OTH Amberg-Weiden

Advanced Topics in Machine Learning

Winter Semester 2024/2025

Prof. Dr.-Ing. Christian Bergler | OTH Amberg-Weiden

Overview

Topics From Last Time: Introduction & Deep Learning Recap

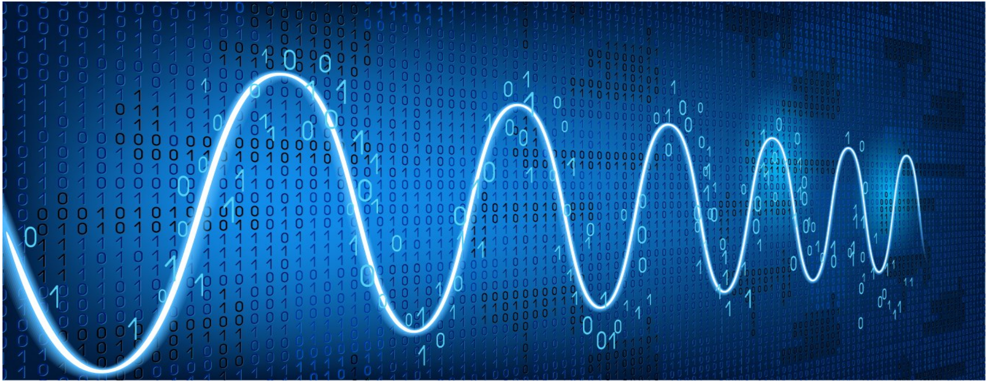
- Recurrent Neural Networks (Forward/Backward Propagation)
- Long Short-Term Memory (LSTM) & Gated Recurrent Unit (GRU) Models
- Attention Mechanism
- Transformer Model

Topics of Today: Acoustic Signal Processing and Multimodal Learning

- Signals and Signal Types
- Audio Signals in Deep Learning
- Analog/Digital (A/D) Conversion (Sampling, Quantization)
- Discrete Fourier Transform (DFT)
- Short-Time Fourier Transform (STFT)
- Spectrogram
- Multimodal Learning

Signal Processing

Signals – What is it about?



- Signal: a real-world signal refers to any physical or abstract quantity, as part of different fields, considered as a function f, which conveys information about the behavior or respective state in a physical system

Source: Image taken from https://www.enclustra.com/en/design-services/digital-signal-processing/

Signal Processing

Signals – Different Fields

- Electronics & Communication: a signal refers to an electrical or electromagnetic representation of data, which can vary over time and can be classified into analog (continuous) & digital signals (discrete)

Signals – Different Fields

OTH Amberg-Weiden

- Electronics & Communication: a signal refers to an electrical or electromagnetic representation of data, which can vary over time and can be classified into analog (continuous) & digital signals (discrete)
- Physics: a signal represents a physical quantity which changes over time, such as sound waves, light waves, or other forms of waves that can carry information

Signal Processing

Signals – Different Fields

- Electronics & Communication: a signal refers to an electrical or electromagnetic representation of data, which can vary over time and can be classified into analog (continuous) & digital signals (discrete)
- Physics: a signal represents a physical quantity which changes over time, such as sound waves, light waves, or other forms of waves that can carry information
- Computer Science: a (bit-wise) signal presents interruptions or notifications that a process or program receives to take a certain action or follow a specific behavior

Signal Processing

Signals – Different Fields

- Electronics & Communication: a signal refers to an electrical or electromagnetic representation of data, which can vary over time and can be classified into analog (continuous) & digital signals (discrete)
- Physics: a signal represents a physical quantity which changes over time, such as sound waves, light waves, or other forms of waves that can carry information
- Computer Science: a (bit-wise) signal presents interruptions or notifications that a process or program receives to take a certain action or follow a specific behavior
- Data Science & Machine Learning: a signal refers to the true, useful information present in a dataset, as opposed to noise, which represents random errors or irrelevant data

Signal Processing

Signals – Different Fields

- Electronics & Communication: a signal refers to an electrical or electromagnetic representation of data, which can vary over time and can be classified into analog (continuous) & digital signals (discrete)
- Physics: a signal represents a physical quantity which changes over time, such as sound waves, light waves, or other forms of waves that can carry information
- Computer Science: a (bit-wise) signal presents interruptions or notifications that a process or program receives to take a certain action or follow a specific behavior
- Data Science & Machine Learning: a signal refers to the true, useful information present in a dataset, as opposed to noise, which represents random errors or irrelevant data
- Natural Language Processing: signal describes patterns or features in text data which helps to identify relevant information, such as keywords, sentiment indicators, or linguistic structures

Signal Processing

Signals – Different Types

- Image Signals: Represented as a matrix of pixel values that vary in intensity and color, forming a visual representation. Images are typically considered two-dimensional signals

