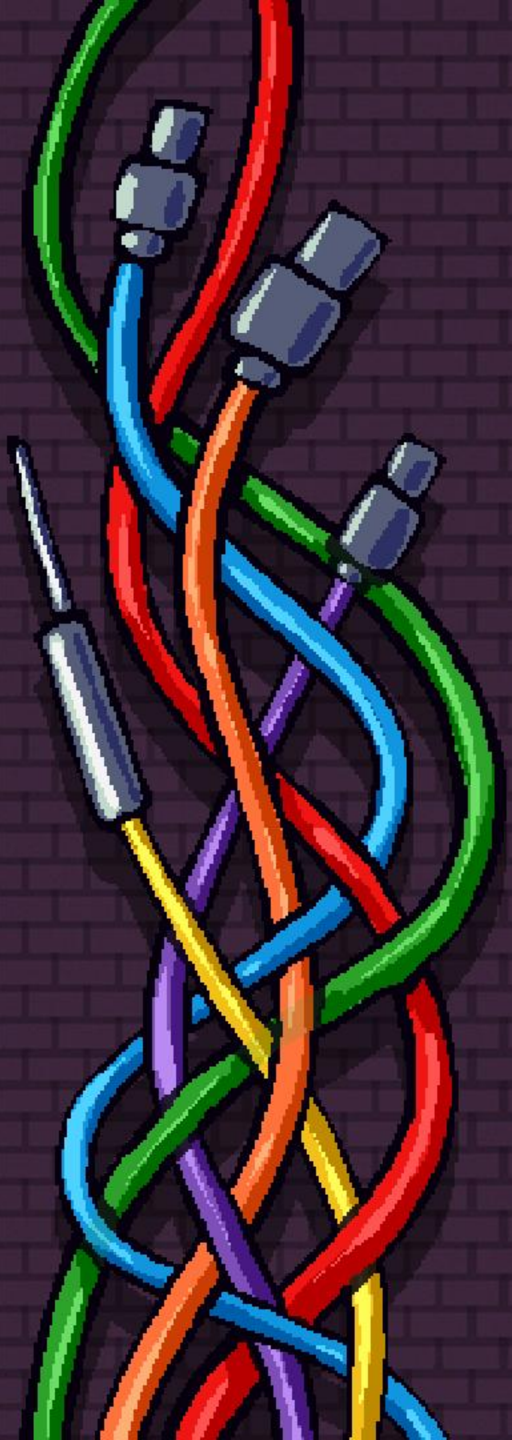
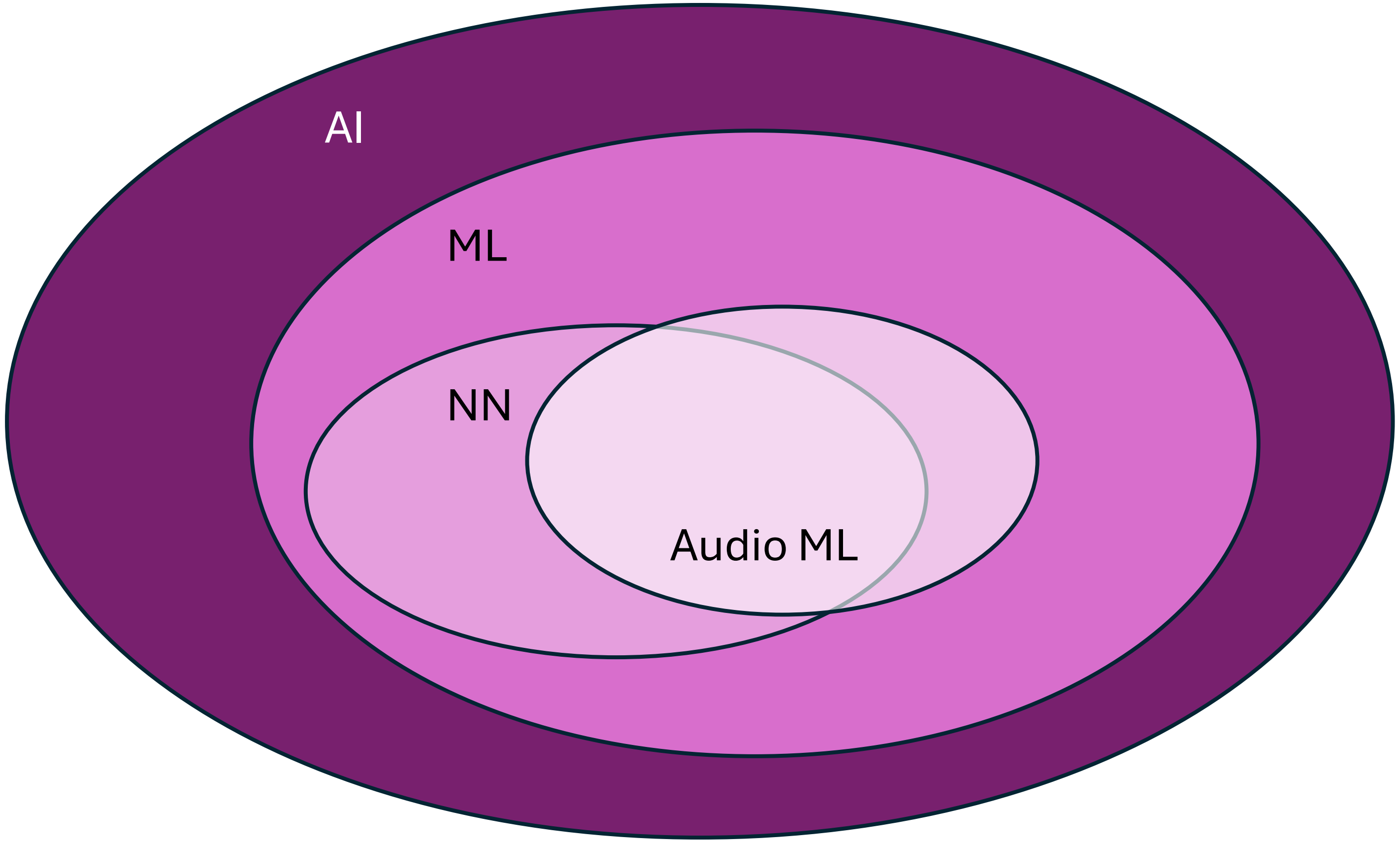


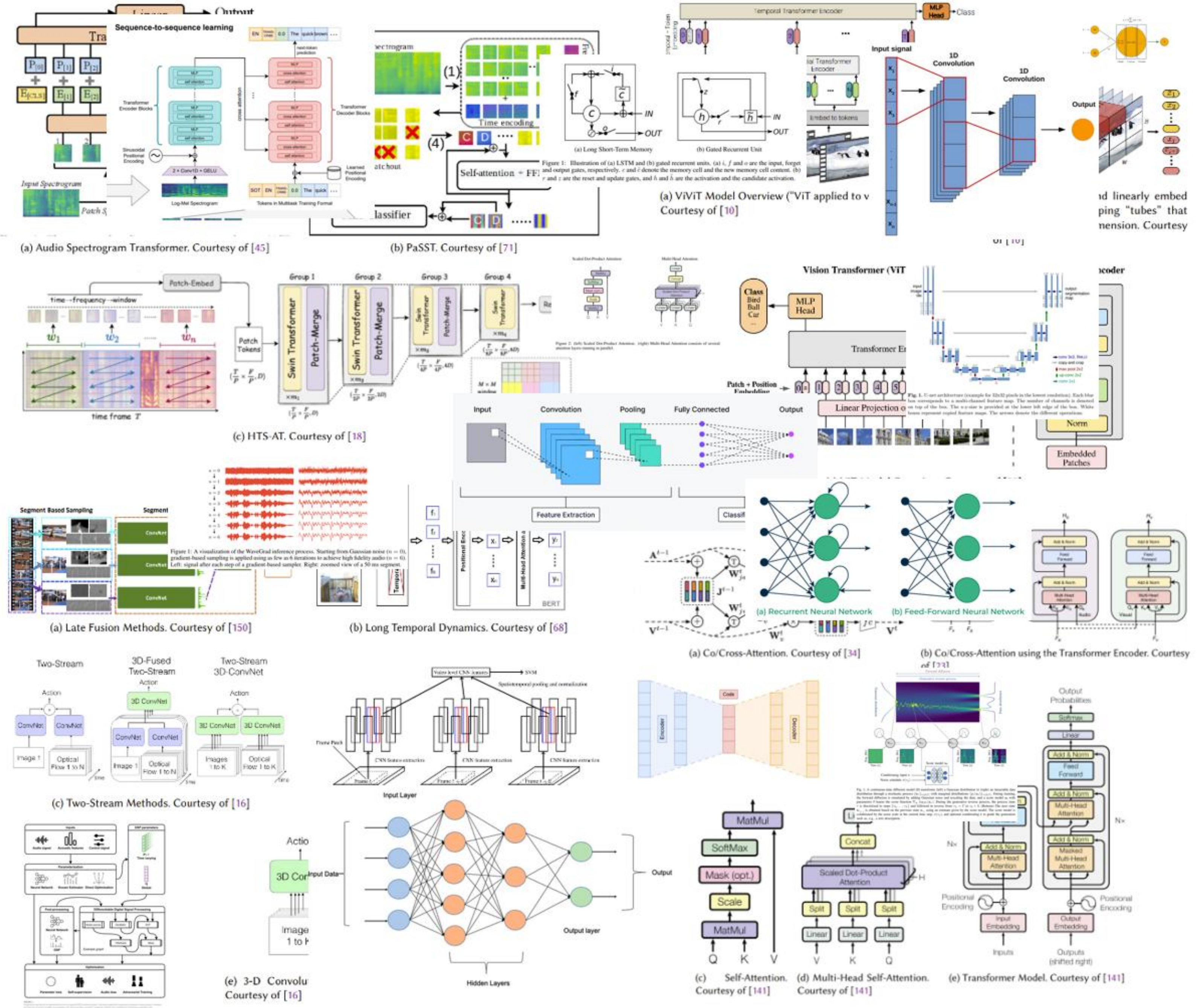
ADCx GATHER

HELICOPTER VIEW OF AUDIO ML

MARTIN SWANHOLM


AI
ML
NN
Audio ML

Model Architectures



Applications – Tasks

Tasks can be framed as mappings across modalities:

- **Audio → Audio** (effects, enhancement)
- **Text → Audio** (generation, synthesis)
- **Multiple Audio channels → Single Audio** (mixing, audio-conditioned transformations, style transfer)
- **Audio → Multiple Audio** (source separation)
- **Audio → Text** (transcription, description)
- **Audio → Symbols / Numbers** (discrete classes, timestamps for segmentation such as beat detection)
- **Audio → Intermediary → Audio** (audio codecs)

Modalities and Representation

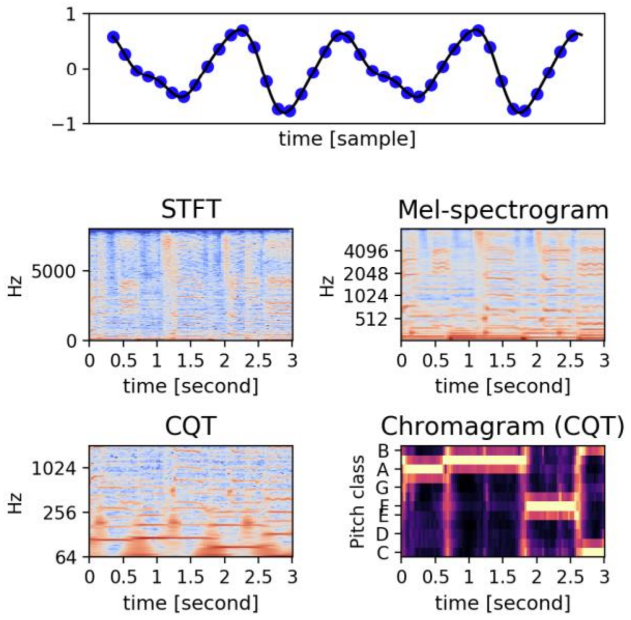
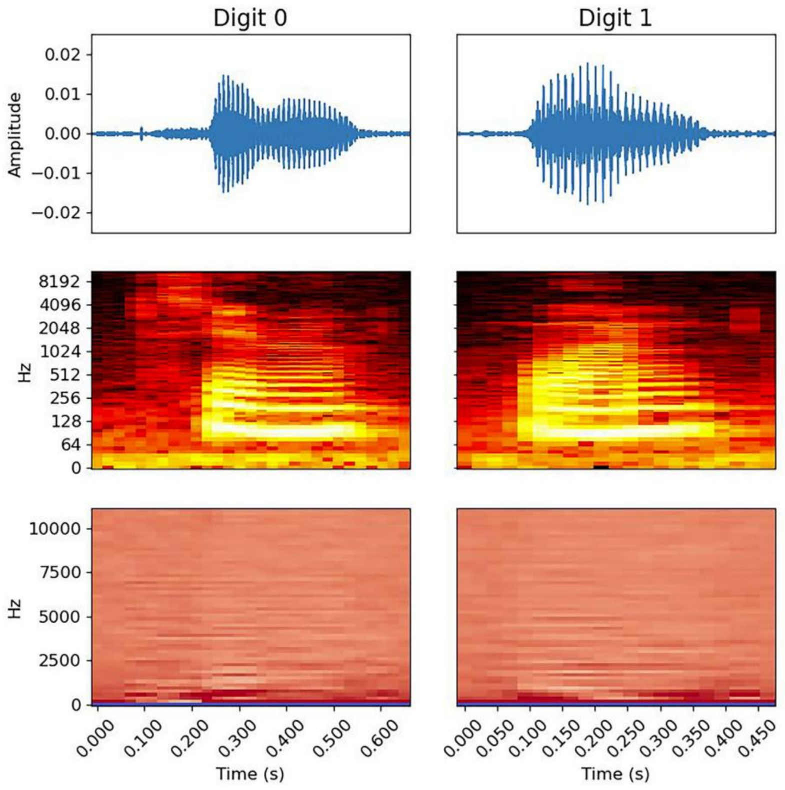


Figure 4: Audio content representations. On the top, a digital audio signal is illustrated with its samples and its continuous waveform part. STFT, mel-spectrogram, CQT, and a chromagram of a music signal are also plotted . Please note the different scales of frequency axes of STFT, melspectrogram, and CQT.

Fig. 8 | Spectra features from audio examples. Examples of audio data (top row) from the AudioMNIST dataset, with features extracted by FFT (middle row) and Mel-frequency cepstral coefficients (bottom row).



Figures from: Choi, Keunwoo & Fazekas, György & Cho, Kyunghyun & Sandler, Mark. (2017). A Tutorial on Deep Learning for Music Information Retrieval. https://arxiv.org/abs/1709.04396 and McCarthy, R.A., Zhang, Y., Verburg, S.A. et al. Machine Learning in Acoustics: A Review and Open-source Repository. npj Acoust. 1, 18 (2025). https://doi.org/10.1038/s44384-025-00021-w

The Premise

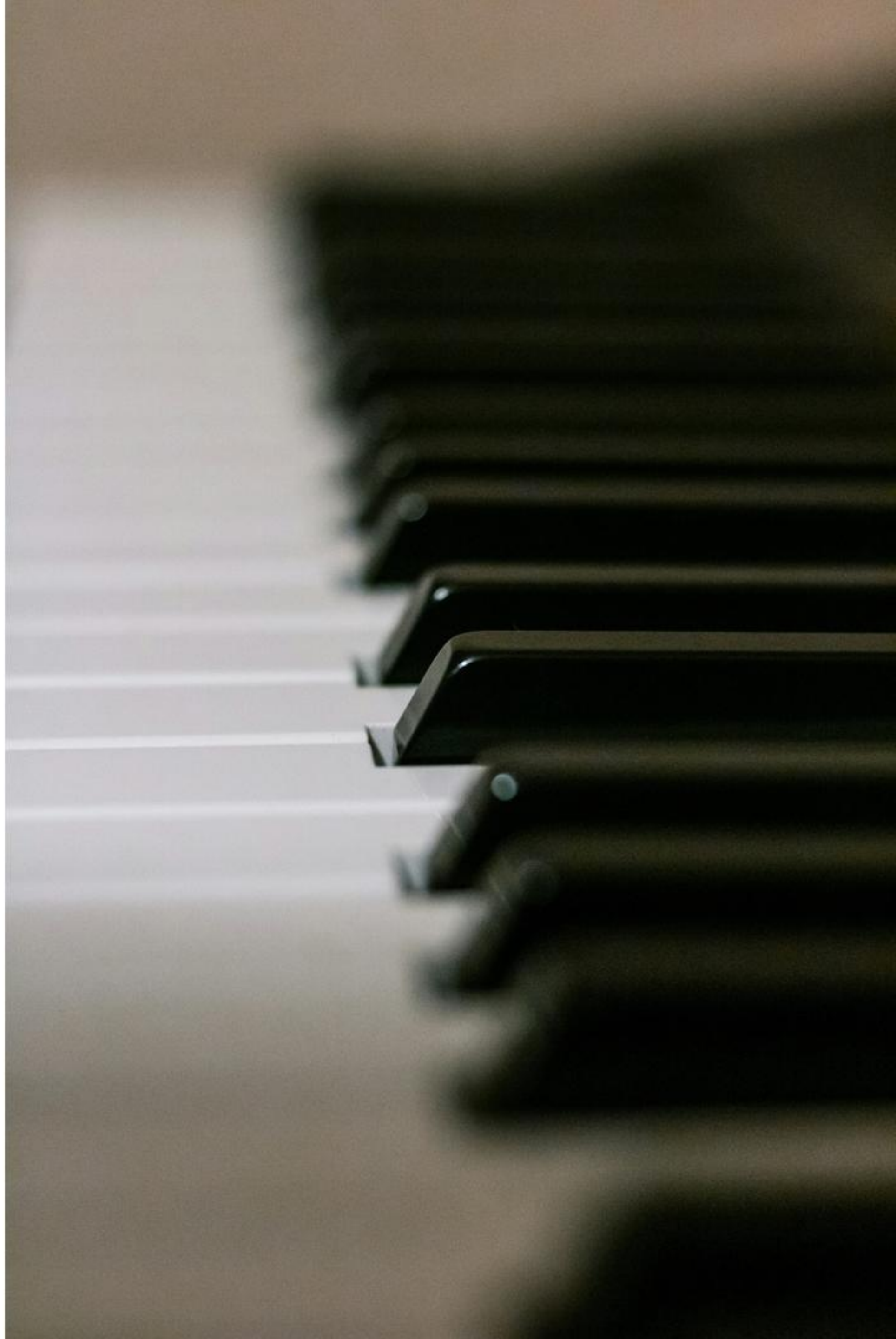


Photo by Victor Barrios on Unsplash

Deepest level

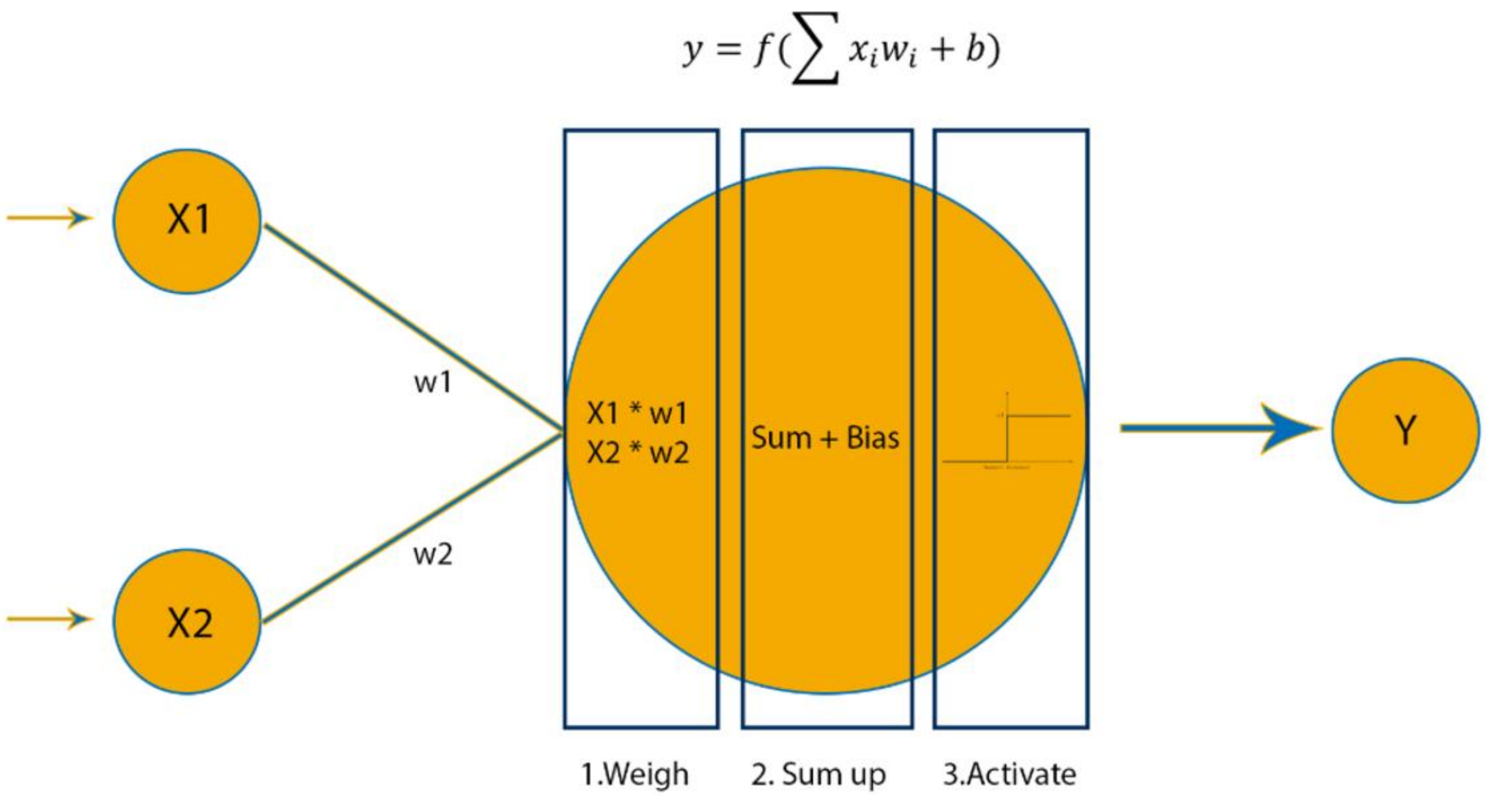


Figure from: https://telefonicatech.uk/wp-content/uploads/adatis/PERCEPTRON-OR-Implementation.gif

