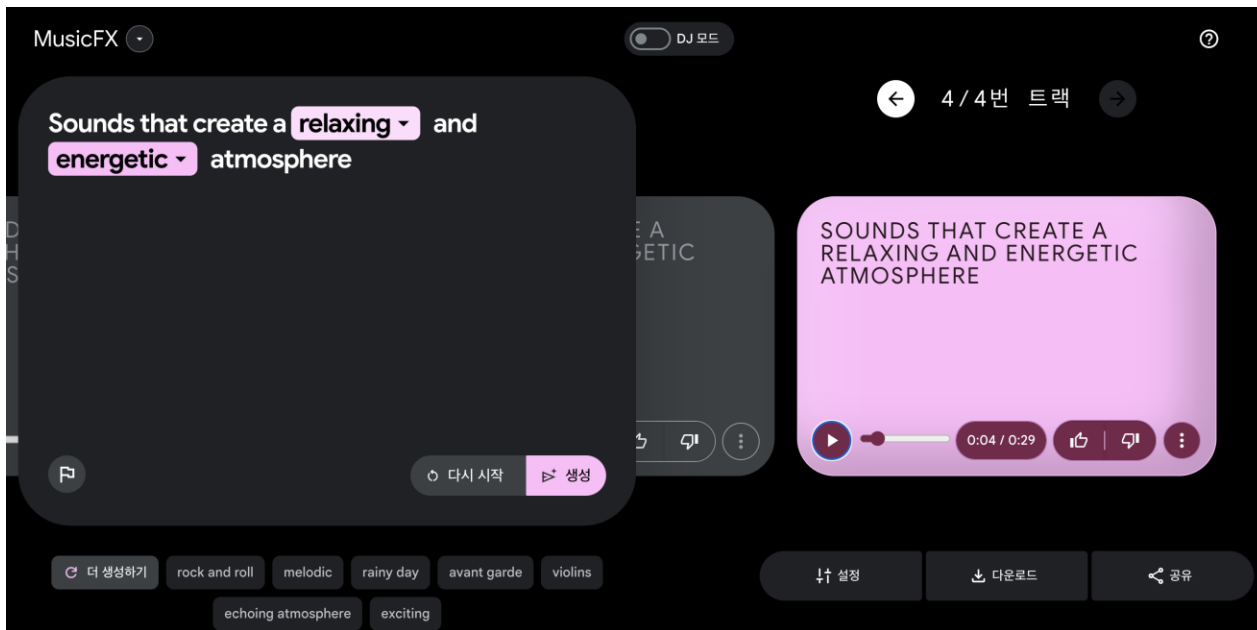
MusicLM

- Audio language model using acoustics tokens
 - text conditioning and/or melodic conditioning



<https://google-research.github.io/seanet/musiclm/examples/>

MusicLM



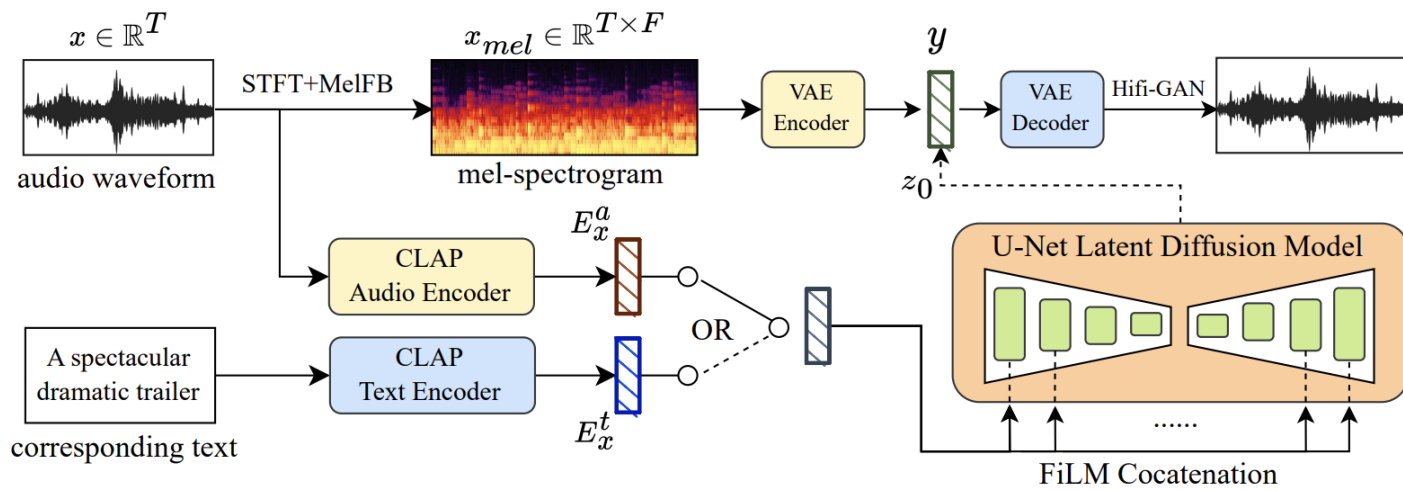
Google MusicFX



<https://aitestkitchen.withgoogle.com/ko/tools/music-fx>

MusicLDM

- Text-to-music generation model using a Latent Diffusion Model (LDM)
 - Use VAE on spectrogram to obtain latent space and HiFi-GAN for vocoder
 - CLAP (audio-text joint embedding model) for text conditioning