Signal Processing

Signals – Different Types

- Image Signals: Represented as a matrix of pixel values that vary in intensity and color, forming a visual representation. Images are typically considered two-dimensional signals
- Sensor Signals: Data from sensors (such as temperature, pressure, motion sensors) that continuously measure physical phenomena and convert them into electrical signals or data streams for downstream analysis

Signal Processing

Signals – Different Types

- Image Signals: Represented as a matrix of pixel values that vary in intensity and color, forming a visual representation. Images are typically considered two-dimensional signals
- Sensor Signals: Data from sensors (such as temperature, pressure, motion sensors) that continuously measure physical phenomena and convert them into electrical signals or data streams for downstream analysis
- Text Signals: Text is a sequence of characters or words that conveys information in a structured form, carrying linguistic, syntactic, and semantic patterns

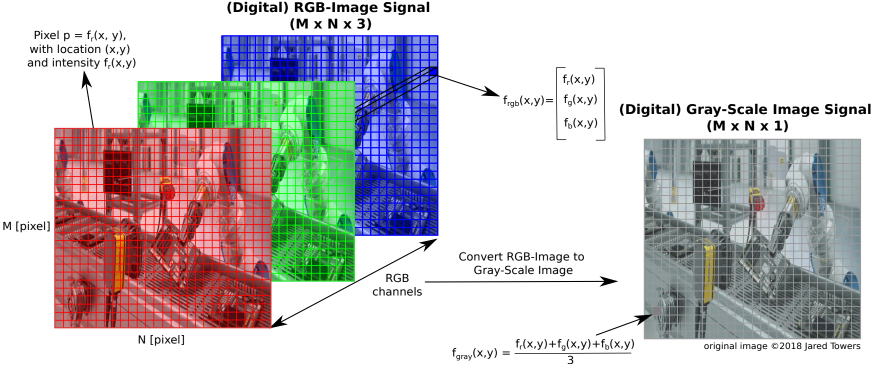
Signal Processing

Signals – Different Types

- Image Signals: Represented as a matrix of pixel values that vary in intensity and color, forming a visual representation. Images are typically considered two-dimensional signals
- Sensor Signals: Data from sensors (such as temperature, pressure, motion sensors) that continuously measure physical phenomena and convert them into electrical signals or data streams for downstream analysis
- Text Signals: Text is a sequence of characters or words that conveys information in a structured form, carrying linguistic, syntactic, and semantic patterns
- Audio Signals: Represented as waveforms that vary over time, capturing sound. They are continuous signals (analog) or can be digitized (digital), such as in speech or music

Signal Processing

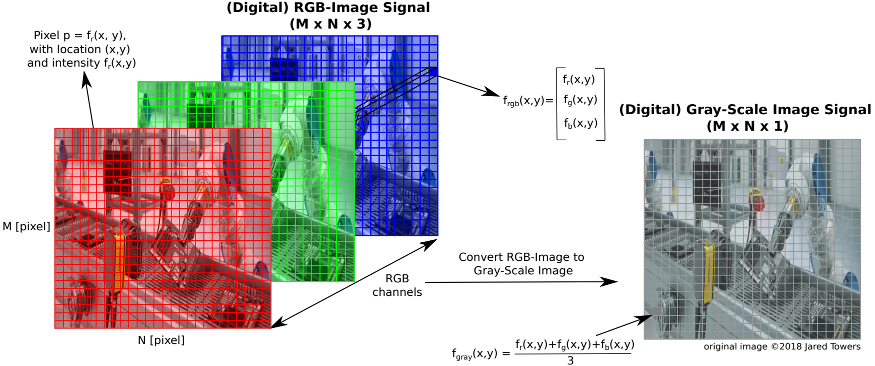
Signals – Different Types – (Digital) Image



- 2D-Representation: spatial coordinates (x, y), intensity value $f(x, y)$ (=pixel)

Signal Processing

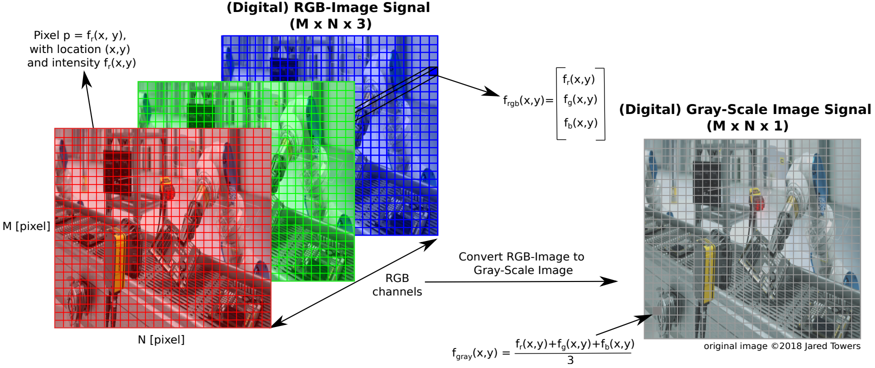
Signals – Different Types – (Digital) Image