FCN

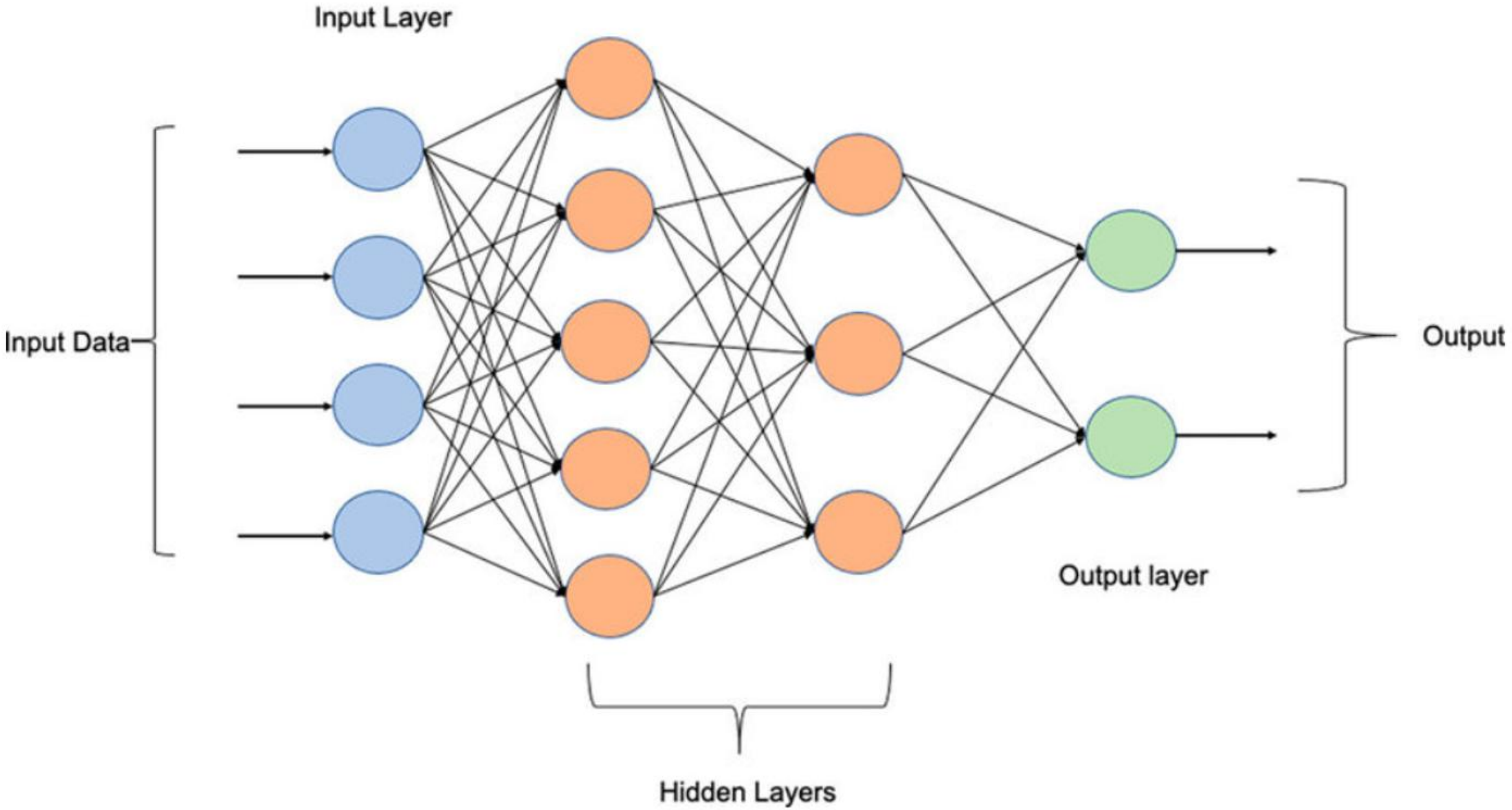


Figure from: Afan, Haitham & Ibrahem Ahmed Osman, Ahmedbahaaaldin & Essam, Yusuf & Ali Najah Ahmed, Al-Mahfoodh & Huang, Yuk & Kisi, Ozgur & Sherif, Mohsen & Sefelnasr, Ahmed & Chau, Kwok & El-Shafie, Ahmed. (2021). Modeling the fluctuations of groundwater level by employing ensemble deep learning techniques. Engineering Applications of Computational Fluid Mechanics. 15. 1420-1439. 10.1080/19942060.2021.1974093.

CNN

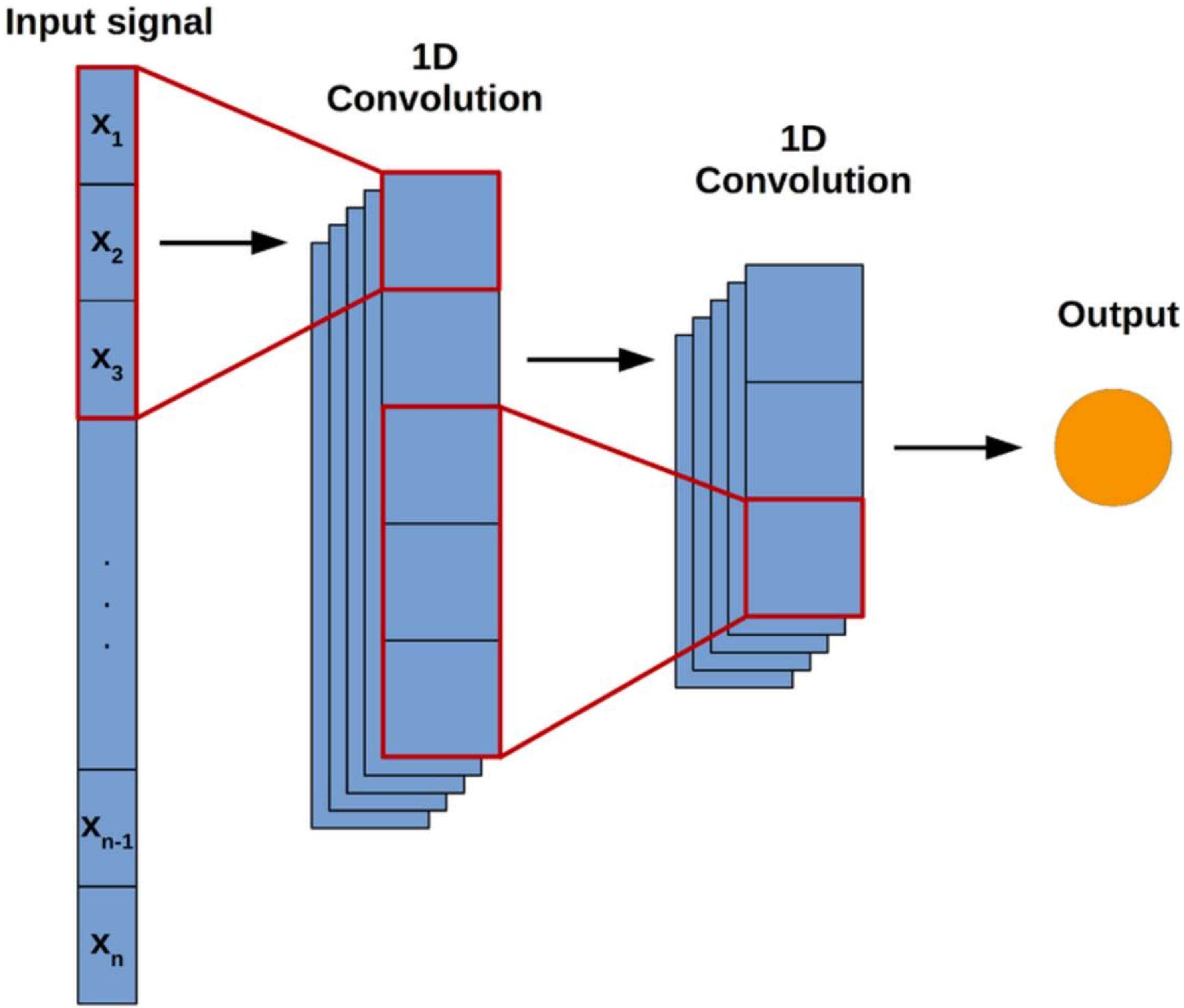


Figure from: Shenfield, Alex & Howarth, Martin. (2020). A Novel Deep Learning Model for the Detection and Identification of Rolling Element-Bearing Faults. Sensors. 20. 10.3390/s20185112.
https://www.researchgate.net/publication/344229502_A_Novel_Deep_Learning_Model_for_the_Detection_and_Identification_of_Rolling_Element-Bearing_Faults/citation/download

Autoencoder

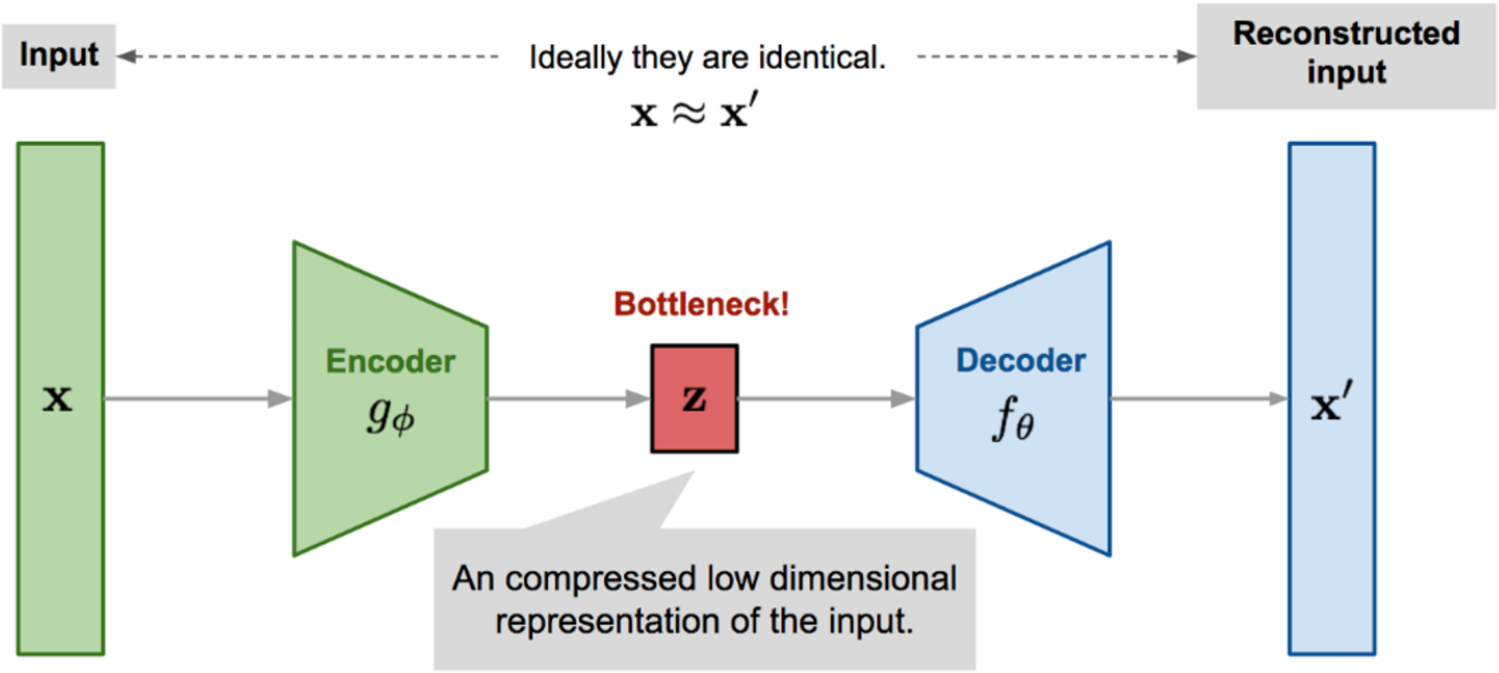


Figure from: Weng, Lilian: "From Autoencoder to Beta-VAE", https://lilianweng.github.io/posts/2018-08-12-vae/

Variational Autoencoder

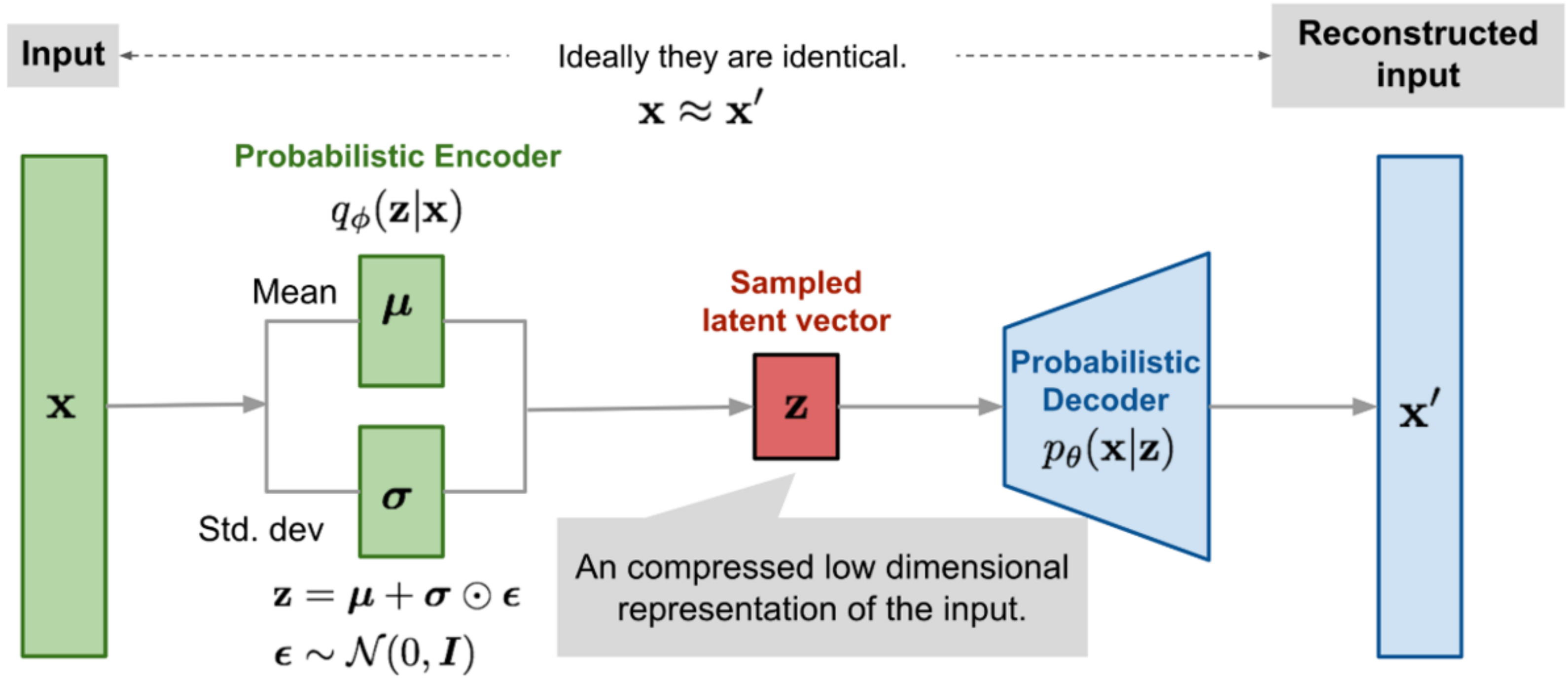


Figure from: Weng, Lilian: "From Autoencoder to Beta-VAE", https://lilianweng.github.io/posts/2018-08-12-vae/

U-Net

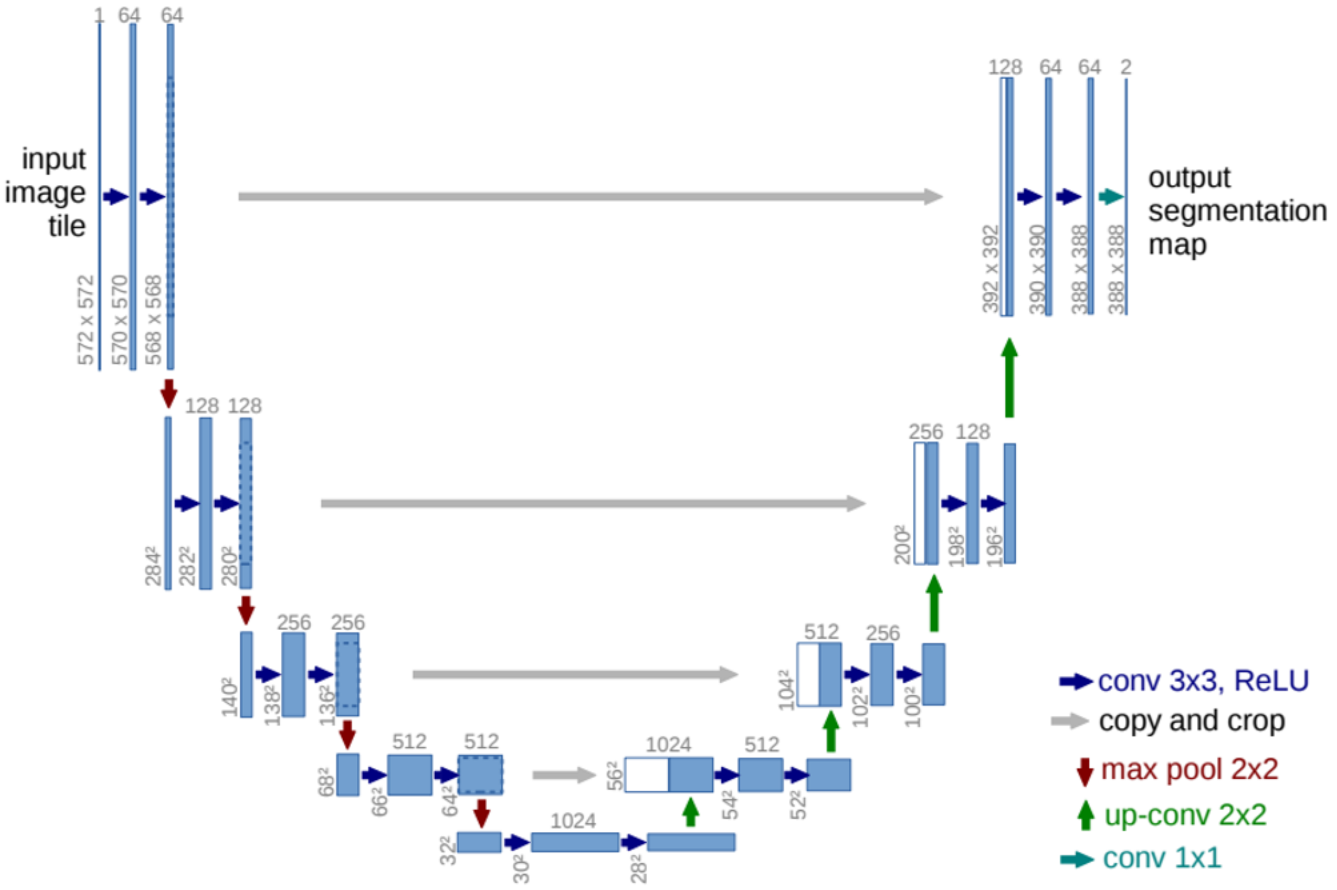


Fig. 1. U-net architecture (example for 32x32 pixels in the lowest resolution). Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations.