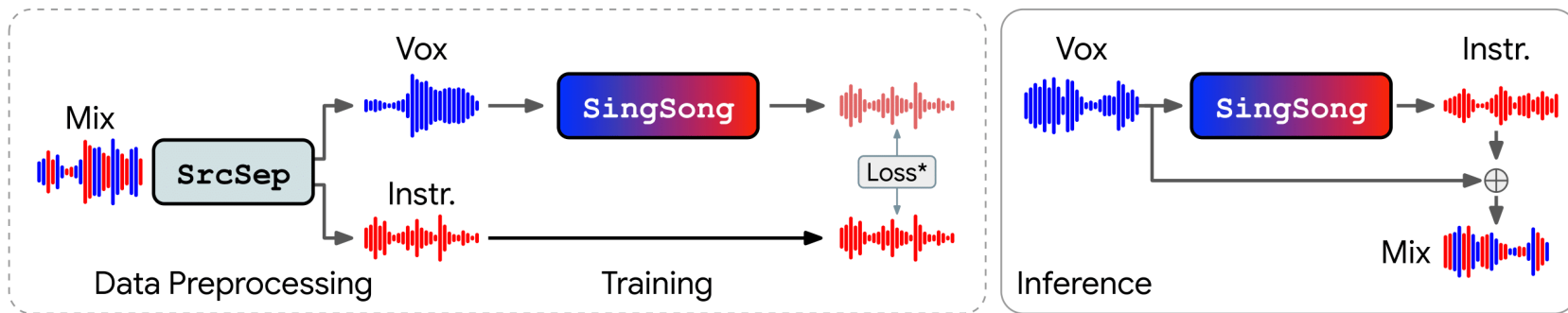




<https://musicldm.github.io/>

Arrangement Generation

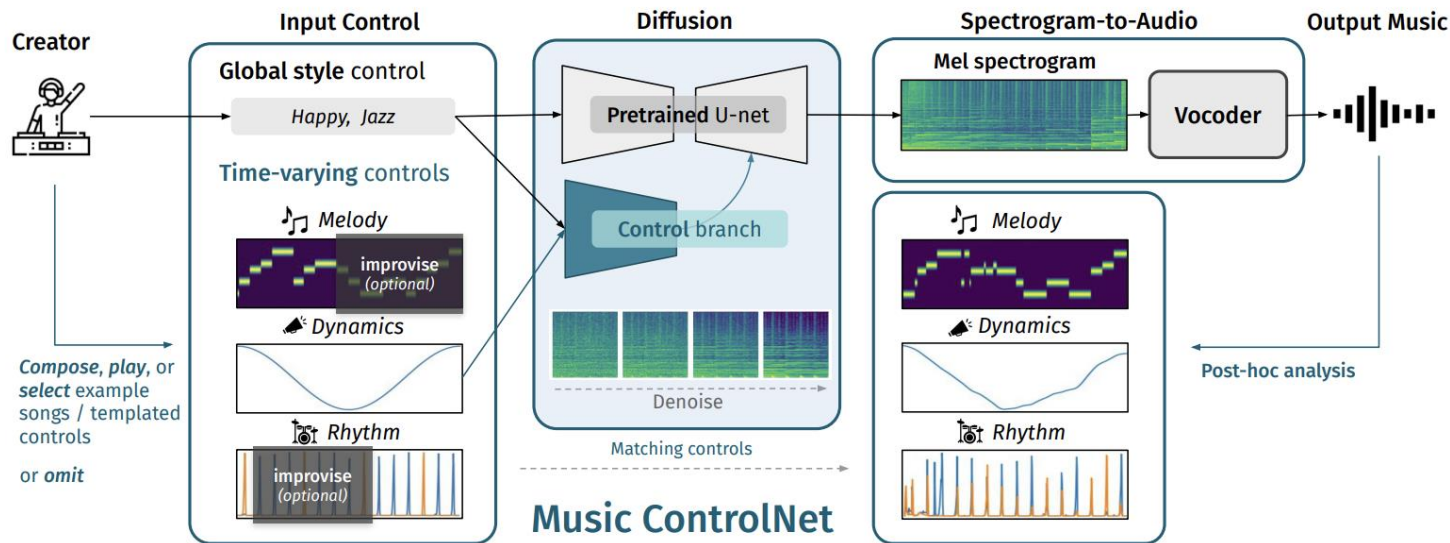
- Generate instrumental sounds given singing voice



<https://storage.googleapis.com/sing-song/index.html>

Music ControlNet

- Allow temporal music features as a time-varying control
 - Melody (chroma), dynamic curve, beat position



<https://musiccontrolnet.github.io/web/>

Music ControlNet: Multiple Time-varying Controls for Music Generation

Demo Video

Shih-Lun Wu^{1,2*}, Chris Donahue¹, Shinji Watanabe¹, & Nicholas J. Bryan²

¹School of Computer Science, Carnegie Mellon University

²Adobe Research

*Work done during an internship at Adobe Research.

<https://www.youtube.com/watch?v=QVr-S-DyccU>

Music Generation Services

- Suno: <https://suno.com/>
- Stable Audio: <https://www.stableaudio.com/>
- Mix Audio: <https://mix.audio/>

Concept Video