- 2D-Representation: spatial coordinates (x, y), intensity value $f(x, y)$ (=pixel)
- $x, y \in \mathbb{Z}^2$, Gray-Scale $f(x, y) = \vec{y}_{1\times1}$ (scalar!) $= [f_g(x, y)]$, Color-Scale $f(x, y) = \vec{y}_{3\times1} = [f_r(x, y), f_g(x, y), f_b(x, y)]^T$

Signal Processing

Signals – Different Types – (Digital) Image



- 2D-Representation: spatial coordinates (x, y), intensity value $f(x, y)$ (=pixel)
- $x, y \in \mathbb{Z}^2$, Gray-Scale $f(x, y) = \vec{y}_{1\times1}$ (scalar!) $= [f_g(x, y)]$, Color-Scale $f(x, y) = \vec{y}_{3\times1} = [f_r(x, y), f_g(x, y), f_b(x, y)]^T$
- Digital RGB-Image $(M \times N \times 3)$, Gray-Scale Image $(M \times N \times 1)$

Signal Processing

Signals – Different Types – Sensor

- Sensor signals measure physical quantities like temperature, distance, light, and others

Source: Image from https://www.volersystems.com/blog/understanding-sensor-signal-conditioning-for-precise-data-acquisition

Signal Processing

Signals – Different Types – Sensor

- Sensor signals measure physical quantities like temperature, distance, light, and others
- Mathematically it describes a function over time $s(t)$, with time t and amplitude $s(t)$

Source: Image from https://www.volersystems.com/blog/understanding-sensor-signal-conditioning-for-precise-data-acquisition

Signal Processing

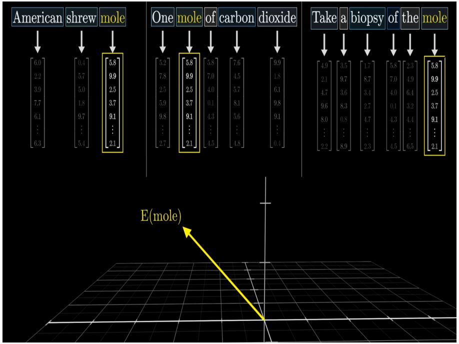
Signals – Different Types – Sensor

OTH Amberg-Weiden

- Sensor signals measure physical quantities like temperature, distance, light, and others
- Mathematically it describes a function over time $s(t)$, with time t and amplitude $s(t)$
- In machine (deep) learning it is also known as Time Series (data) or a continuous function comprising readings from a sensor (digitized via sampling & quantization)

Source: Image from https://www.volersystems.com/blog/understanding-sensor-signal-conditioning-for-precise-data-acquisition

- Text as String (sequence of words & characters)

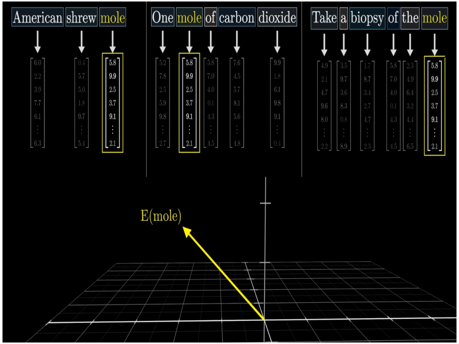


Source: Images taken from YouTube 3Brown1Blue – https://www.youtube.com/watch?v=eMlx5fFNoYc

Signal Processing

Signals – Different Types – Text

- Text as String (sequence of words & characters)
- Word Embedding describes words as numerical vectors in a continuous and multi-dimensional space

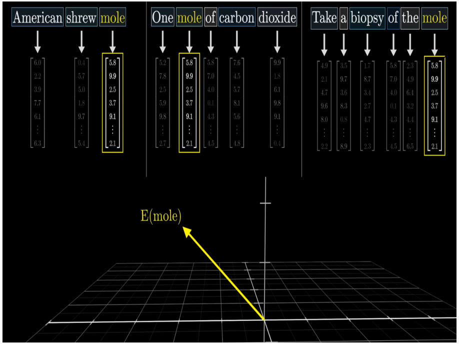


Source: Images taken from YouTube 3Brown1Blue – https://www.youtube.com/watch?v=eMlx5fFNoYc

Signal Processing

Signals – Different Types – Text

- Text as String (sequence of words & characters)
- Word Embedding describes words as numerical vectors in a continuous and multi-dimensional space
- Word Similarity describes vectors which present related directions and magnitudes

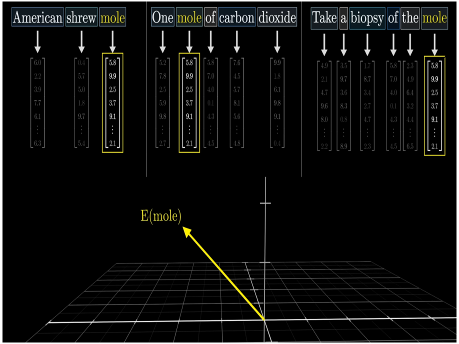


Source: Images taken from YouTube 3Brown1Blue – https://www.youtube.com/watch?v=eMlx5fFNoYc

Signal Processing

Signals – Different Types – Text

- Text as String (sequence of words & characters)
- Word Embedding describes words as numerical vectors in a continuous and multi-dimensional space
- Word Similarity describes vectors which present related directions and magnitudes
- Similarity: Dot-product ($\vec{u} \cdot \vec{v} = \sum_{i=1}^{N} u_i v_i$) $\rightarrow$ Indicator of a high similarity ($>>$) for long vectors $\rightarrow$ Cosine Similarity (only directional!)