Figure from: Ronneberger, Olaf: "U-Net: Convolutional Networks for Biomedical Image Segmentation", https://arxiv.org/pdf/1505.04597

2-D CNN

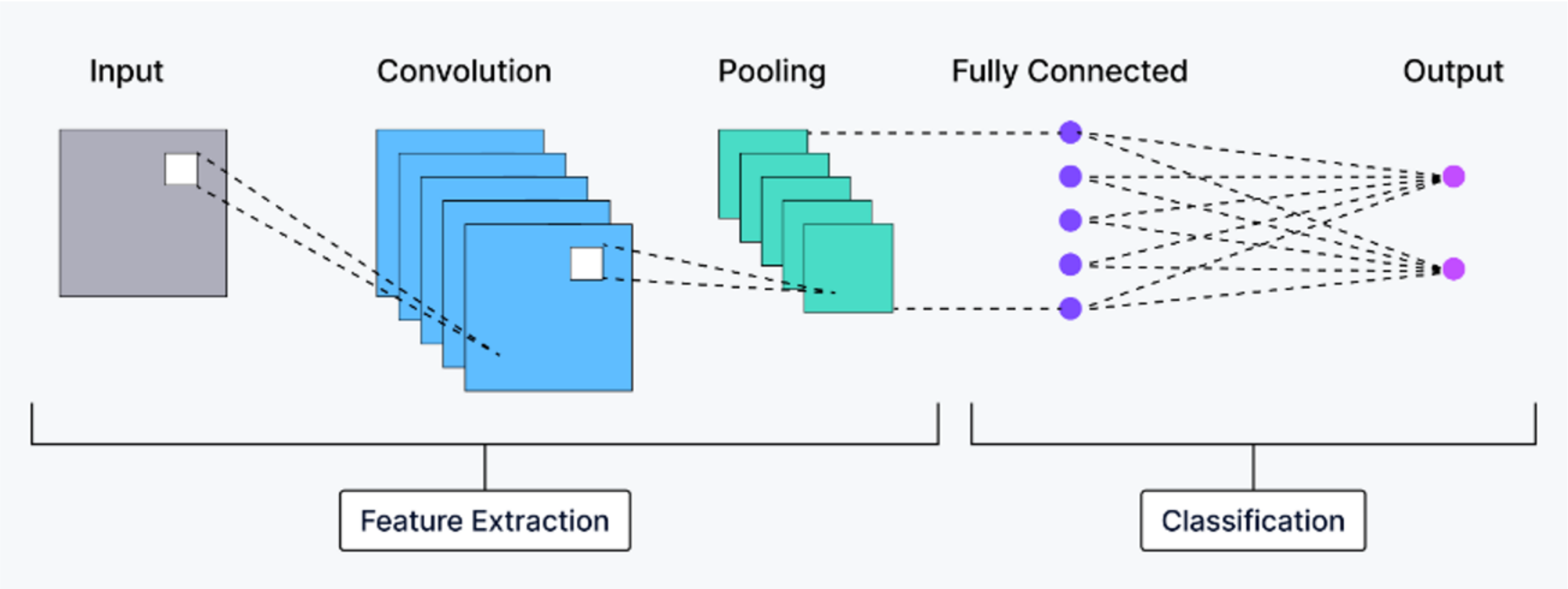


Figure from: https://zilliz.com/glossary/convolutional-neural-network

RNN

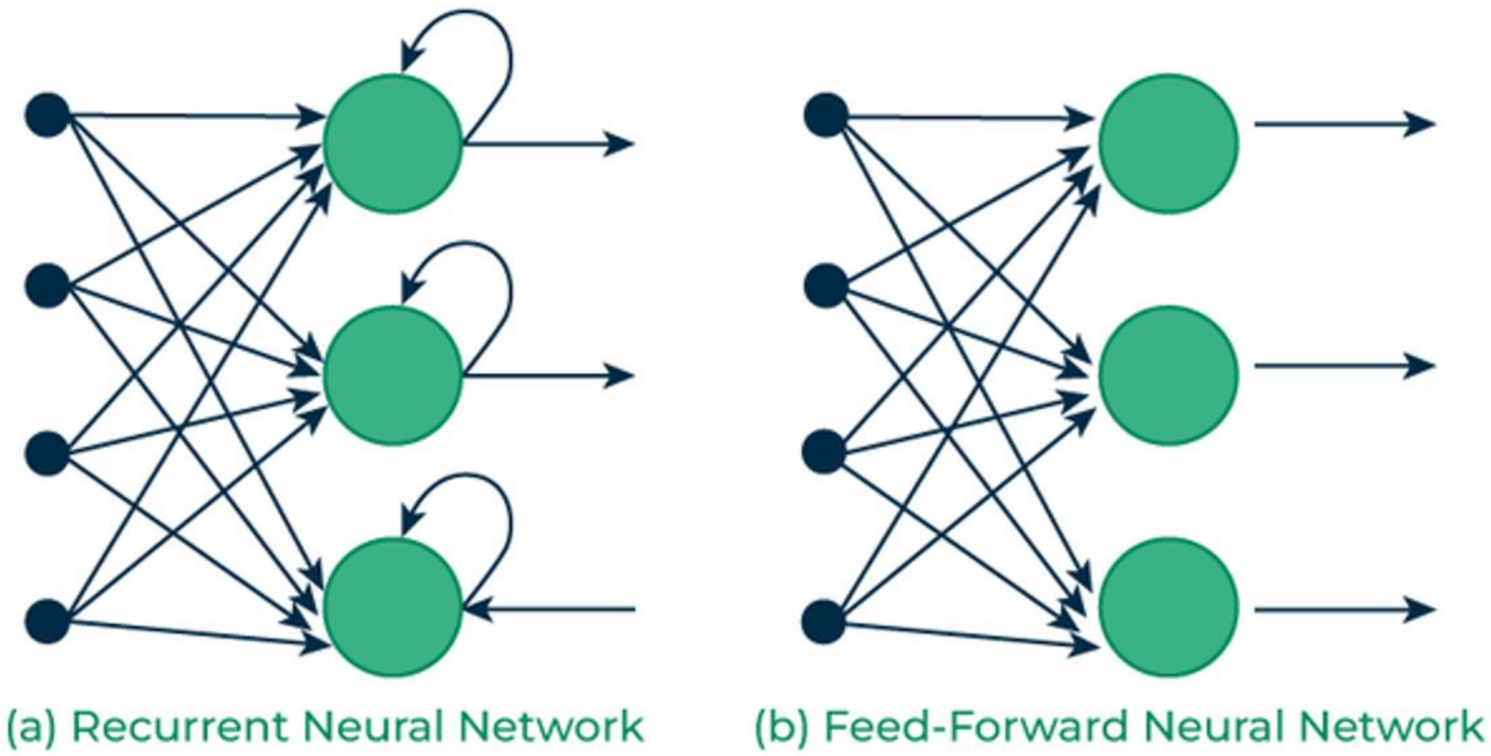
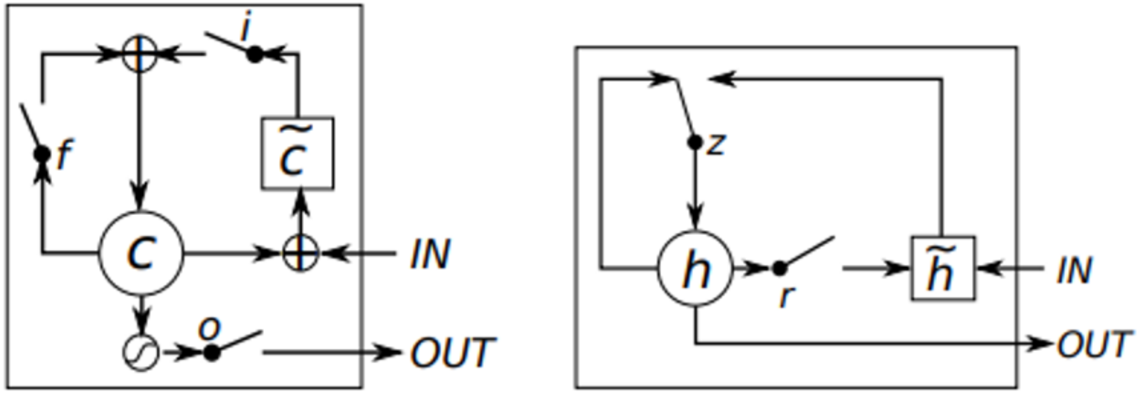


Figure courtesy of: "Introduction to Recurrent Neural Networks", https://www.geeksforgeeks.org/machine-learning/introduction-to-recurrent-neural-network/

LSTM GRU



(a) Long Short-Term Memory (b) Gated Recurrent Unit

Figure 1: Illustration of (a) LSTM and (b) gated recurrent units. (a) i, f and o are the input, forget and output gates, respectively. c and $\tilde{c}$ denote the memory cell and the new memory cell content. (b) r and z are the reset and update gates, and h and $\tilde{h}$ are the activation and the candidate activation.

Figure from: Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. https://arxiv.org/pdf/1412.3555

Transformer

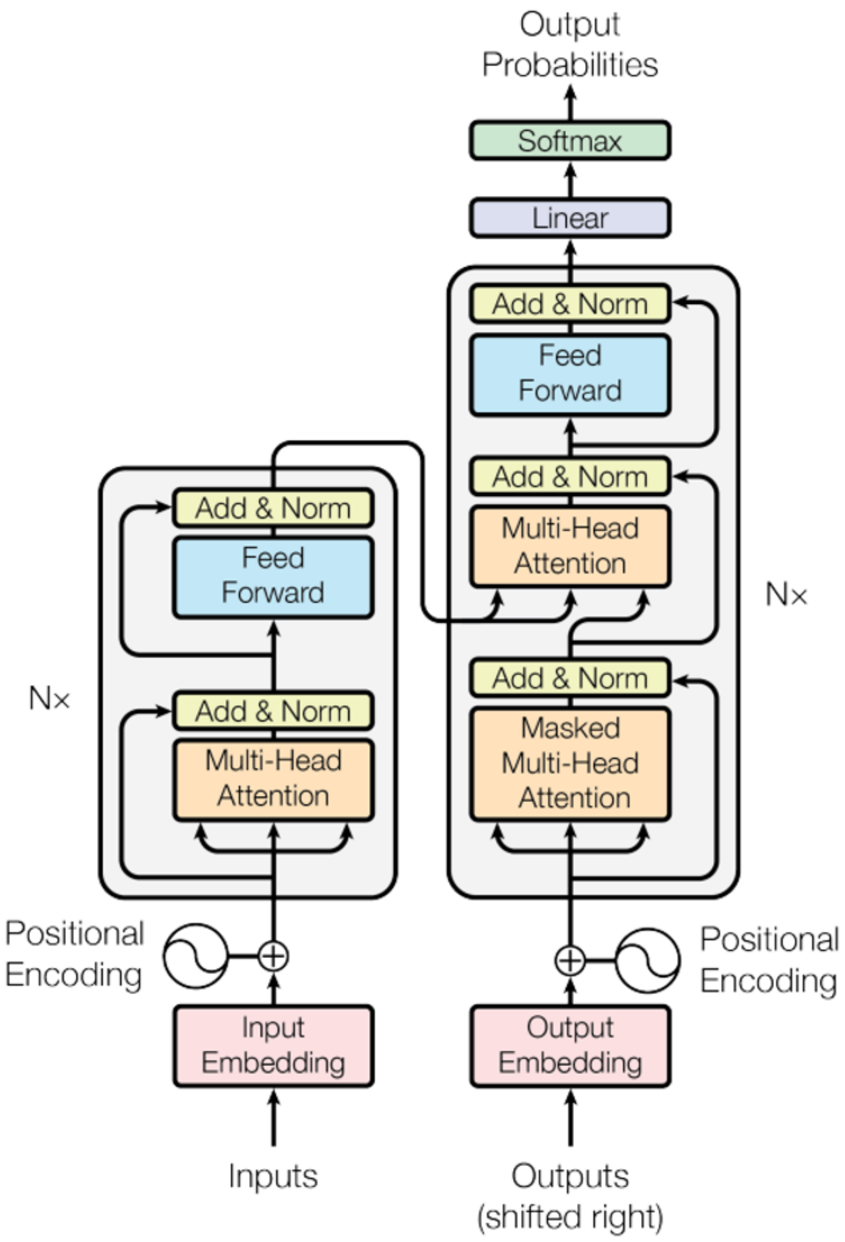


Figure 1: The Transformer - model architecture.

Figure from: Vaswani, Ashish et al.: "Attention Is All You Need", https://papers.nips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

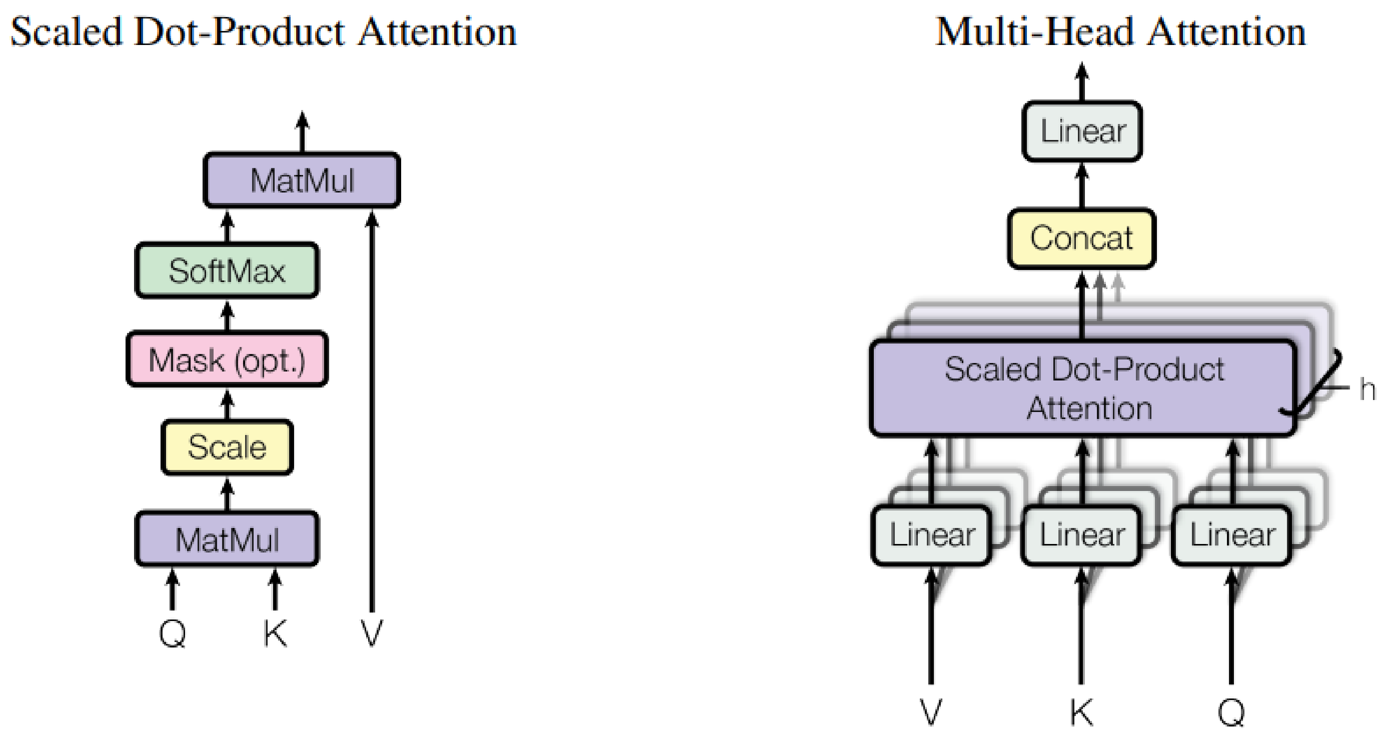


Figure 2: (left) Scaled Dot-Product Attention. (right) Multi-Head Attention consists of several attention layers running in parallel.

Figure from: Vaswani, Ashish et al.: “Attention Is All You Need”, https://papers.nips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

Whisper Architecture

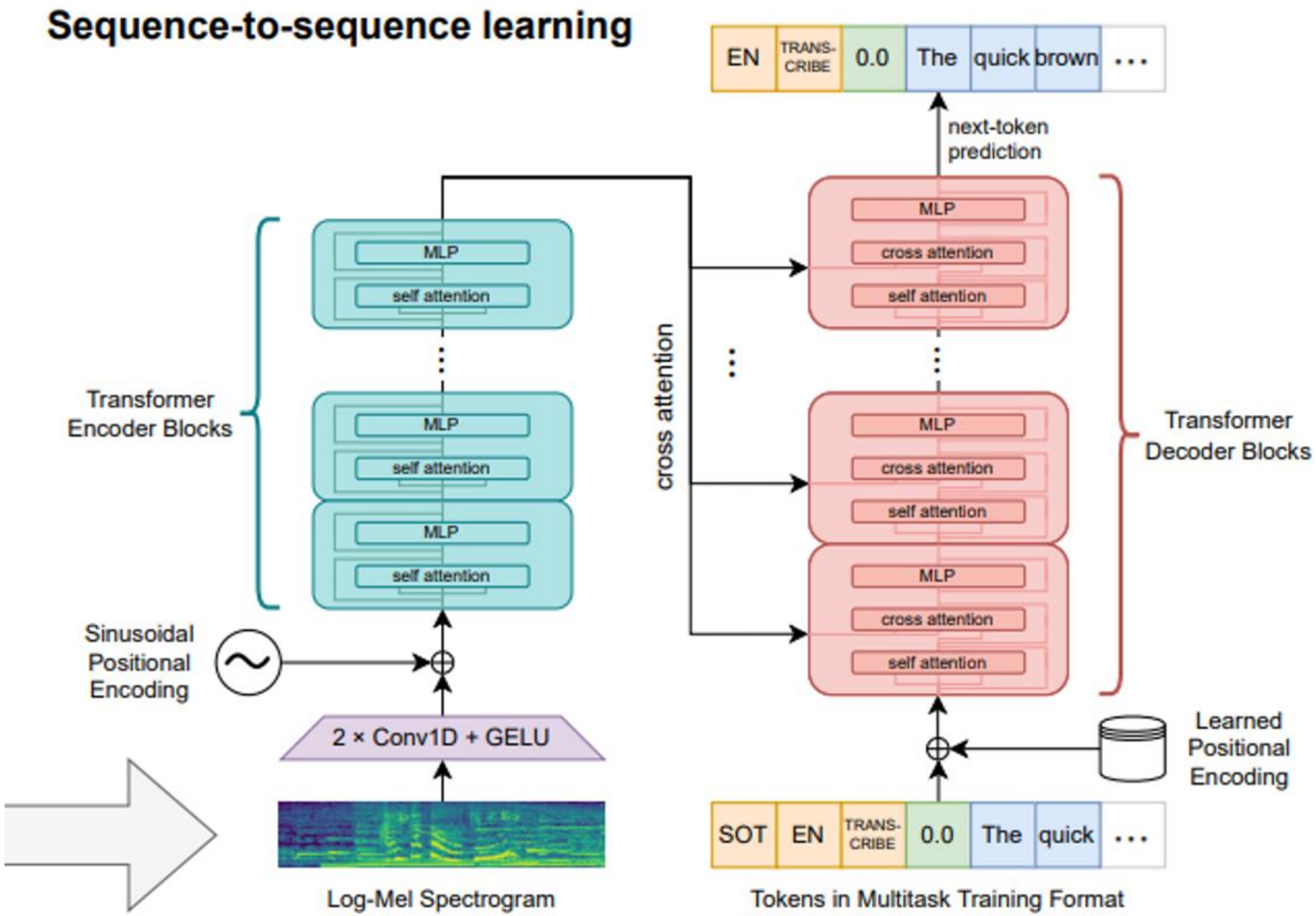


Figure from: Radford, Alec et al.: "Robust Speech Recognition via Large-Scale Weak Supervision", https://cdn.openai.com/papers/whisper.pdf

DDSP

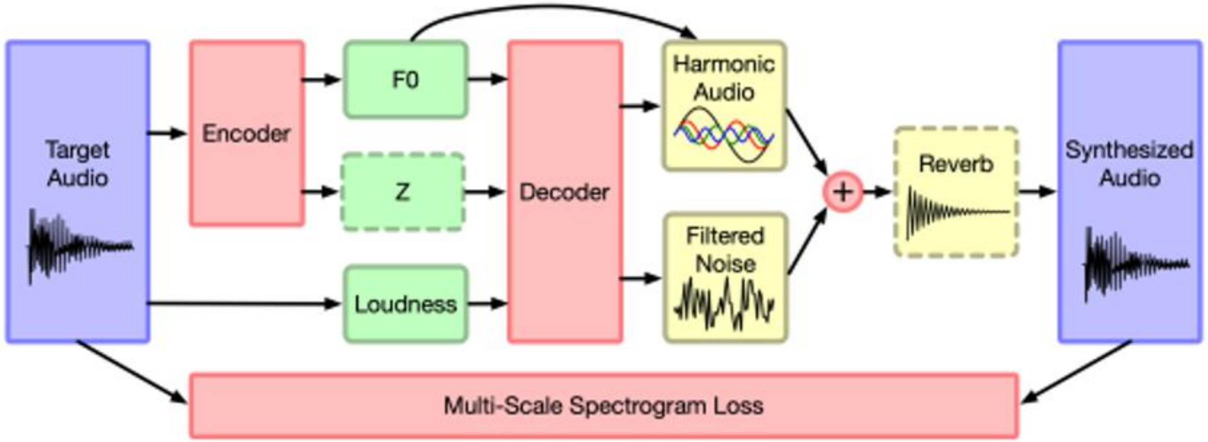


Figure 2: Autoencoder architecture. Red components are part of the neural network architecture, green components are the latent representation, and yellow components are deterministic synthesizers and effects. Components with dashed borders are not used in all of our experiments. Namely, z is not used in the model trained on solo violin, and reverb is not used in the models trained on NSynth. See the appendix for more detailed diagrams of the neural network components.