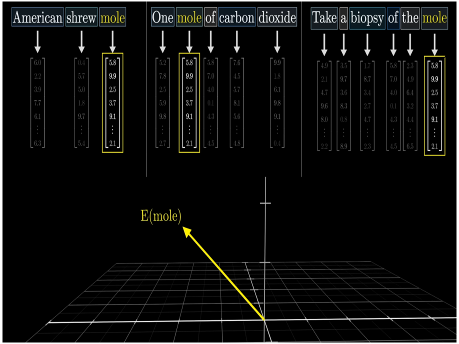


Source: Images taken from YouTube 3Brown1Blue – https://www.youtube.com/watch?v=eMlx5fFNoYc

Signal Processing

Signals – Different Types – Text

- Text as String (sequence of words & characters)
- Word Embedding describes words as numerical vectors in a continuous and multi-dimensional space
- Word Similarity describes vectors which present related directions and magnitudes
- Similarity: Dot-product ($\vec{u} \cdot \vec{v} = \sum_{i=1}^{N} u_i v_i$) $\rightarrow$ Indicator of a high similarity ($>>$) for long vectors $\rightarrow$ Cosine Similarity (only directional!)
  - $cos(\Theta) = cos(\vec{u}, \vec{v}) = \frac{\vec{u}\,\vec{v}}{||\vec{u}||\;||\vec{v}||} = \frac{\sum_{i=1}^{N} u_i v_i}{\sqrt{\sum_{i=1}^{N} u_i^2}\,\sqrt{\sum_{i=1}^{N} v_i^2}}$
  - $cos(\Theta) = cos(\vec{u}, \vec{v}) = 1 \rightarrow$ same direction!
  - $cos(\Theta) = cos(\vec{u}, \vec{v}) = 0 \rightarrow$ orthogonal!
  - $cos(\Theta) = cos(\vec{u}, \vec{v}) = -1 \rightarrow$ opposite direction!



Source: Images taken from YouTube 3Brown1Blue – https://www.youtube.com/watch?v=eMlx5fFNoYc

Signal Processing

Signals – Different Types – Audio

- Acoustic signals (audio) $f(t)$ describe sound waves, which are pressure variations in the air over time t (time Series), leading to a pressure-time graph, also known as waveform

Source: Image from https://www.lafilm.edu/blog/the-importance-of-sound/

Signal Processing

Signals – Different Types – Audio

- Acoustic signals (audio) $f(t)$ describe sound waves, which are pressure variations in the air over time t (time Series), leading to a pressure-time graph, also known as waveform
- An audio signal is a real-valued representation (analog vs. digital)
  - Analog: raw and unprocessed waveforms as they appear in reality

Source: Image from https://www.lafilm.edu/blog/the-importance-of-sound/

Signal Processing

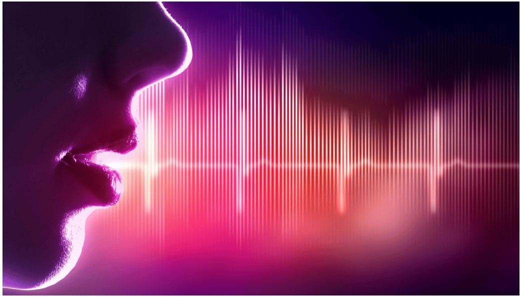
Signals – Different Types – Audio

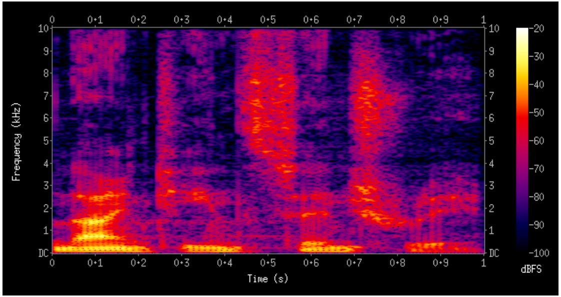
- Acoustic signals (audio) $f(t)$ describe sound waves, which are pressure variations in the air over time t (time Series), leading to a pressure-time graph, also known as waveform
- An audio signal is a real-valued representation (analog vs. digital)
  - Analog: raw and unprocessed waveforms as they appear in reality
  - Digital: transformed and preprocessed signals for machine interpretation