Figure from: Engel, Jesse et al.: "DDSP: Differentiable Digital Signal Processing", https://arxiv.org/pdf/2001.04643

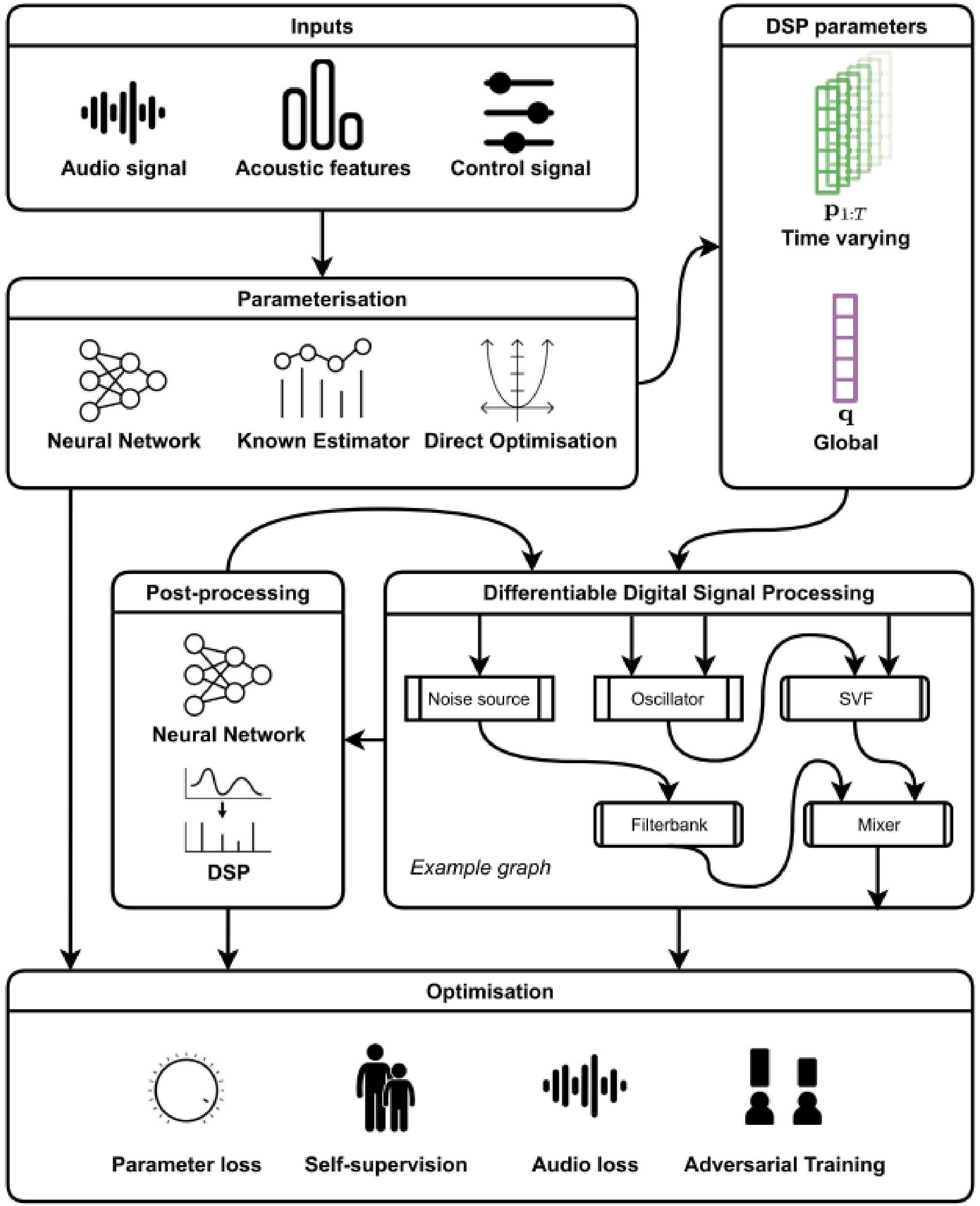


FIGURE 1
A high level overview of the general structure of a typical DDSP synthesis system. Not every depicted component is present in every system, however we find this structure broadly encompasses the work we have surveyed. Graphical symbols are included for illustrative purposes only.

Figure from: Hayes, Ben et al.: “A review of differentiable digital signal processing for music and speech synthesis”, https://comma.eecs.qmul.ac.uk/assets/pdf/Hayes_DDSP_Review.pdf

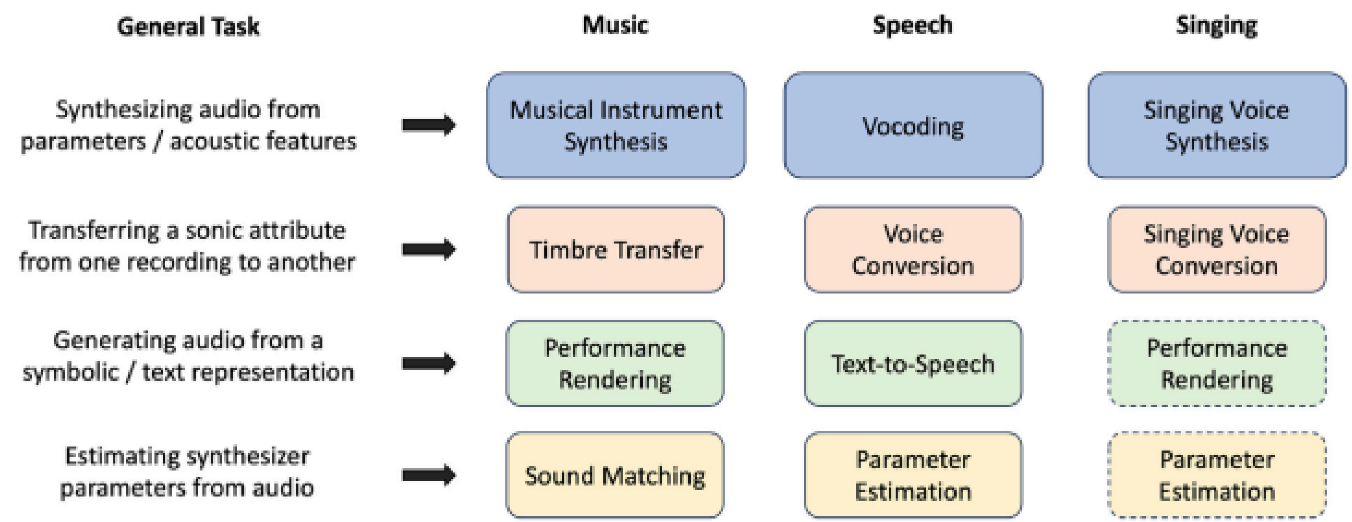
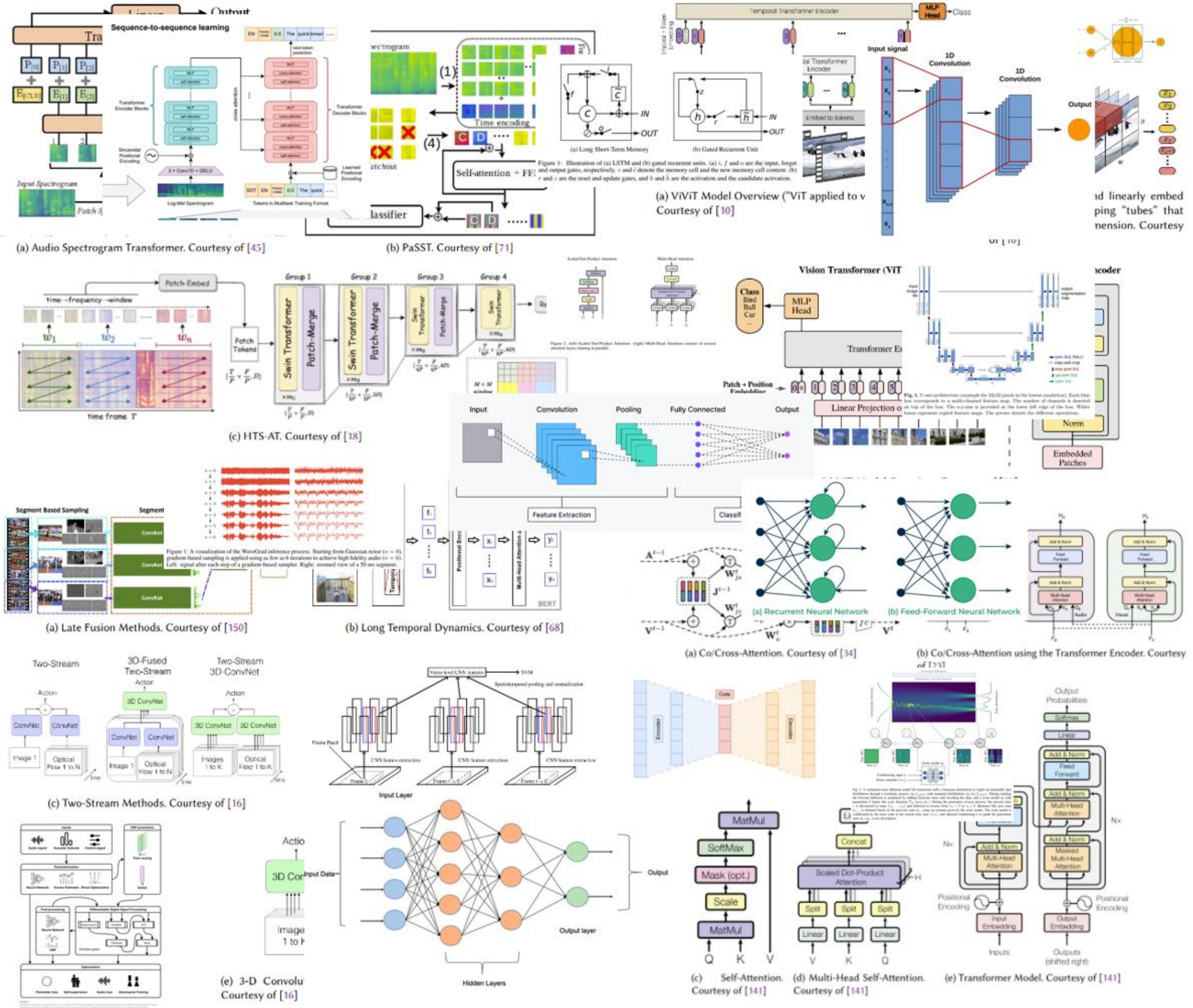


FIGURE 3
A high level view of audio synthesis tasks to which DDSP has been applied. Further discussion on each is presented in Section 2.

Figure from: Hayes, Ben et al.: "A review of differentiable digital signal processing for music and speech synthesis", https://comma.eecs.qmul.ac.uk/assets/pdf/Hayes_DDSP_Review.pdf

hybrid models



GAN

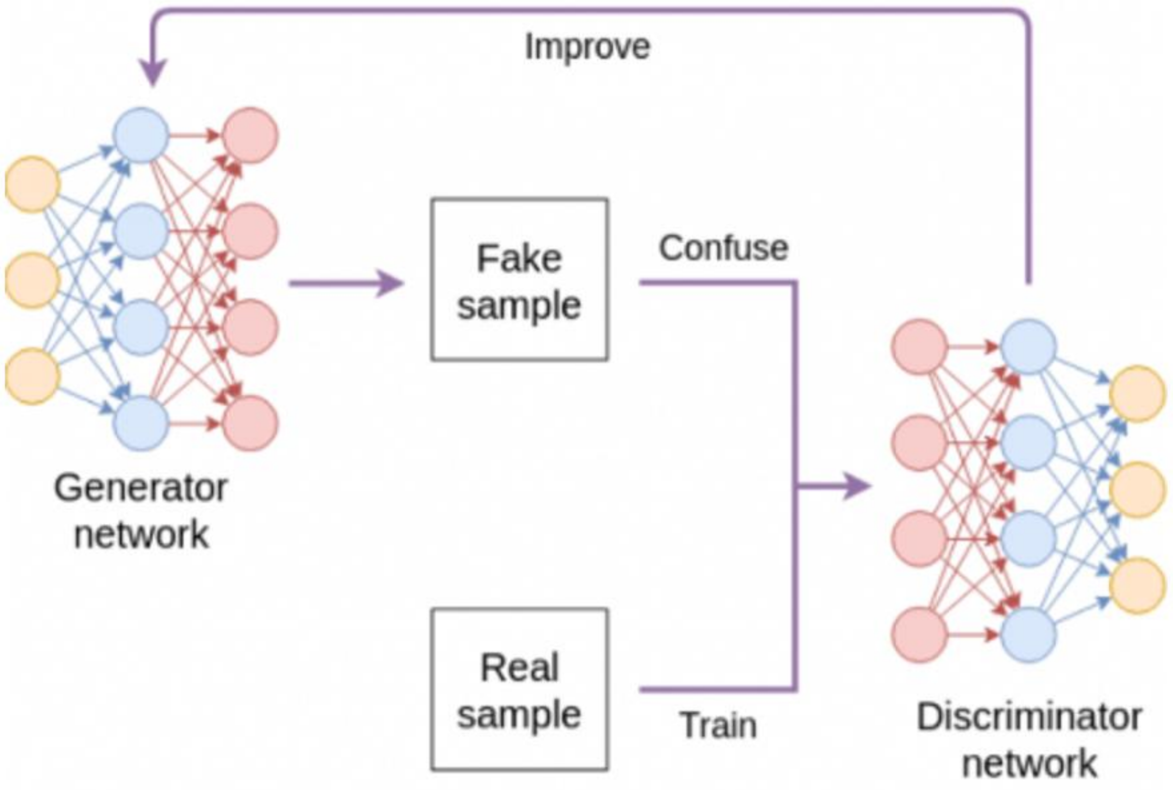


Figure from: Abascal, Miguel Ángel Hoyo: “Generative Adversarial Networks: A rivalry that strengthens”, https://quantdare.com/generative-adversarial-networks-a-rivalry-that-strengthens/

Diffusion

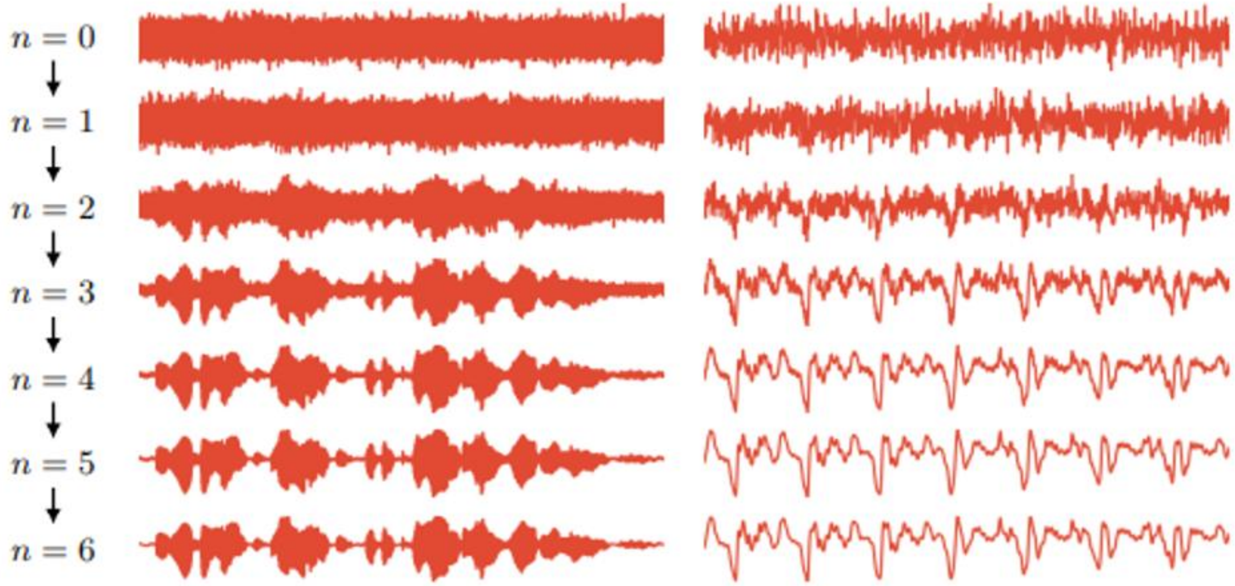


Figure 1: A visualization of the WaveGrad inference process. Starting from Gaussian noise ($n = 0$), gradient-based sampling is applied using as few as 6 iterations to achieve high fidelity audio ($n = 6$). Left: signal after each step of a gradient-based sampler. Right: zoomed view of a 50 ms segment.

Figure from: Chen, Nanxin et al.: "WAVEGRAD: ESTIMATING GRADIENTS FOR WAVEFORM GENERATION", ICLR 2021, https://openreview.net/pdf?id=NsMLjcFaO8O

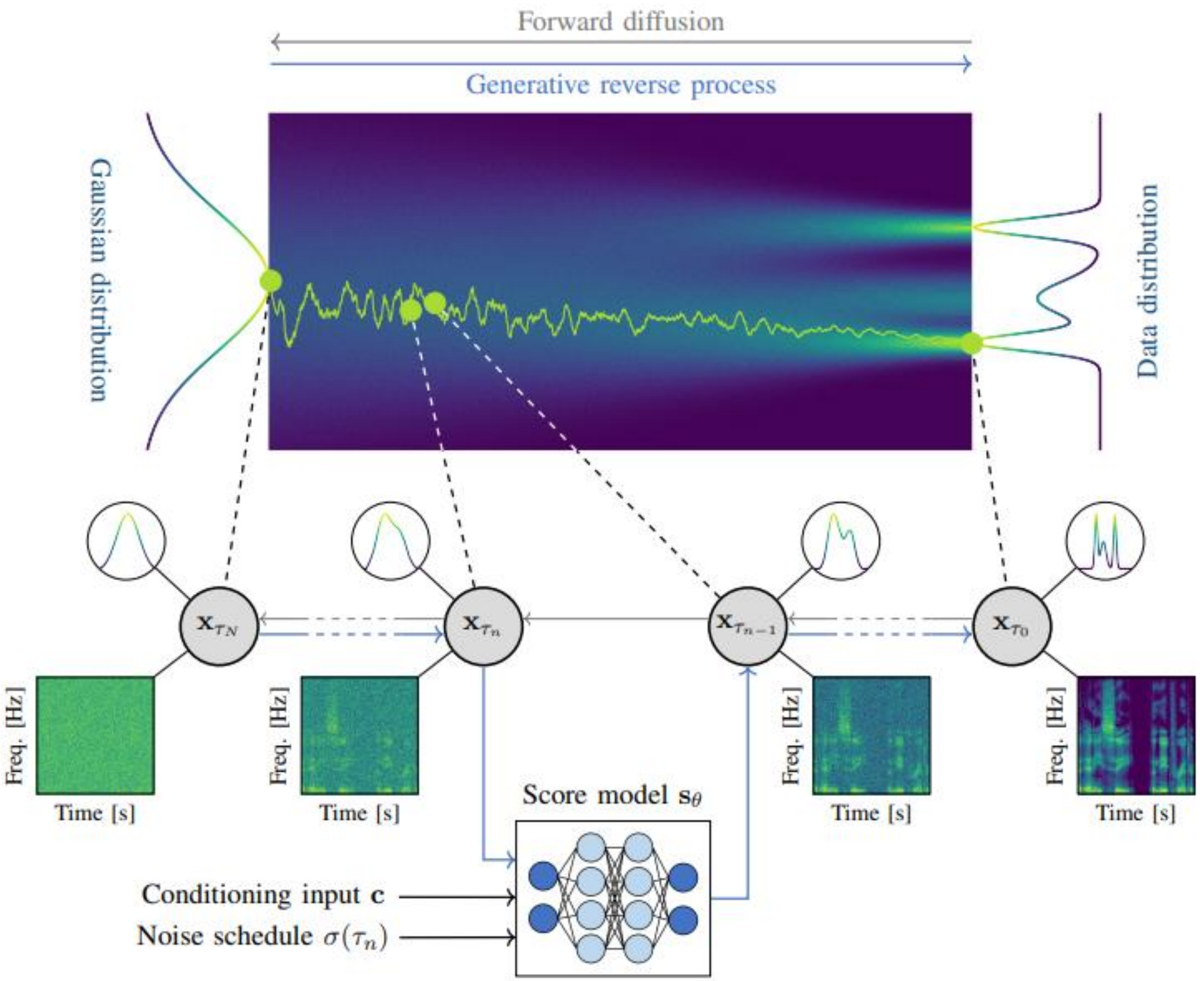


Fig. 1: A continuous-time diffusion model [8] transforms (left) a Gaussian distribution to (right) an intractable data distribution through a stochastic process $\{\mathbf{x}_\tau\}_{\tau\in[0,T]}$ with marginal distributions $\{p_\tau(\mathbf{x}_\tau)\}_{\tau\in[0,T]}$. During training, the forward diffusion is simulated by adding Gaussian noise and rescaling the data, and a score model $\mathbf{s}_\theta$ with parameters θ learns the score function $\nabla_{\mathbf{x}_\tau} \log p_\tau(\mathbf{x}_\tau)$. During the generative reverse process, the process time τ is discretized to steps $\{\tau_0, \ldots, \tau_N\}$ and followed in reverse from $\tau_N = T$ to $\tau_0 = 0$. (Bottom) The next state $\mathbf{x}_{\tau_{n-1}}$ is obtained based on the previous state $\mathbf{x}_{\tau_n}$ using an estimate given by the score model. The score model is conditioned by the noise scale at the current time step, $\sigma(\tau_n)$, and optional conditioning $\mathbf{c}$ to guide the generation such as, e.g., a text description.

Figure from: Lemercier, Jean-Marie et al.: “Diffusion Models for Audio Restoration”, https://arxiv.org/pdf/2402.09821

Summing up

Architecture / Method	Fidelity	Latency	Accuracy / Quality	Compute Load	Long-term context	Training Data Needs	Streaming / Realtime	Suitable For
1-D CNN (waveform)	Low–medium	Low	Decent local detail	Moderate	Very limited	Moderate	Yes	Denoising, onset detection, beat tracking
2-D CNN (time/freq)	Medium–high	Low–medium	Strong on local structure	Moderate–high	Limited (short RF)	Large	Yes	ASR, tagging, classification, enhancement
Autoencoder	Low–medium (lossy)	Low	Compression / feature rep	Low–moderate	Very limited	Small–moderate	Yes	Compression, embedding, codec-like tasks
Variational Autoencoder	Medium	Low–medium	Smooth latent control	Moderate	Limited	Moderate–large	Partial	Generative sound morphing, latent exploration
U-Net	High (local detail)	Medium	Strong detail retention	Moderate–high	Limited (often short context)	Large	Yes (with tweaks)	Source separation, denoising, enhancement
RNN / LSTM / GRU	Medium	Medium	Decent if tuned	Moderate–high	Good (bounded memory)	Moderate–large	Yes, with limits	ASR, transcription, sequence labeling
Transformer	High	High (non-stream)	State of art	Very high	Excellent (if big context)	Very large	Tricky but possible	ASR, transcription, TTS, classification, tagging, music modeling
DDSP	High (naturalness)	Low–medium	Strong with priors	Moderate	Limited (often short context)	Moderate–large	Yes	Instrument synthesis, timbre transfer, effect modeling
GAN	High (if stable)	Low at inference	Variable	High (train) / Low (infer)	Depends underlying arch	Very large	Yes	Audio generation, style transfer, enhancement
Diffusion	Very high	Very high (slow)	State of art generative	Extremely high	Weak (no long seq mem)	Enormous	No	High-fidelity generation (music, speech), offline enhancement

Note: This table is mostly for fun! Unfortunately, this is far too complex to summarize like this.
In reality, it requires lots of dedicated and interesting experimentation for each task/application...

Run the experiments!

Audio Examples

System	Application/Task	Year	Architectures and methods	Representations	Team
Dragon NaturallySpeaking v.11	Automatic Speech Recognition (ASR, STT)	2010	“Pre-NN Era”: GMM-HMM (Hidden Markov Models + Gaussian Mixtures)	MFCC → Phonemes → Text	Nuance / Dragon
Whisper	Automatic Speech Recognition (ASR, STT)	2022	Transformer encoder–decoder + small CNN front-end	Mel-spectrogram → Text tokens	OpenAI
Flite + HTS	Speech Synthesis (TTS)	2012	“Pre-NN Era”: HSMM + parametric synthesis	Text labels → Phonemes → MFCC/F0 → LPC synthesis	CMU + Nagoya Inst. of Technology (HTS)
HSMM–FCN (MDN-HSMM)	Speech Synthesis (TTS)	2016	“Early NN era”: FCN (MLP) + HSMM + vocoder synthesis	Text labels → Phoneme/state features → Acoustic params	Nagoya Inst. of Technology (HTS)
WaveNet	Speech Synthesis (TTS)	2016	Causal dilated CNN	Text (conditioning) → μ-law 8-bit waveform (PCM)	DeepMind
StyleTTS 2	Speech Synthesis (TTS)	2023	Transformer (text encoder) + CNN (style encoder) + FCNs (duration/prosody) + CNN GAN (decoder/vocoder)	Text → Acoustic features (dur/prosody + style) → Mel-spectrogram	NTU Singapore
Clara	Music Generation (symbolic)	2018	LSTM	MIDI tokens	OpenAI (Christine McLeavey Payne)
Music Transformer	Music Generation (symbolic)	2018	Transformer (relative attention)	MIDI tokens	Google Magenta
SampleRNN	Music Generation (audio)	2017	RNN (hierarchical) + FCN	Quantized waveform samples (μ-law), hierarchical RNN states	Mila (Université de Montréal)
Jukebox	Music Generation (audio)	2020	CNN VQ-VAE + Transformers (priors & upsamplers)	VQ-VAE codes (multi-level)	OpenAI
MusicLM	Music Generation (audio)	2023	Transformer (semantic + acoustic) + CNN AE (SoundStream)	Semantic embeddings (MuLan), Acoustic tokens (SoundStream)	Google Research + IRCAM
MusicGen	Music Generation (audio)	2023	Transformer (generation) + CNN AE (EnCodec)	Acoustic tokens (EnCodec), Conditioning: Text embeddings / acoustic tokens (melody)	Meta AI (FAIR)
Stable Audio 2	Music Generation (audio)	2024	CNN AE + Diffusion with Transformer backbone (DiT)	STFT-based latent embeddings	Stability AI

System

Dragon NaturallySpeaking v.11

Application/Task:

Automatic Speech Recognition (ASR, STT)

Architectures and methods

“Pre-NN Era”

Proprietary - probably GMM-HMM

(Hidden Markov Models + Gaussian Mixtures)

Representations

MFCC → Phonemes → Text

Team

Nuance / Dragon

Year

2010

System
Whisper

Application/Task:

Automatic Speech Recognition (ASR, STT)

Architectures and methods

Transformer encoder–decoder

+ small CNN front-end

Representations

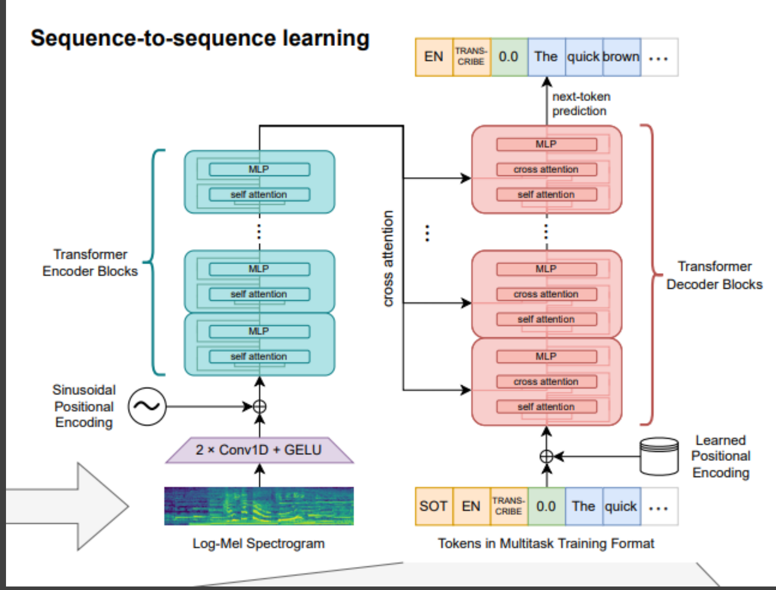
Mel-spectrogram → Text tokens

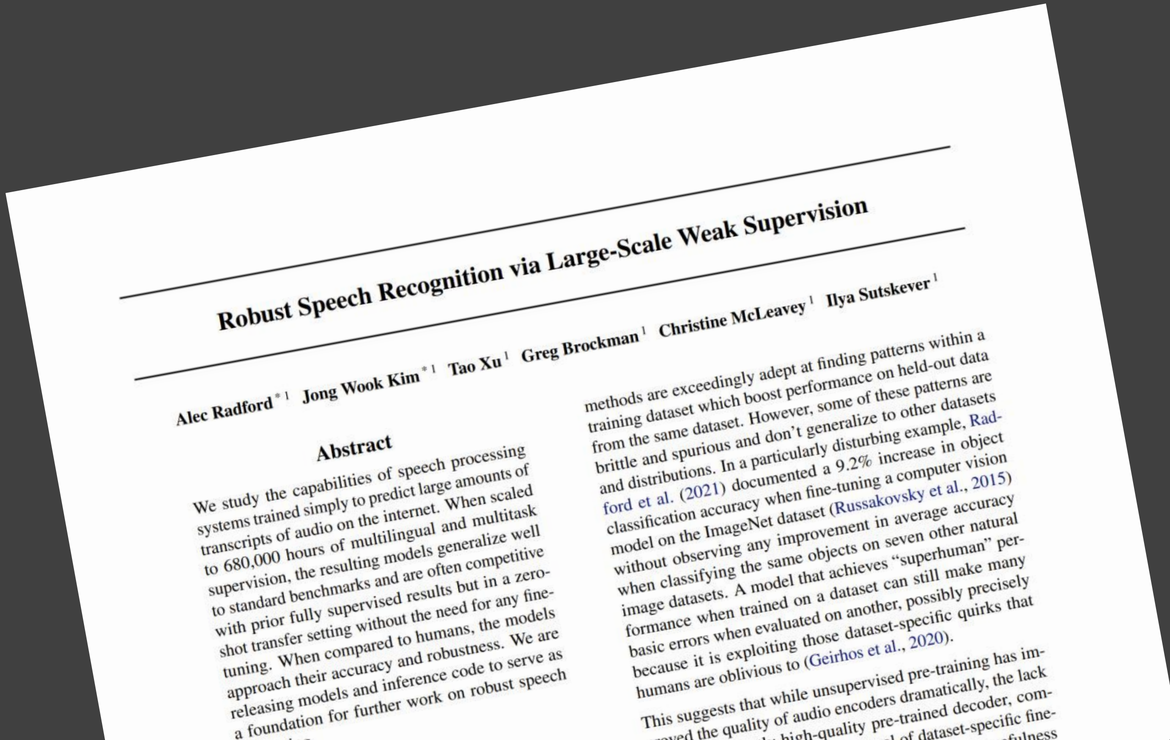
Team

OpenAI

Year

2022





Robust Speech Recognition via Large-Scale Weak Supervision

Alec Radford [*1] Jong Wook Kim [*1] Tao Xu [1] Greg Brockman [1] Christine McLeavey [1] Ilya Sutskever [1]

Abstract

We study the capabilities of speech processing systems trained simply to predict large amounts of transcripts of audio on the internet. When scaled to 680,000 hours of multilingual and multitask supervision, the resulting models generalize well to standard benchmarks and are often competitive with prior fully supervised results but in a zero-shot transfer setting without the need for any fine-tuning. When compared to humans, the models approach their accuracy and robustness. We are releasing models and inference code to serve as a foundation for further work on robust speech

methods are exceedingly adept at finding patterns within a training dataset which boost performance on held-out data from the same dataset. However, some of these patterns are brittle and spurious and don't generalize to other datasets and distributions. In a particularly disturbing example, Radford et al. (2021) documented a 9.2% increase in object classification accuracy when fine-tuning a computer vision model on the ImageNet dataset (Russakovsky et al., 2015) without observing any improvement in average accuracy when classifying the same objects on seven other natural image datasets. A model that achieves "superhuman" performance when trained on a dataset can still make many basic errors when evaluated on another, possibly precisely because it is exploiting those dataset-specific quirks that humans are oblivious to (Geirhos et al., 2020).

This suggests that while unsupervised pre-training has improved the quality of audio encoders dramatically, the lack

System

Flite + HTS

Application/Task:

Speech Synthesis (TTS)

Architectures and methods

“Pre-NN Era” - HSMM + parametric synthesis

Representations

Text labels → Phonemes → MFCC/F0

→ LPC synthesis → Waveform

Team

CMU + Nagoya Inst. of Technology (HTS group)

Year

2012

System

DNN-HSMM

Application/Task:

Speech Synthesis (TTS)

Architectures and methods

“Early NN-era”

FCN (MLP) + HSMM + Vocoder Synthesis)

Representations

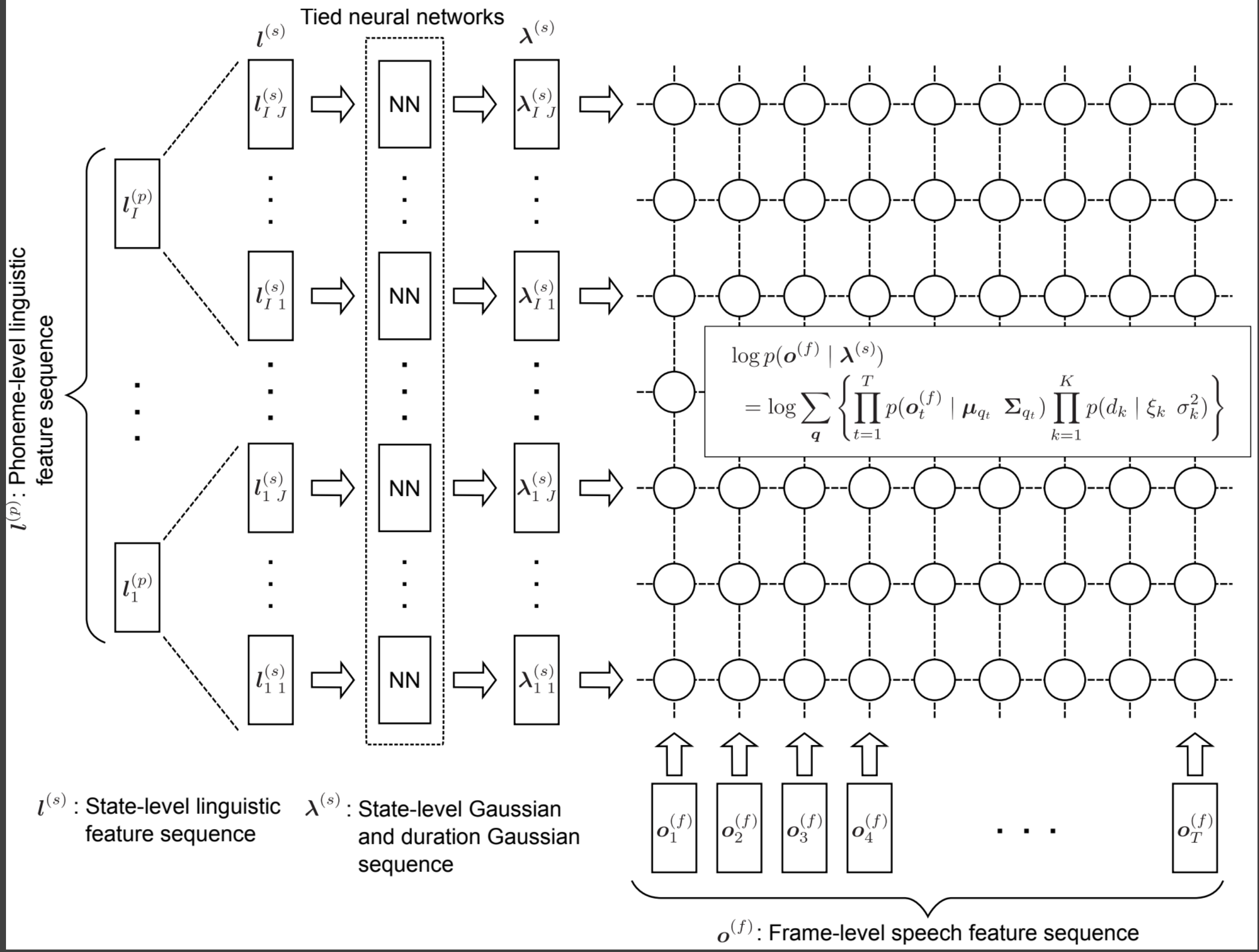
Text labels → Phonemes/state → Acoustic params

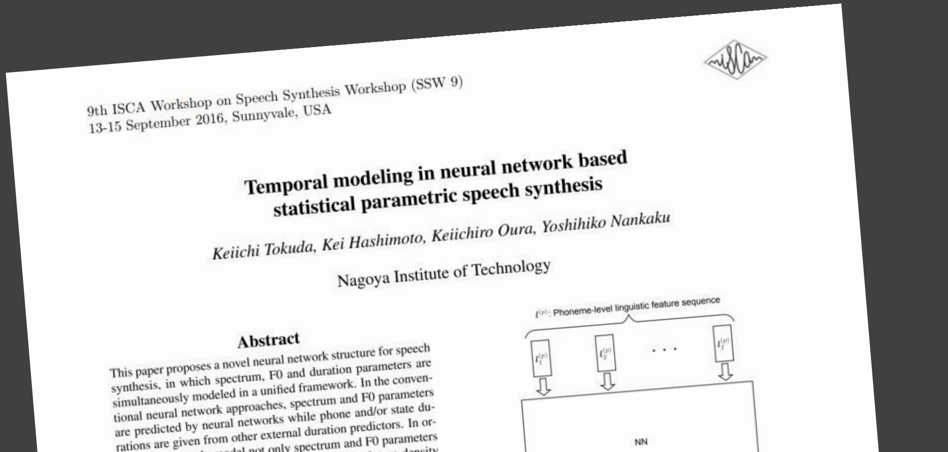
Team

Nagoya Inst. of Technology (HTS group)

Year

2016





9th ISCA Workshop on Speech Synthesis Workshop (SSW 9)
13-15 September 2016, Sunnyvale, USA

Temporal modeling in neural network based statistical parametric speech synthesis

Keiichi Tokuda, Kei Hashimoto, Keiichiro Oura, Yoshihiko Nankaku

Nagoya Institute of Technology

Abstract

This paper proposes a novel neural network structure for speech synthesis, in which spectrum, F0 and duration parameters are simultaneously modeled in a unified framework. In the conventional neural network approaches, spectrum and F0 parameters are predicted by neural networks while phone and/or state durations are given from other external duration predictors. In or-

System

WaveNet

Application/Task:

Speech Synthesis (TTS)

Architectures and methods

Causal Dilated CNN

Representations

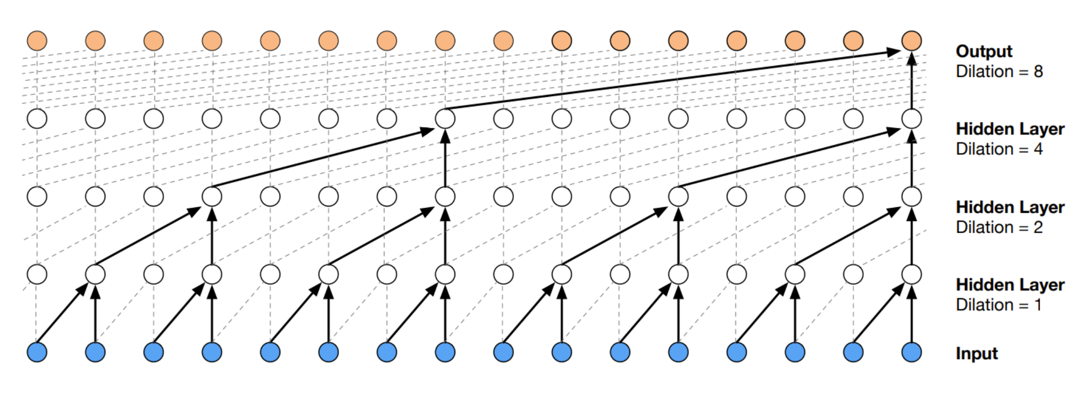
Text (conditioning) → μ-law 8-bit waveform (PCM)

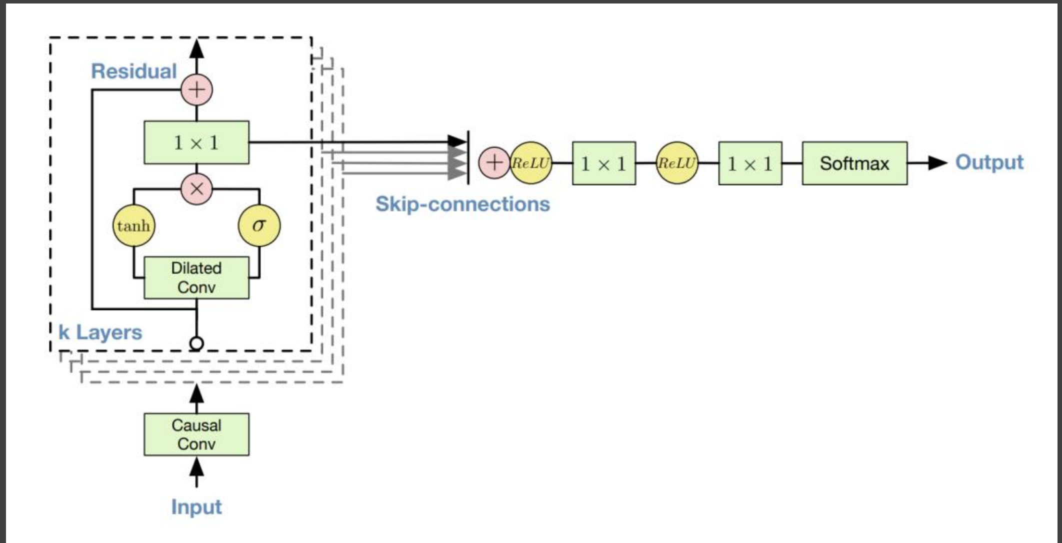
Team

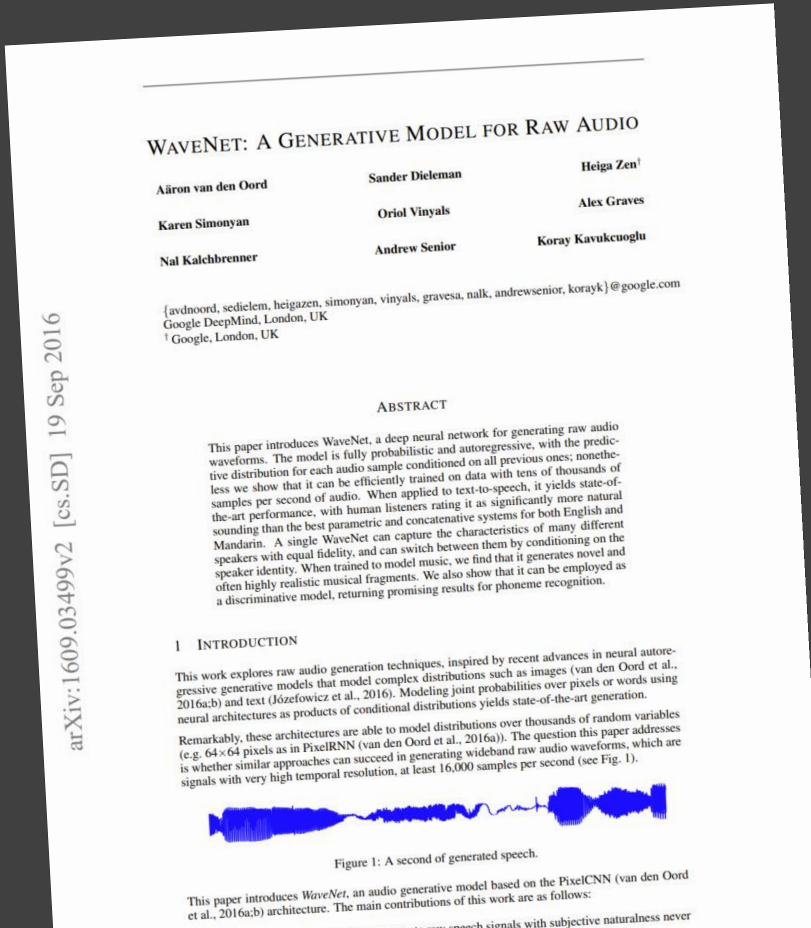
DeepMind

Year

2016







arXiv:1609.03499v2 [cs.SD] 19 Sep 2016

WAVENET: A GENERATIVE MODEL FOR RAW AUDIO

Aäron van den Oord, Sander Dieleman, Heiga Zen†, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, Koray Kavukcuoglu

{avdnoord, sedielem, heigazen, simonyan, vinyals, gravesa, nalk, andrewsenior, korayk}@google.com
Google DeepMind, London, UK
† Google, London, UK

ABSTRACT

This paper introduces WaveNet, a deep neural network for generating raw audio waveforms. The model is fully probabilistic and autoregressive, with the predictive distribution for each audio sample conditioned on all previous ones; nonetheless we show that it can be efficiently trained on data with tens of thousands of samples per second of audio. When applied to text-to-speech, it yields state-of-the-art performance, with human listeners rating it as significantly more natural sounding than the best parametric and concatenative systems for both English and Mandarin. A single WaveNet can capture the characteristics of many different speakers with equal fidelity, and can switch between them by conditioning on the speaker identity. When trained to model music, we find that it generates novel and often highly realistic musical fragments. We also show that it can be employed as a discriminative model, returning promising results for phoneme recognition.