Source: Image from https://www.lafilm.edu/blog/the-importance-of-sound/

Acoustic Signal Processing

Audio Signals (Waveform, Spectrogram)



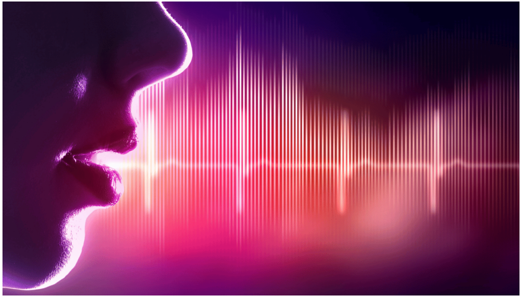


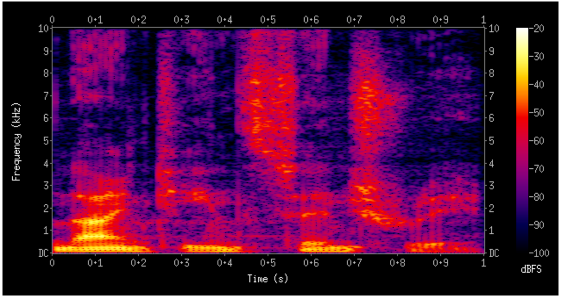
- Acoustic signal in the format of a waveform (time-domain representation of the acoustic signal with the amplitude $f(t)$ change over time t) $\rightarrow 1 \times N$ ($N =$ sampling points)

Source: https://www.levelsmusicproduction.com/blog/unleashing-the-power-of-sound-9-characteristics-of-a-sound-wave
Source: https://en.wikipedia.org/wiki/Spectrogram

Acoustic Signal Processing

Audio Signals (Waveform, Spectrogram)





- Acoustic signal in the format of a waveform (time-domain representation of the acoustic signal with the amplitude $f(t)$ change over time t) $\rightarrow 1 \times N$ ($N =$ sampling points)
- Acoustic signal in the format of a spectrogram (time-frequency representation of the acoustic signal with the amplitude/power of the frequency $f(\omega)$ as coloring) $\rightarrow$ $T \times F \times 1$ ($T =$ time, $F =$ frequency) or complex variant ($T \times F \times 2$ (Re, Im))

Source: https://www.levelsmusicproduction.com/blog/unleashing-the-power-of-sound-9-characteristics-of-a-sound-wave
Source: https://en.wikipedia.org/wiki/Spectrogram

Acoustic Signal Processing

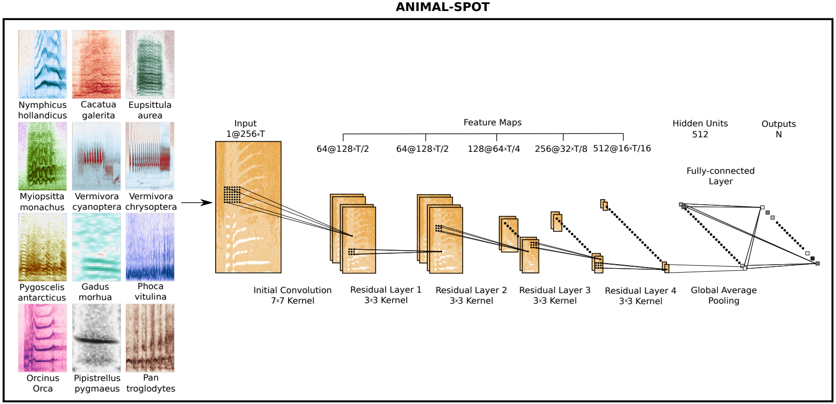
Audio Signals in Deep Learning – Applications

Automatic Speech Recognition (ASR)

- $P(W|A) = \frac{P(A|W)\ P(W)}{P(A)}$, with $P(A|W)$=Acoustic Model, $P(W)$=Language Model

Source: Image taken from https://www.iosb.fraunhofer.de/en/competences/image-exploitation/interactive-analysis-diagnosis/explainable-ai.html

Audio Signal Classification



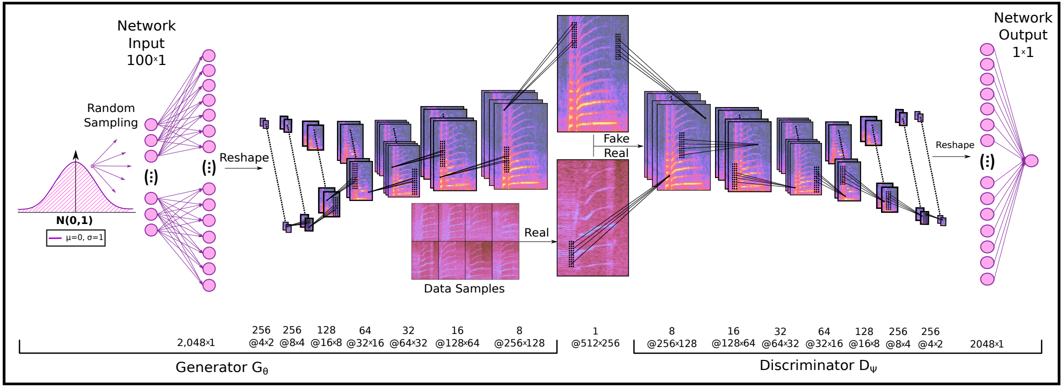
- Acoustic event classification (speaker, emotions, sentiments, vocalization paradigms, pathology, voice activity,...)