1 INTRODUCTION

This work explores raw audio generation techniques, inspired by recent advances in neural autoregressive generative models that model complex distributions such as images (van den Oord et al., 2016a;b) and text (Józefowicz et al., 2016). Modeling joint probabilities over pixels or words using neural architectures as products of conditional distributions yields state-of-the-art generation.

Remarkably, these architectures are able to model distributions over thousands of random variables (e.g. 64×64 pixels as in PixelRNN (van den Oord et al., 2016a)). The question this paper addresses is whether similar approaches can succeed in generating wideband raw audio waveforms, which are signals with very high temporal resolution, at least 16,000 samples per second (see Fig. 1).

Figure 1: A second of generated speech.

This paper introduces *WaveNet*, an audio generative model based on the PixelCNN (van den Oord et al., 2016a;b) architecture. The main contributions of this work are as follows:

System

StyleTTS 2

Application/Task:

Speech Synthesis (TTS)

Architectures and methods

Hybrid: Transformer (text encoder) + CNN (style encoder) + FCNs (duration & prosody predictors) + CNN GAN (decoder)

Representations

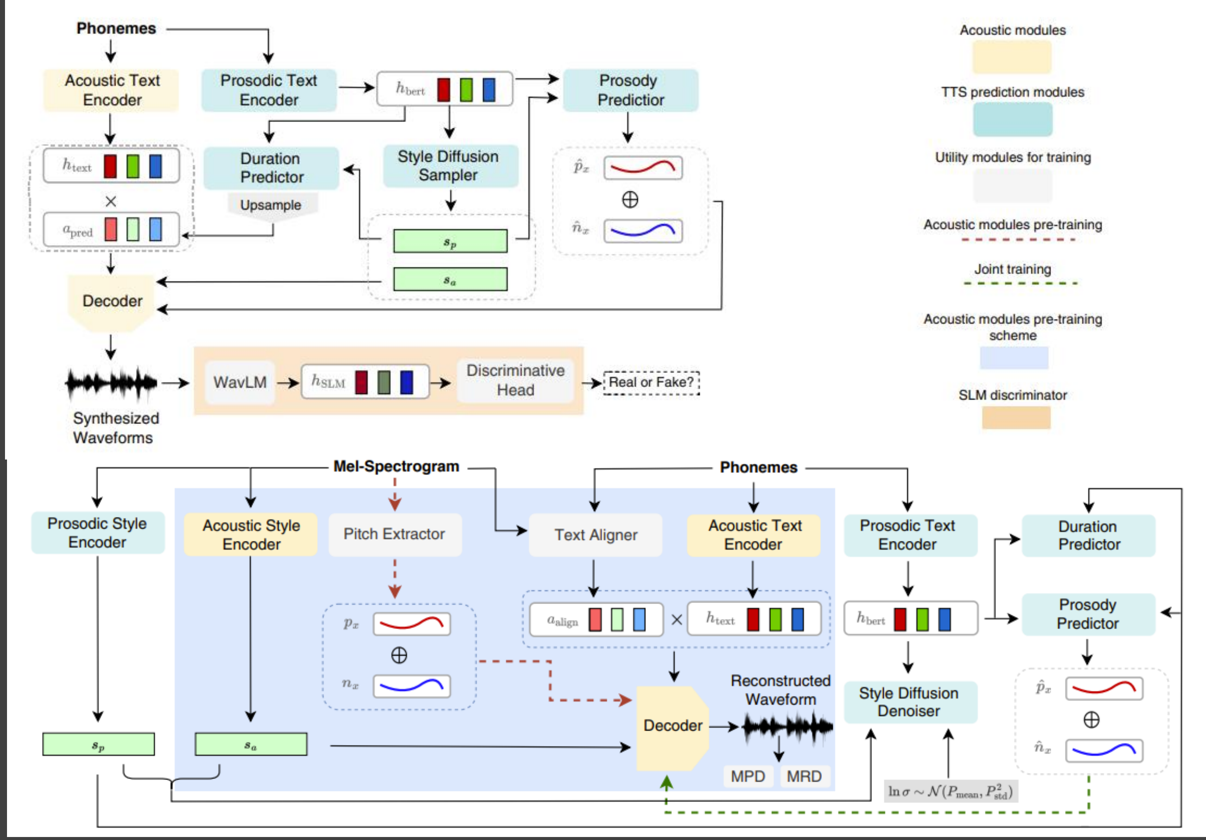
Text → Acoustic features (dur/prosody + style conditioning) → Mel-spectrogram

Team

NTU Singapore

Year

2023



StyleTTS 2: Towards Human-Level Text-to-Speech through Style Diffusion and Adversarial Training with Large Speech Language Models

Yinghao Aaron Li Cong Han Vinay S. Raghavan
Gavin Mischler Nima Mesgarani
Columbia University
{yl4579,ch3212,vsr2119,gm2944,nm2764}@columbia.edu

[...ss.AS] 20 Nov 2023

Abstract

In this paper, we present StyleTTS 2, a text-to-speech (TTS) model that leverages style diffusion and adversarial training with large speech language models (SLMs) to achieve human-level TTS synthesis. StyleTTS 2 differs from its predecessor by modeling styles as a latent random variable through diffusion models to generate the most suitable style for the text without requiring reference speech, achieving efficient latent diffusion while benefiting from the diverse speech synthesis offered by diffusion models. Furthermore, we employ large pre-trained SLMs, such as

System

Clara

Application/Task:

Music Generation (symbolic)

Architectures and methods

LSTM

Representations

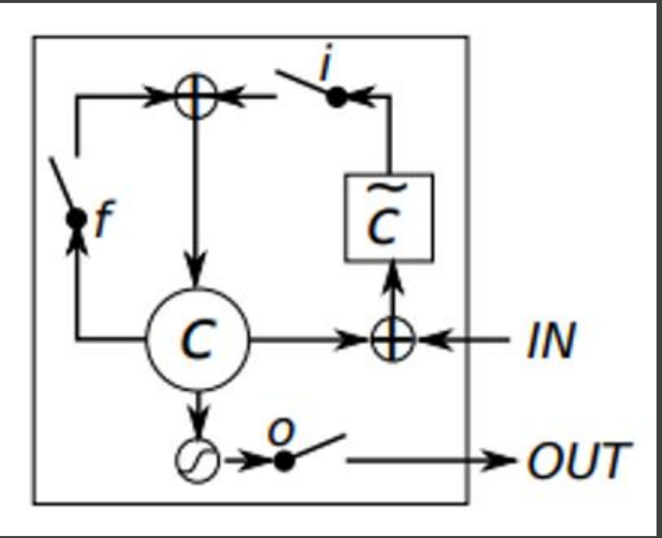
MIDI tokens

Team

Christine McLeavey Payne

Year

2018



System

Music Transformer

Application/Task:

Music Generation (symbolic)

Architectures and methods

Transformer (relative attention)

Representations

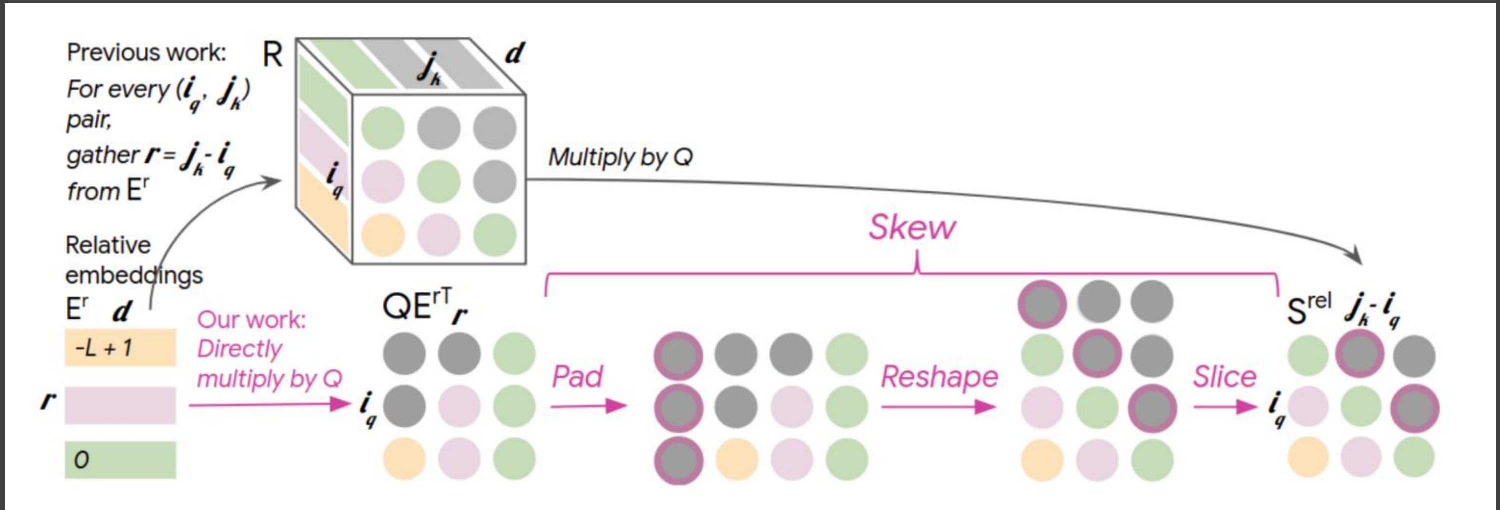
MIDI tokens

Team

Magenta (Google)

Year

2018



MUSIC TRANSFORMER:
GENERATING MUSIC WITH LONG-TERM STRUCTURE

Cheng-Zhi Anna Huang* Ashish Vaswani Jakob Uszkoreit Noam Shazeer
Ian Simon Curtis Hawthorne Andrew M. Dai Matthew D. Hoffman
Monica Dinculescu Douglas Eck
Google Brain

ABSTRACT

Music relies heavily on repetition to build structure and meaning. Self-reference occurs on multiple timescales, from motifs to phrases to reusing of entire sections of music, such as in pieces with ABA structure. The Transformer (Vaswani et al., 2017), a sequence model based on self-attention, has achieved compelling results in many generation tasks that require maintaining long-range coherence. This suggests that self-attention might also be well-suited to modeling music. In musical composition and performance, however, relative timing is critically important. Existing approaches for representing relative positional information in the Transformer modulate attention based on pairwise distance (Shaw et al., 2018). This is impractical for long sequences such as musical compositions since their memory complexity for intermediate relative information is quadratic in the sequence length. We propose an algorithm that reduces their intermediate memory requirement to linear in the sequence length. This enables us to demonstrate that a Transformer with our modified relative attention mechanism can generate minute-long compositions (thousands of steps, four times the length modeled in Oore et al. (2018)) with compelling structure, generate continuations that coherently elaborate on a given motif, and in a seq2seq setup generate accompaniments conditioned on melodies[1]. We evaluate the Transformer with our relative attention mechanism on two datasets, JSB Chorales and Piano-e-Competition, and obtain state-of-the-art results on the latter.

arXiv:1809.04281v3 [cs.LG] 12 Dec 2018

1 INTRODUCTION

System

SampleRNN

Application/Task:

Music generation (audio)

Architectures and methods

RNN (hierarchical) + FCN

Representations

Waveform (μ-law PCM)

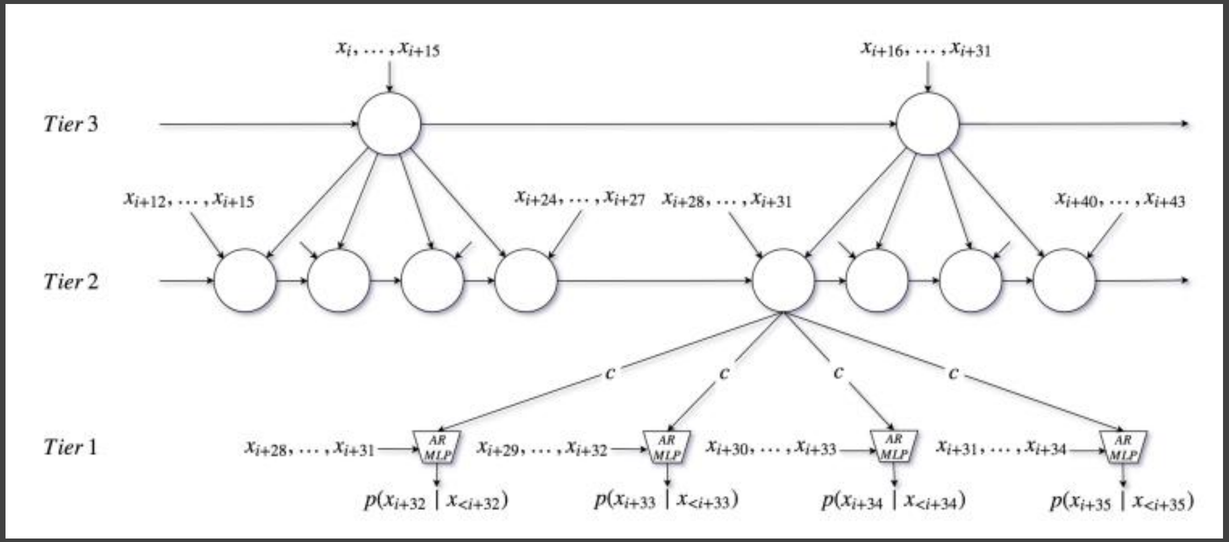
Hierarchical sample-level embeddings

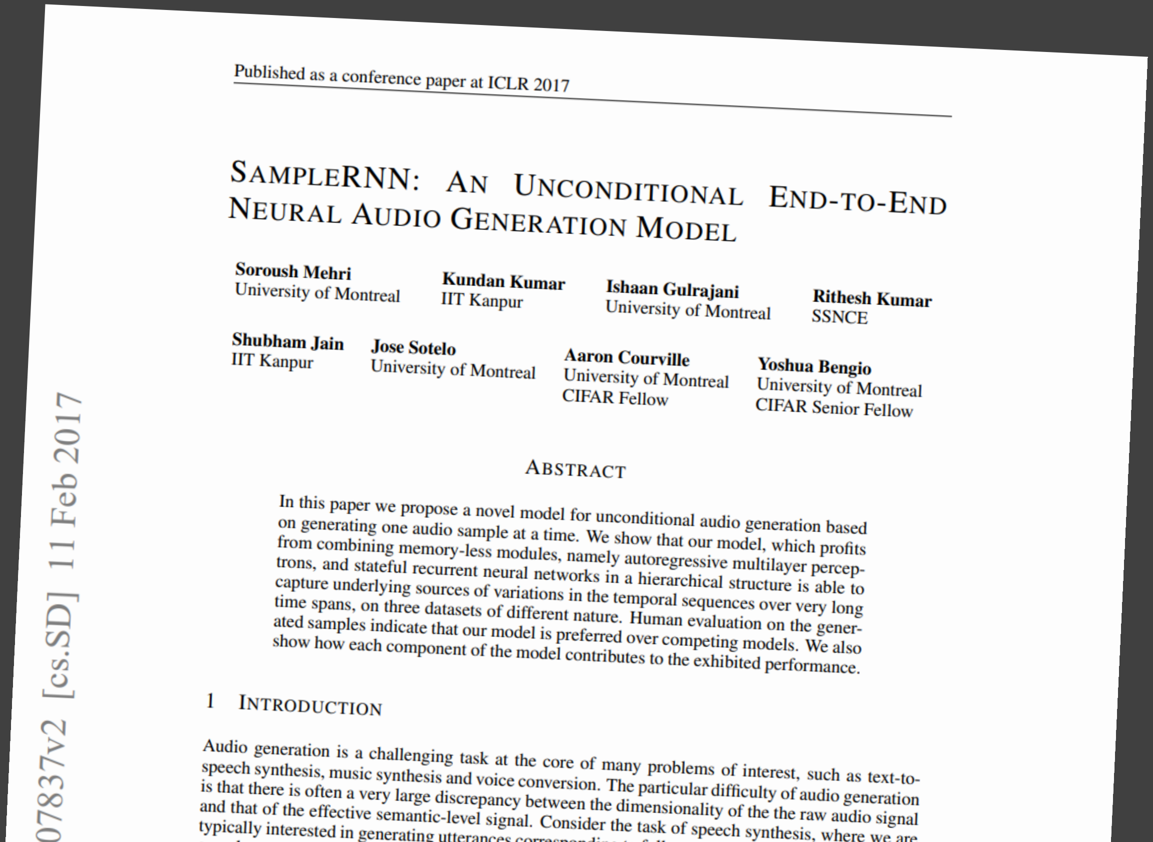
Team

Mila (Université de Montréal)

Year

2017





Published as a conference paper at ICLR 2017

SAMPLERNN: AN UNCONDITIONAL END-TO-END NEURAL AUDIO GENERATION MODEL

Soroush Mehri University of Montreal
Kundan Kumar IIT Kanpur
Ishaan Gulrajani University of Montreal
Rithesh Kumar SSNCE
Shubham Jain IIT Kanpur
Jose Sotelo University of Montreal
Aaron Courville University of Montreal CIFAR Fellow
Yoshua Bengio University of Montreal CIFAR Senior Fellow

ABSTRACT

In this paper we propose a novel model for unconditional audio generation based on generating one audio sample at a time. We show that our model, which profits from combining memory-less modules, namely autoregressive multilayer perceptrons, and stateful recurrent neural networks in a hierarchical structure is able to capture underlying sources of variations in the temporal sequences over very long time spans, on three datasets of different nature. Human evaluation on the generated samples indicate that our model is preferred over competing models. We also show how each component of the model contributes to the exhibited performance.