Source: Image taken from "Dissertation Christian Bergler"

OTH Amberg-Weiden

Speech Synthesis

ORCA-WHISPER



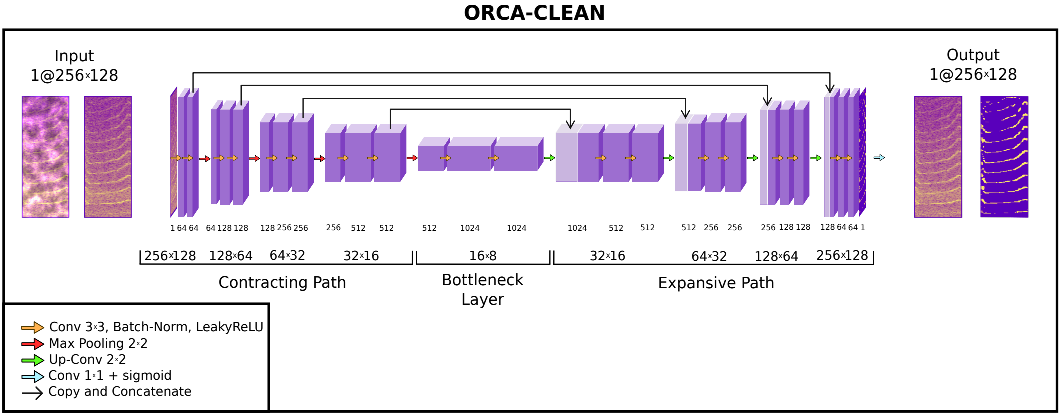
- Text-To-Speech (TTS) synthesis, voice generation (deep fakes), speech translation, ...

Source: Image taken from https://medium.com/@globalbizoutlook/ai-voice-generators-what-are-they-and-how-do-they-work-60e6c8067e4c

Acoustic Signal Processing

Audio Signals in Deep Learning – Applications

Acoustic Enhancement and Noise Reduction

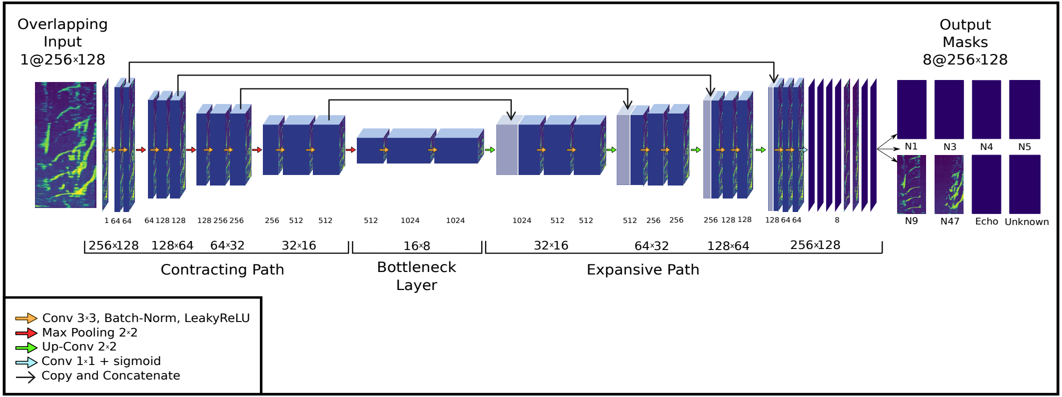


- Denoising (Noise2Noise concept, binary masking, noisy vs. clean)

Source: Image taken from "Dissertation Christian Bergler"

Sound Source Separation

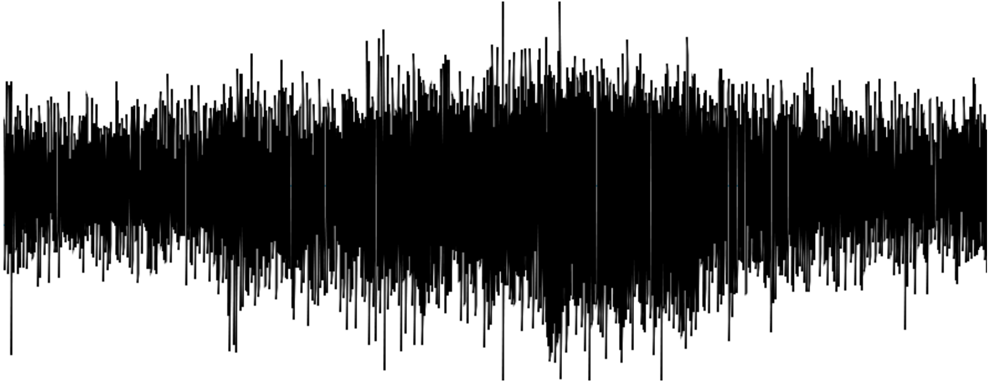
ORCA-PARTY



- Source separation, speaker separation, patter separation (“Cocktail Party Effect”)

Source: Image taken from “Dissertation Christian Bergler”

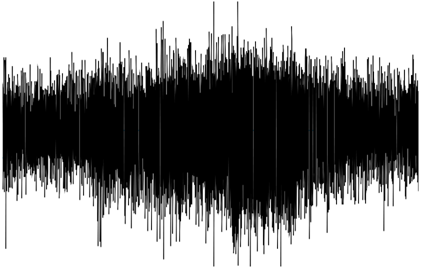
Soundwave



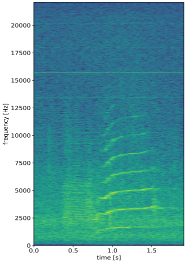
- Pure waveform (amplitude $f(t)$ over time t) is difficult to interpret!
- What are the individual components of an audio signal and how do I find patterns?

Acoustic Signal Processing
Audio Analysis

- Goal: Analysis of analog audio signals (waveform) by investigating the spectral envelope (spectrum) in order to derive the characteristic of various signals (e.g. human speech, animal sounds, etc.)



Waveform



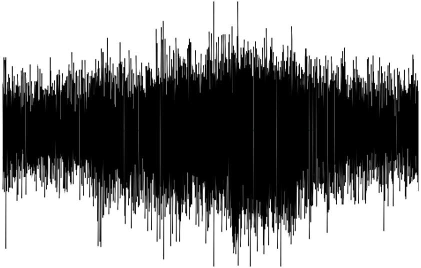
Spectrogam

Source: Image taken from FAU-Lecture Slides "Praktikum Representation Learning" (Bergler, Christlein)

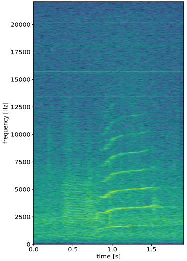
Acoustic Signal Processing

Audio Analysis

- Goal: Analysis of analog audio signals (waveform) by investigating the spectral envelope (spectrum) in order to derive the characteristic of various signals (e.g. human speech, animal sounds, etc.)



Waveform



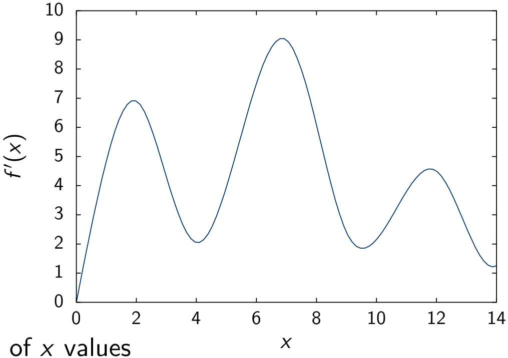
Spectrogam

- Approach: Sampling and Quantization (Digitization), Short Time Fourier Transform (STFT), Spectrogram (time and spectral visualization)

Source: Image taken from FAU-Lecture Slides "Praktikum Representation Learning" (Bergler, Christlein)

Acoustic Signal Processing

Analog/Digital (A/D) Conversion

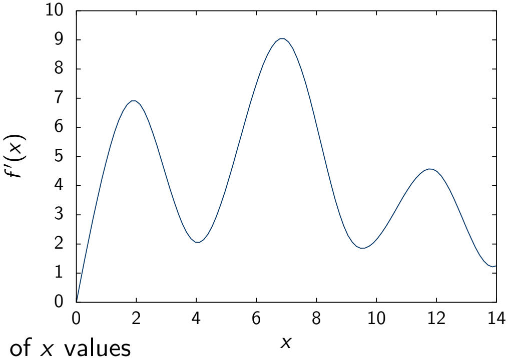


- Analog signals:
  - Continuous range of x values
  - Continuous range of amplitude/function values $f'(x)$

Source: Image taken from FAU-Lecture Slides "Praktikum Representation Learning" (Bergler, Christlein)

Acoustic Signal Processing

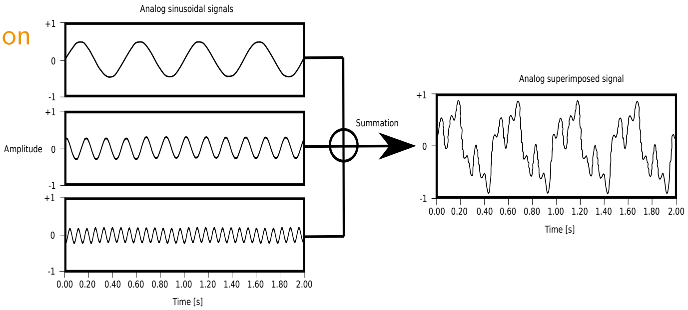
Analog/Digital (A/D) Conversion



- Analog signals:
  - Continuous range of x values
  - Continuous range of amplitude/function values $f'(x)$
- Digital signals:
  - Only a finite amount of values can be stored
  - Finite number of bits (discrete values)