1 INTRODUCTION

Audio generation is a challenging task at the core of many problems of interest, such as text-to-speech synthesis, music synthesis and voice conversion. The particular difficulty of audio generation is that there is often a very large discrepancy between the dimensionality of the raw audio signal and that of the effective semantic-level signal. Consider the task of speech synthesis, where we are typically interested in generating utterances correspo…

07837v2 [cs.SD] 11 Feb 2017

System

Jukebox

Application/Task:

Music generation (audio)

Architectures and methods

Hybrid:

CNN (encoder/decoder) + Transformer (priors)

Representations

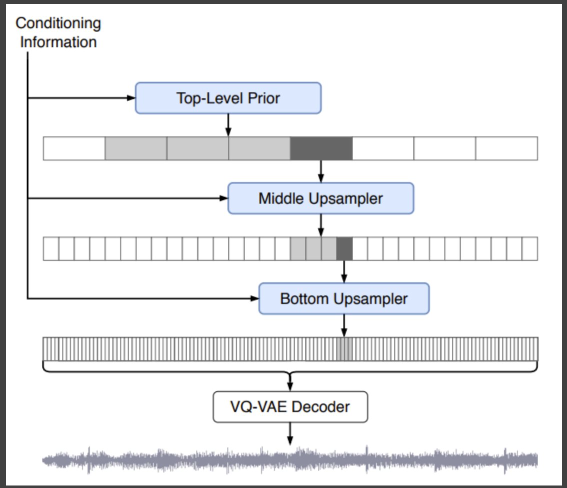
VQ-VAE tokens

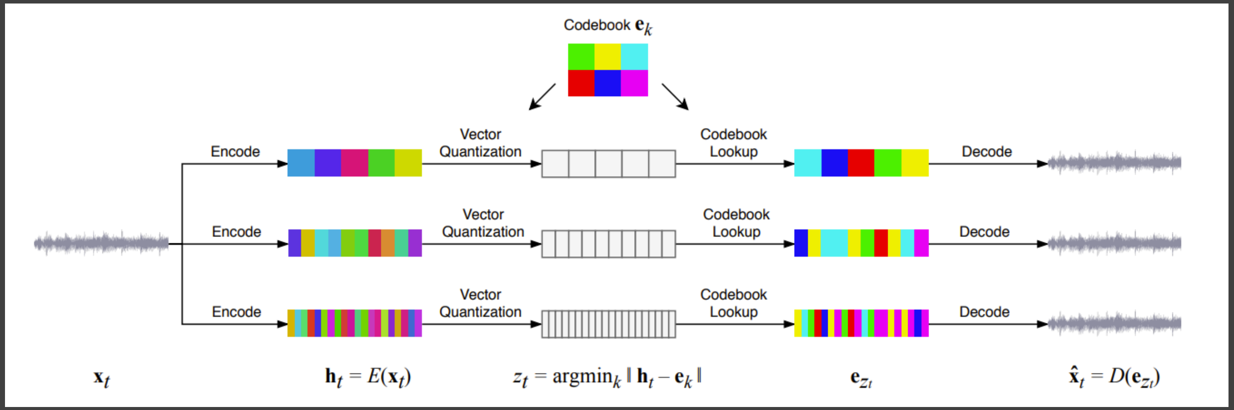
Team

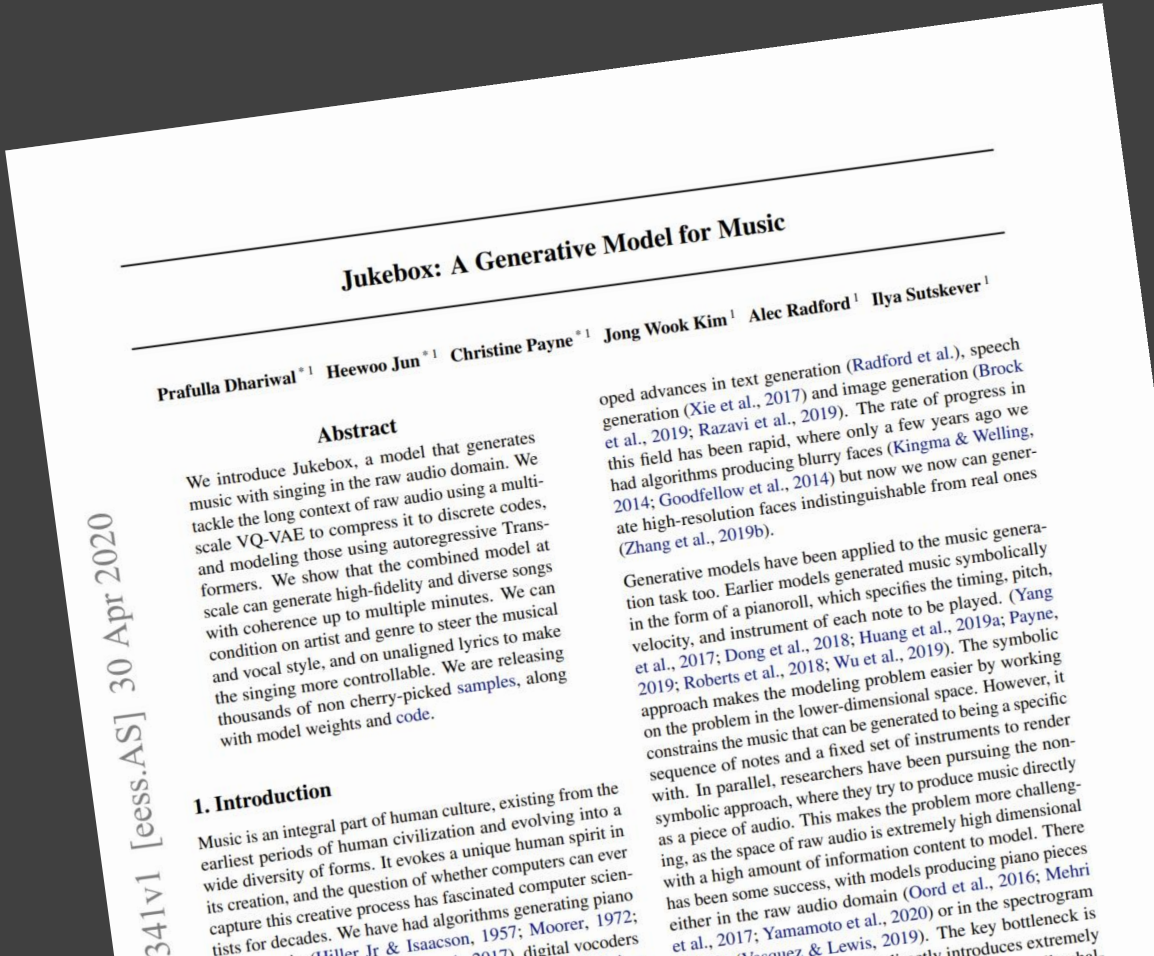
OpenAI

Year

2020







Jukebox: A Generative Model for Music

Prafulla Dhariwal [*1] Heewoo Jun [*1] Christine Payne [*1] Jong Wook Kim [1] Alec Radford [1] Ilya Sutskever [1]

Abstract

We introduce Jukebox, a model that generates music with singing in the raw audio domain. We tackle the long context of raw audio using a multi-scale VQ-VAE to compress it to discrete codes, and modeling those using autoregressive Transformers. We show that the combined model at scale can generate high-fidelity and diverse songs with coherence up to multiple minutes. We can condition on artist and genre to steer the musical and vocal style, and on unaligned lyrics to make the singing more controllable. We are releasing thousands of non cherry-picked samples, along with model weights and code.

1. Introduction

Music is an integral part of human culture, existing from the earliest periods of human civilization and evolving into a wide diversity of forms. It evokes a unique human spirit in its creation, and the question of whether computers can ever capture this creative process has fascinated computer scientists for decades. We have had algorithms generating piano

oped advances in text generation (Radford et al.), speech generation (Xie et al., 2017) and image generation (Brock et al., 2019; Razavi et al., 2019). The rate of progress in this field has been rapid, where only a few years ago we had algorithms producing blurry faces (Kingma & Welling, 2014; Goodfellow et al., 2014) but now we now can generate high-resolution faces indistinguishable from real ones (Zhang et al., 2019b).

Generative models have been applied to the music generation task too. Earlier models generated music symbolically in the form of a pianoroll, which specifies the timing, pitch, velocity, and instrument of each note to be played. (Yang et al., 2017; Dong et al., 2018; Huang et al., 2019a; Payne, 2019; Roberts et al., 2018; Wu et al., 2019). The symbolic approach makes the modeling problem easier by working on the problem in the lower-dimensional space. However, it constrains the music that can be generated to being a specific sequence of notes and a fixed set of instruments to render with. In parallel, researchers have been pursuing the non-symbolic approach, where they try to produce music directly as a piece of audio. This makes the problem more challenging, as the space of raw audio is extremely high dimensional with a high amount of information content to model. There has been some success, with models producing piano pieces either in the raw audio domain (Oord et al., 2016; Mehri et al., 2017; Yamamoto et al., 2020) or in the spectrogram

30 Apr 2020 [eess.AS]

System

MusicLM

Application/Task:

Music generation (audio)

Architectures and methods

Transformer

(+ CNN-based Autoencoder for SoundStream codec)

Representations

Semantic tokens (MuLan)

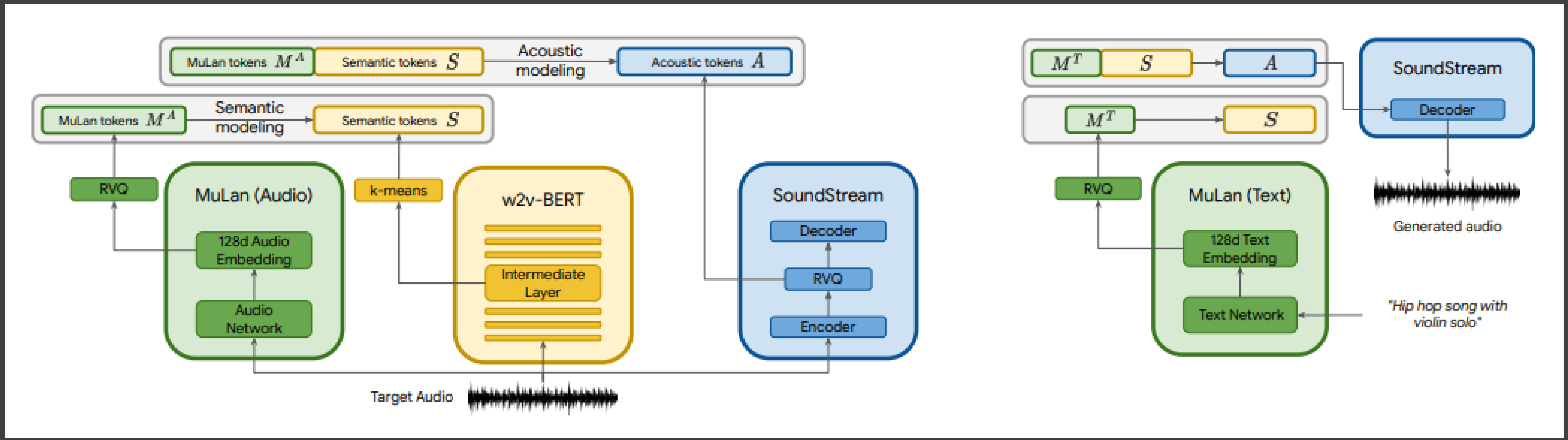
Acoustic tokens (SoundStream)

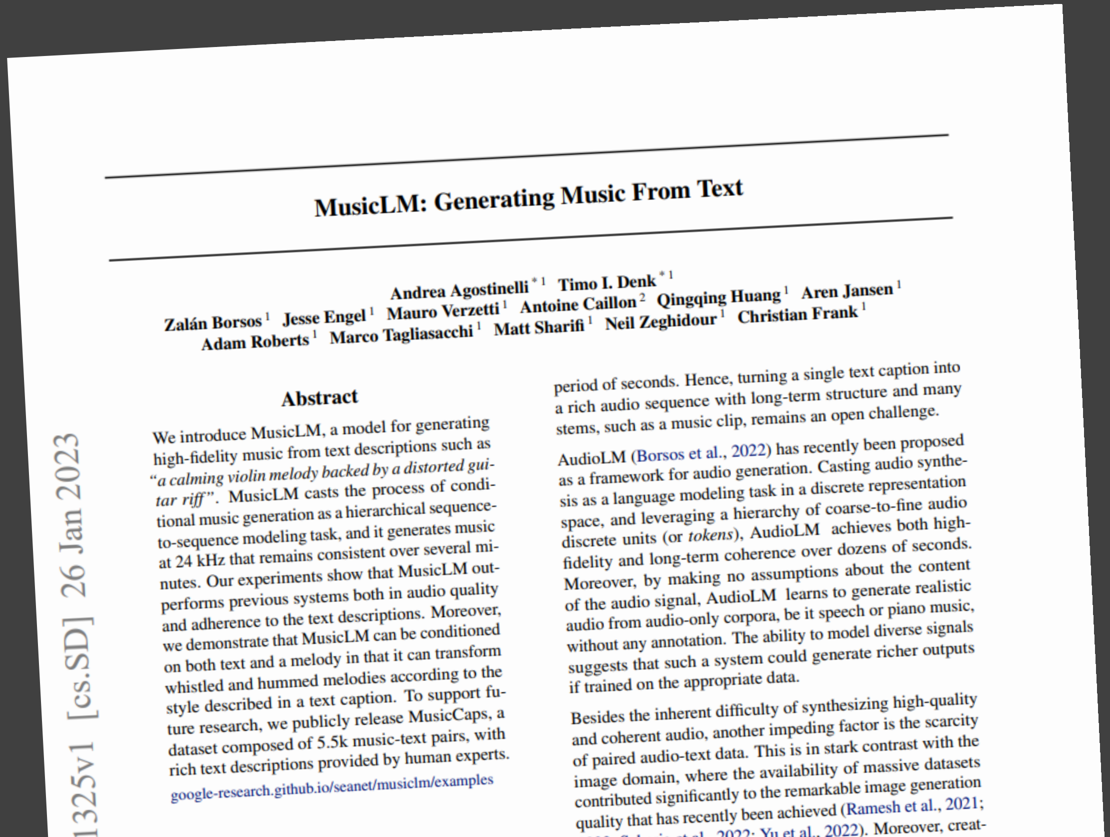
Team

Google Research, IRCAM

Year

2023





MusicLM: Generating Music From Text

Andrea Agostinelli [*1] Timo I. Denk [*1]
Zalán Borsos [1] Jesse Engel [1] Mauro Verzetti [1] Antoine Caillon [2] Qingqing Huang [1] Aren Jansen [1]
Adam Roberts [1] Marco Tagliasacchi [1] Matt Sharifi [1] Neil Zeghidour [1] Christian Frank [1]

Abstract

We introduce MusicLM, a model for generating high-fidelity music from text descriptions such as *"a calming violin melody backed by a distorted guitar riff"*. MusicLM casts the process of conditional music generation as a hierarchical sequence-to-sequence modeling task, and it generates music at 24 kHz that remains consistent over several minutes. Our experiments show that MusicLM outperforms previous systems both in audio quality and adherence to the text descriptions. Moreover, we demonstrate that MusicLM can be conditioned on both text and a melody in that it can transform whistled and hummed melodies according to the style described in a text caption. To support future research, we publicly release MusicCaps, a dataset composed of 5.5k music-text pairs, with rich text descriptions provided by human experts.
google-research.github.io/seanet/musiclm/examples

1325v1 [cs.SD] 26 Jan 2023

period of seconds. Hence, turning a single text caption into a rich audio sequence with long-term structure and many stems, such as a music clip, remains an open challenge.

AudioLM (Borsos et al., 2022) has recently been proposed as a framework for audio generation. Casting audio synthesis as a language modeling task in a discrete representation space, and leveraging a hierarchy of coarse-to-fine audio discrete units (or *tokens*), AudioLM achieves both high-fidelity and long-term coherence over dozens of seconds. Moreover, by making no assumptions about the content of the audio signal, AudioLM learns to generate realistic audio from audio-only corpora, be it speech or piano music, without any annotation. The ability to model diverse signals suggests that such a system could generate richer outputs if trained on the appropriate data.

Besides the inherent difficulty of synthesizing high-quality and coherent audio, another impeding factor is the scarcity of paired audio-text data. This is in stark contrast with the image domain, where the availability of massive datasets contributed significantly to the remarkable image generation quality that has recently been achieved (Ramesh et al., 2021; ... 2022; Yu et al., 2022). Moreover, creat-

System

MusicLM

Application/Task:

Music generation (audio)

Architectures and methods

Transformer (generation)

+ CNN AE (SoundStream codec)

Representations

Semantic tokens/embeddings (MuLan)

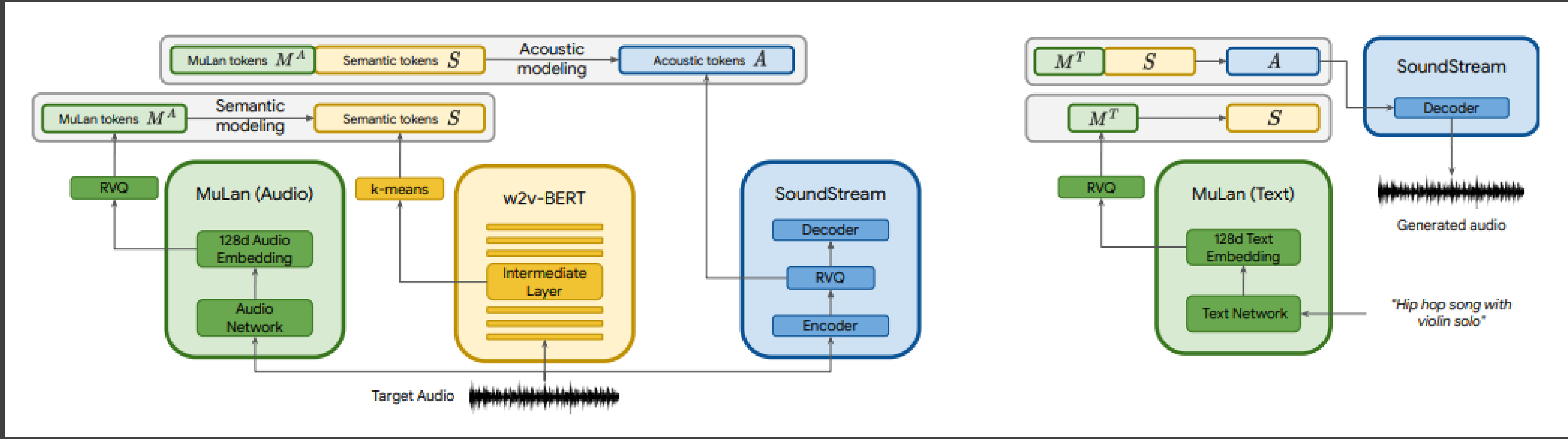
Acoustic tokens (SoundStream)

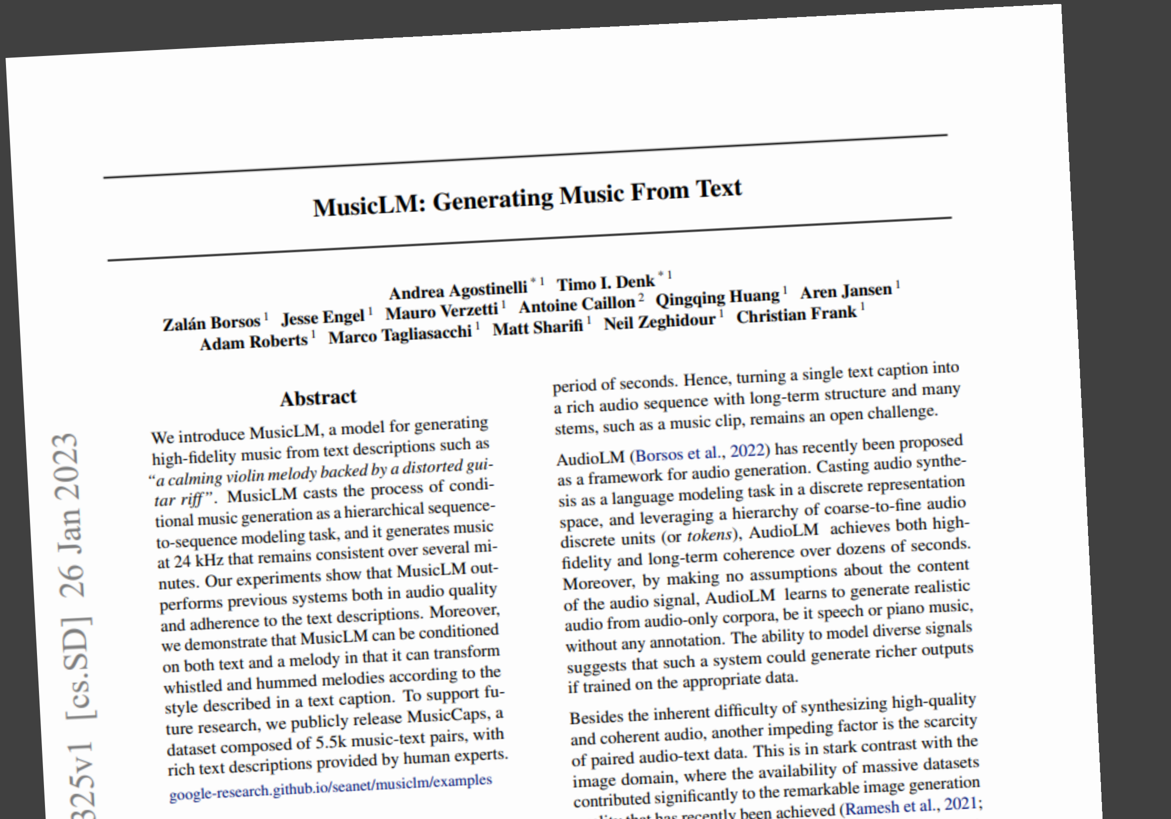
Team

Google Research, IRCAM

Year

2023





MusicLM: Generating Music From Text

Andrea Agostinelli [*1] Timo I. Denk [*1]
Zalán Borsos [1] Jesse Engel [1] Mauro Verzetti [1] Antoine Caillon [2] Qingqing Huang [1] Aren Jansen [1]
Adam Roberts [1] Marco Tagliasacchi [1] Matt Sharifi [1] Neil Zeghidour [1] Christian Frank [1]

Abstract

We introduce MusicLM, a model for generating high-fidelity music from text descriptions such as *"a calming violin melody backed by a distorted guitar riff"*. MusicLM casts the process of conditional music generation as a hierarchical sequence-to-sequence modeling task, and it generates music at 24 kHz that remains consistent over several minutes. Our experiments show that MusicLM outperforms previous systems both in audio quality and adherence to the text descriptions. Moreover, we demonstrate that MusicLM can be conditioned on both text and a melody in that it can transform whistled and hummed melodies according to the style described in a text caption. To support future research, we publicly release MusicCaps, a dataset composed of 5.5k music-text pairs, with rich text descriptions provided by human experts.
google-research.github.io/seanet/musiclm/examples

325v1 [cs.SD] 26 Jan 2023

period of seconds. Hence, turning a single text caption into a rich audio sequence with long-term structure and many stems, such as a music clip, remains an open challenge.

AudioLM (Borsos et al., 2022) has recently been proposed as a framework for audio generation. Casting audio synthesis as a language modeling task in a discrete representation space, and leveraging a hierarchy of coarse-to-fine audio discrete units (or *tokens*), AudioLM achieves both high-fidelity and long-term coherence over dozens of seconds. Moreover, by making no assumptions about the content of the audio signal, AudioLM learns to generate realistic audio from audio-only corpora, be it speech or piano music, without any annotation. The ability to model diverse signals suggests that such a system could generate richer outputs if trained on the appropriate data.

Besides the inherent difficulty of synthesizing high-quality and coherent audio, another impeding factor is the scarcity of paired audio-text data. This is in stark contrast with the image domain, where the availability of massive datasets contributed significantly to the remarkable image generation ... that has recently been achieved (Ramesh et al., 2021;

System

MusicGen

Application/Task:

Music generation (audio)

Architectures and methods

Transformer (generation)

+ CNN AE (EnCodec)

Representations

Acoustic tokens (EnCodec)

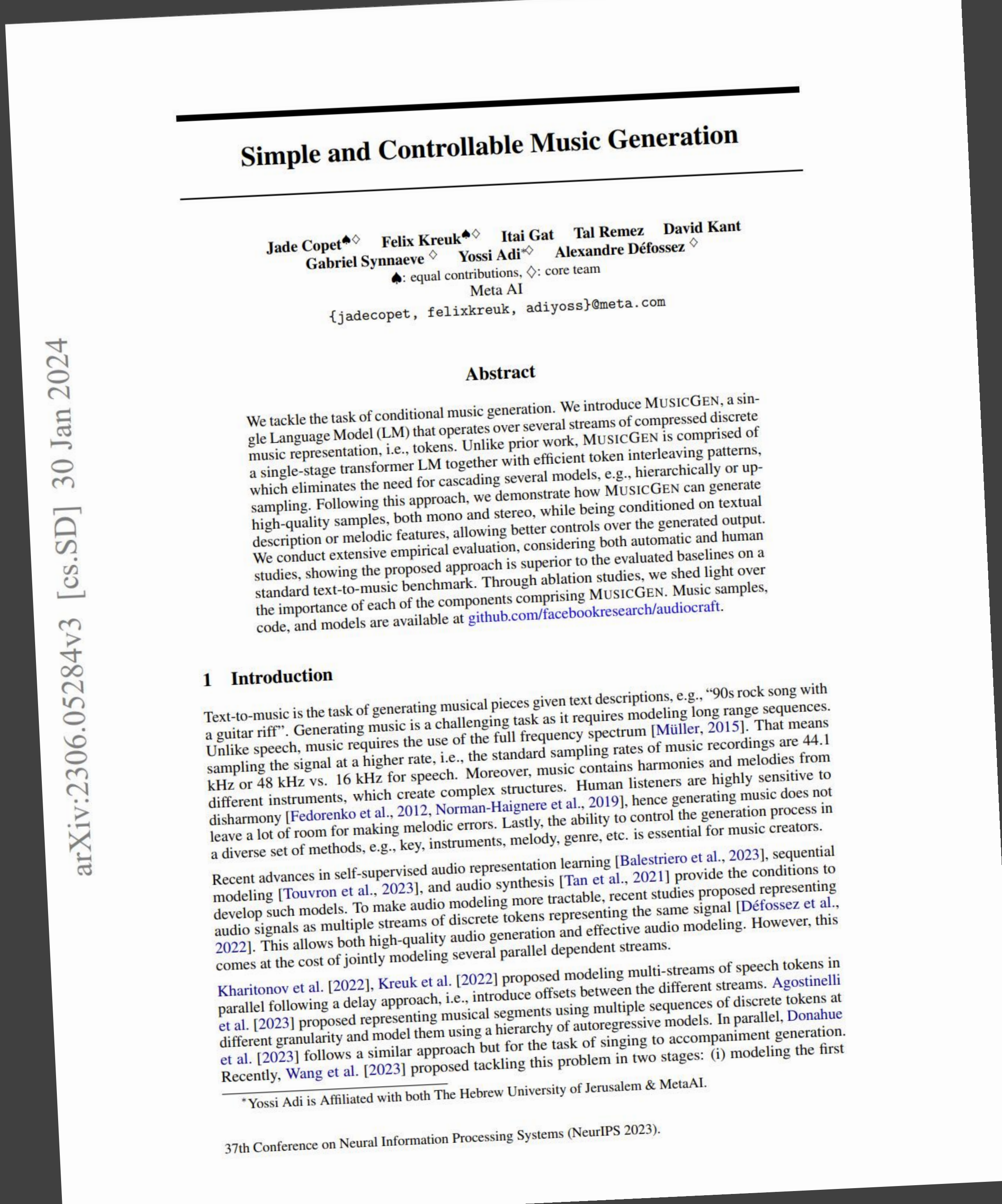
Conditioning: Text embeddings / acoustic tokens

Team

Meta (FAIR)

Year

2023



arXiv:2306.05284v3 [cs.SD] 30 Jan 2024

Simple and Controllable Music Generation

Jade Copet♠◇ Felix Kreuk♠◇ Itai Gat Tal Remez David Kant
Gabriel Synnaeve◇ Yossi Adi*◇ Alexandre Défossez◇
♠: equal contributions, ◇: core team
Meta AI
{jadecopet, felixkreuk, adiyoss}@meta.com

Abstract

We tackle the task of conditional music generation. We introduce MusicGen, a single Language Model (LM) that operates over several streams of compressed discrete music representation, i.e., tokens. Unlike prior work, MusicGen is comprised of a single-stage transformer LM together with efficient token interleaving patterns, which eliminates the need for cascading several models, e.g., hierarchically or upsampling. Following this approach, we demonstrate how MusicGen can generate high-quality samples, both mono and stereo, while being conditioned on textual description or melodic features, allowing better controls over the generated output. We conduct extensive empirical evaluation, considering both automatic and human studies, showing the proposed approach is superior to the evaluated baselines on a standard text-to-music benchmark. Through ablation studies, we shed light over the importance of each of the components comprising MusicGen. Music samples, code, and models are available at github.com/facebookresearch/audiocraft.

1 Introduction

Text-to-music is the task of generating musical pieces given text descriptions, e.g., "90s rock song with a guitar riff". Generating music is a challenging task as it requires modeling long range sequences. Unlike speech, music requires the use of the full frequency spectrum [Müller, 2015]. That means sampling the signal at a higher rate, i.e., the standard sampling rates of music recordings are 44.1 kHz or 48 kHz vs. 16 kHz for speech. Moreover, music contains harmonies and melodies from different instruments, which create complex structures. Human listeners are highly sensitive to disharmony [Fedorenko et al., 2012, Norman-Haignere et al., 2019], hence generating music does not leave a lot of room for making melodic errors. Lastly, the ability to control the generation process in a diverse set of methods, e.g., key, instruments, melody, genre, etc. is essential for music creators.

Recent advances in self-supervised audio representation learning [Balestriero et al., 2023], sequential modeling [Touvron et al., 2023], and audio synthesis [Tan et al., 2021] provide the conditions to develop such models. To make audio modeling more tractable, recent studies proposed representing audio signals as multiple streams of discrete tokens representing the same signal [Défossez et al., 2022]. This allows both high-quality audio generation and effective audio modeling. However, this comes at the cost of jointly modeling several parallel dependent streams.

Kharitonov et al. [2022], Kreuk et al. [2022] proposed modeling multi-streams of speech tokens in parallel following a delay approach, i.e., introduce offsets between the different streams. Agostinelli et al. [2023] proposed representing musical segments using multiple sequences of discrete tokens at different granularity and model them using a hierarchy of autoregressive models. In parallel, Donahue et al. [2023] follows a similar approach but for the task of singing to accompaniment generation. Recently, Wang et al. [2023] proposed tackling this problem in two stages: (i) modeling the first

*Yossi Adi is Affiliated with both The Hebrew University of Jerusalem & MetaAI.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

System

Stable Audio 2

Application/Task:

Music generation (audio)

Architectures and methods

Diffusion with Transformer backbone (DiT) + CNN AE

Representations

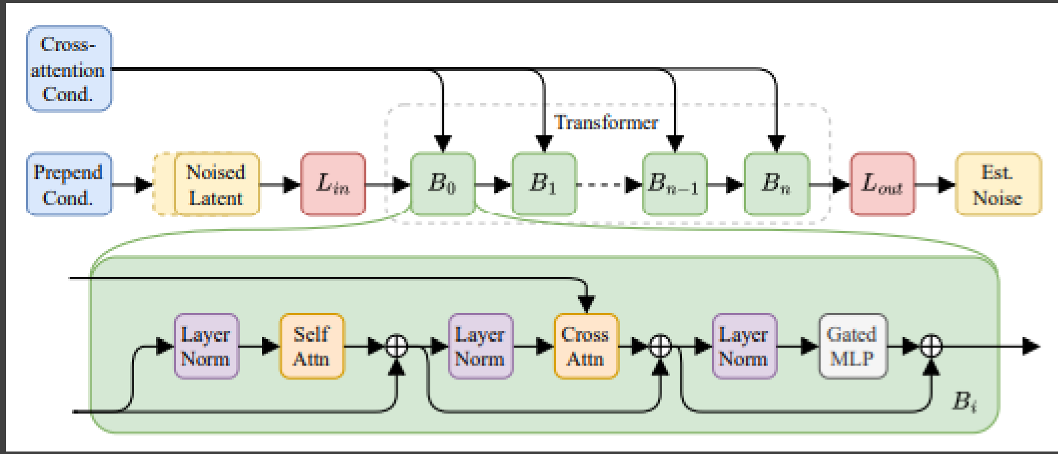
STFT-based latent embeddings

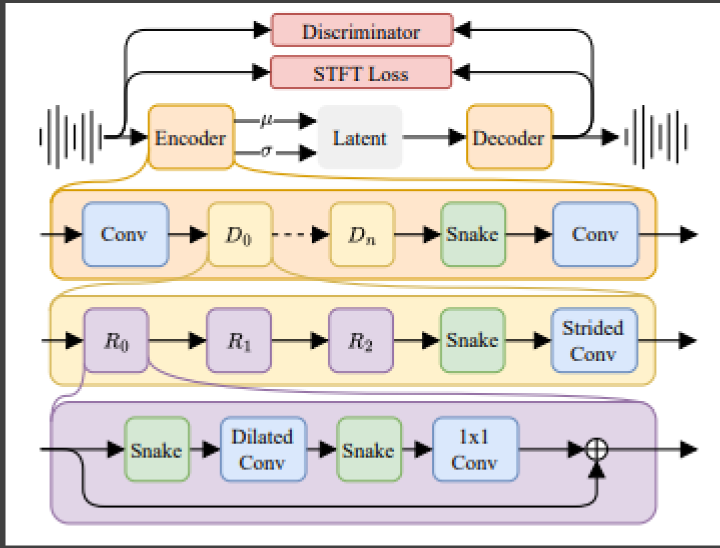
Team

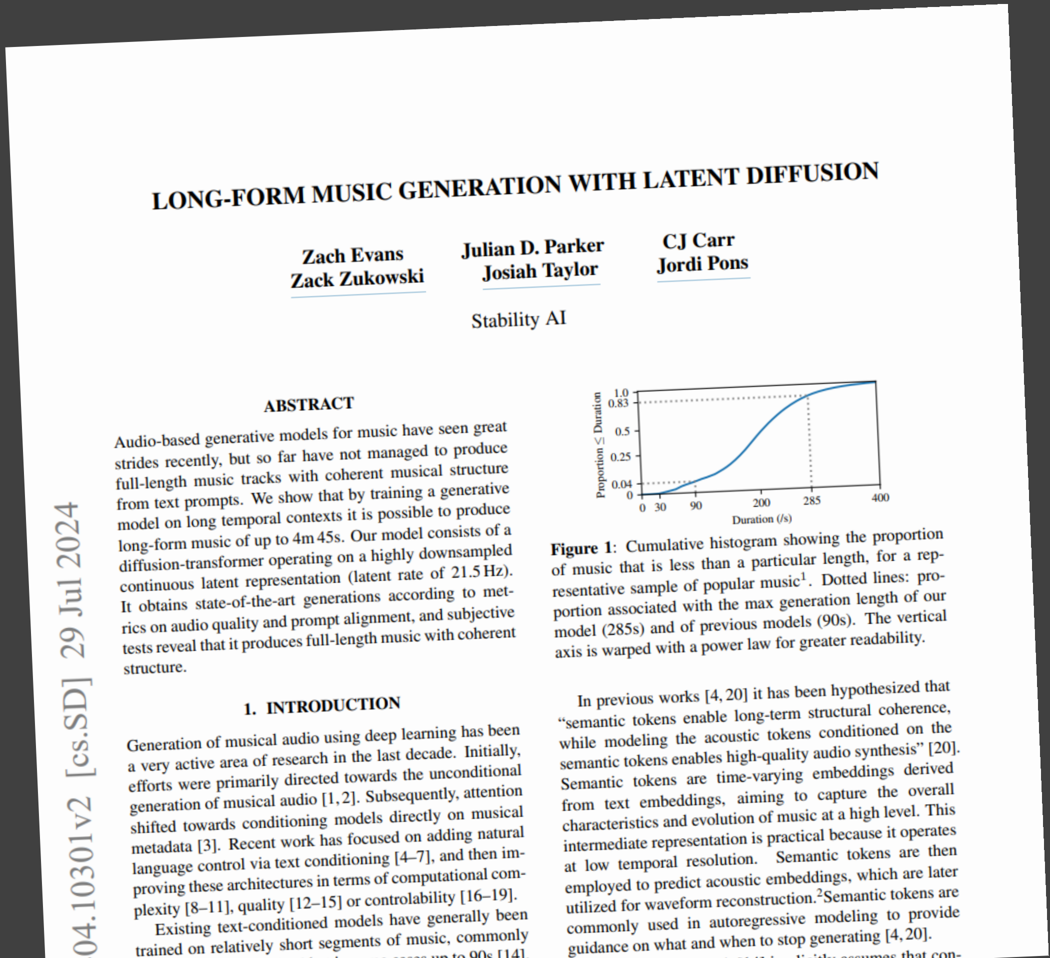
Stability AI

Year

2024







LONG-FORM MUSIC GENERATION WITH LATENT DIFFUSION

Zach Evans
Zack Zukowski
Julian D. Parker
Josiah Taylor
CJ Carr
Jordi Pons

Stability AI

arXiv:2404.10301v2 [cs.SD] 29 Jul 2024

ABSTRACT

Audio-based generative models for music have seen great strides recently, but so far have not managed to produce full-length music tracks with coherent musical structure from text prompts. We show that by training a generative model on long temporal contexts it is possible to produce long-form music of up to 4m 45s. Our model consists of a diffusion-transformer operating on a highly downsampled continuous latent representation (latent rate of 21.5 Hz). It obtains state-of-the-art generations according to metrics on audio quality and prompt alignment, and subjective tests reveal that it produces full-length music with coherent structure.

Figure 1: Cumulative histogram showing the proportion of music that is less than a particular length, for a representative sample of popular music[1]. Dotted lines: proportion associated with the max generation length of our model (285s) and of previous models (90s). The vertical axis is warped with a power law for greater readability.

1. INTRODUCTION

Generation of musical audio using deep learning has been a very active area of research in the last decade. Initially, efforts were primarily directed towards the unconditional generation of musical audio [1,2]. Subsequently, attention shifted towards conditioning models directly on musical metadata [3]. Recent work has focused on adding natural language control via text conditioning [4–7], and then improving these architectures in terms of computational complexity [8–11], quality [12–15] or controlability [16–19].

Existing text-conditioned models have generally been trained on relatively short segments of music, commonly

In previous works [4, 20] it has been hypothesized that "semantic tokens enable long-term structural coherence, while modeling the acoustic tokens conditioned on the semantic tokens enables high-quality audio synthesis" [20]. Semantic tokens are time-varying embeddings derived from text embeddings, aiming to capture the overall characteristics and evolution of music at a high level. This intermediate representation is practical because it operates at low temporal resolution. Semantic tokens are then employed to predict acoustic embeddings, which are later utilized for waveform reconstruction.[2] Semantic tokens are commonly used in autoregressive modeling to provide guidance on what and when to stop generating [4, 20].

List of References for Figures

Modalities and Representation:
Choi, Keunwoo & Fazekas, György & Cho, Kyunghyun & Sandler, Mark. (2017). A Tutorial on Deep Learning for Music Information Retrieval. https://arxiv.org/abs/1709.04396.
McCarthy, R.A., Zhang, Y., Verburg, S.A. et al. Machine Learning in Acoustics: A Review and Open-source Repository. npj Acoust. 1, 18 (2025). https://doi.org/10.1038/s44384-025-00021-w

The Premise:
Photo by Victor Barrios on Unsplash. https://unsplash.com/es/fotos/teclas-blancas-de-piano-zhn3YAFQ-wU

Deepest level:
Malanca, A. (2019, October 14). *Introduction to Artificial Neural Networks*. Telefónica Tech UK Blog. https://telefonicatech.uk/blog/introduction-to-artificial-neural-networks/ https://telefonicatech.uk/wp-content/uploads/adatis/PERCEPTRON-OR-Implementation.gif

FCN (Fully Connected Network, Multi-Layer Perceptron, MLP):
Afan, H., Osman, A., Al-Mahfoodh, A., Essam, Y., Ali Najah Ahmed, Al-Mahfoodh, & Huang, Y., Kisi, O., Sherif, M., Sefelnasr, A., Chau, K., & El-Shafie, A. (2021). *Modeling the fluctuations of groundwater level by employing ensemble deep learning techniques*. Engineering Applications of Computational Fluid Mechanics, 15, 1420-1439. https://doi.org/10.1080/19942060.2021.1974093

Convolutional Neural Network (CNN):
Shenfield, A., & Howarth, M. (2020). *A Novel Deep Learning Model for the Detection and Identification of Rolling Element-Bearing Faults*. Sensors, 20(18), 5112. https://doi.org/10.3390/s20185112

Autoencoder (AE), Variational Autoencoder (VAE): Weng, L. (2018, August 12). *From Autoencoder to Beta-VAE*. Lil'Log. https://lilianweng.github.io/posts/2018-08-12-vae/

U-Net:
Ronneberger, O., Fischer, P., & Brox, T. (2015). *U-Net: Convolutional Networks for Biomedical Image Segmentation*. arXiv preprint arXiv:1505.04597. https://arxiv.org/abs/1505.04597

2-D CNN:
Zilliz. (n.d.). *What is a Convolutional Neural Network? An Engineer's Guide*. Retrieved September 26, 2025, from https://zilliz.com/glossary/convolutional-neural-network

Recurrent Nural Networks (RNN):
GeeksforGeeks. (n.d.). *Introduction to Recurrent Neural Networks*. Retrieved September 16, 2025, from https://www.geeksforgeeks.org/machine-learning/introduction-to-recurrent-neural-network/

LSTM and GRU:
Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). *Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling*. arXiv preprint arXiv:1412.3555. https://arxiv.org/abs/1412.3555

Transformers:
Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). *Attention Is All You Need*. In *Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS'17)*. https://papers.nips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

Whisper Architecture:
Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. *Robust Speech Recognition via Large-Scale Weak Supervision*. https://cdn.openai.com/papers/whisper.pdf

Differential Digital Signal Processing (DDSP):
Engel, J., Hoffman, M., Roberts, A., Eck, D., & Kim, B. (2020). *DDSP: Differentiable Digital Signal Processing*. arXiv preprint arXiv:2001.04643. https://arxiv.org/pdf/2001.04643
Hayes, B., Shier, J., Fazekas, G., McPherson, A., & Saitis, C. (2023). *A review of differentiable digital signal processing for music & speech synthesis*. arXiv preprint arXiv:2308.15422. https://arxiv.org/abs/2308.15422

Diffusion:
Chen, N., Zhang, Y., Zen, H., Weiss, R. J., Norouzi, M., & Chan, W. (2021). *WaveGrad: Estimating Gradients for Waveform Generation*. In *International Conference on Learning Representations (ICLR 2021)*. https://openreview.net/pdf?id=NsMLjcFaO8O
Lemercier, J.-M., Défossez, A., Rodriguez, A., Serizel, R., & Essid, S. (2024). *Diffusion Models for Audio Restoration*. arXiv preprint arXiv:2402.09821. https://arxiv.org/abs/2402.09821