Source: Image taken from FAU-Lecture Slides "Praktikum Representation Learning" (Bergler, Christlein)

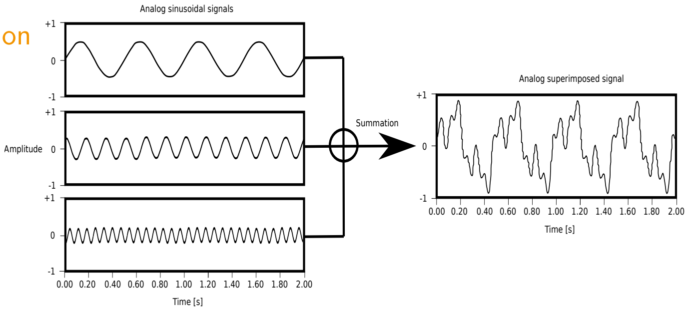
(Analog) Signal Superposition



- A given signal f is considered as periodic using a period $\lambda \in \mathbb{R}_{>0}$ if $f(t) = f(t + \alpha\lambda)$, where $\alpha \in \mathbb{Z}$ for every given $t \in \mathbb{R}$

Source: Image taken from “Dissertation Christian Bergler”

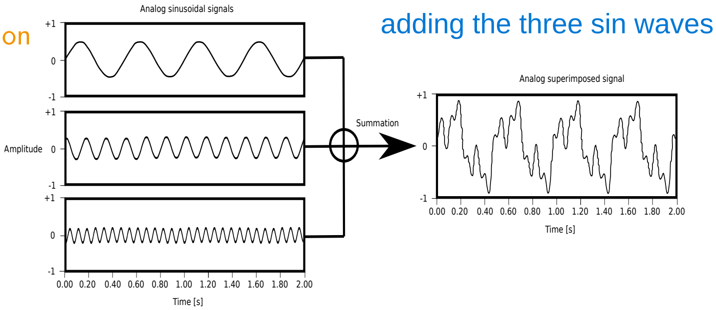
(Analog) Signal Superposition



- A given signal f is considered as periodic using a period $\lambda \in \mathbb{R}_{>0}$ if $f(t) = f(t + \alpha\lambda)$, where $\alpha \in \mathbb{Z}$ for every given $t \in \mathbb{R}$
- $f(t) := A \cdot sin(2\pi(\omega t - \phi))$, with $A =$ Amplitude (loudness), $\omega =$ frequency (pitch), $\lambda = \frac{1}{\omega} =$ period (repeating!), and $\phi =$ phase

Source: Image taken from “Dissertation Christian Bergler”

(Analog) Signal Superposition



- A given signal f is considered as periodic using a period $\lambda \in \mathbb{R}_{>0}$ if $f(t) = f(t + \alpha\lambda)$, where $\alpha \in \mathbb{Z}$ for every given $t \in \mathbb{R}$
- $f(t) := A \cdot sin(2\pi(\omega t - \phi))$, with $A =$ Amplitude (loudness), $\omega =$ frequency (pitch), $\lambda = \frac{1}{\omega} =$ period (repeating!), and $\phi =$ phase
- Superposition: $(f + g)(t) := f(t) + g(t) \rightarrow$ Still periodic signal!

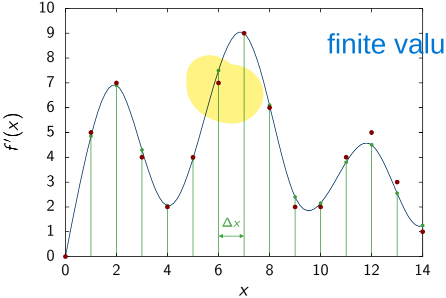
Source: Image taken from "Dissertation Christian Bergler"

Acoustic Signal Processing
Analog/Digital (A/D) Conversion

1khz means in one second we have 1000 dots to make it discreate.

A/D Conversion (Coding) involves:

1. Sampling:
   Measuring the amplitude/function values at a finite number of positions
2. Quantization:
   Representing the amplitude values by a finite number of natural numbers

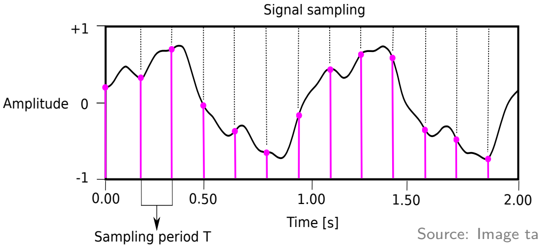


finite values of amplitude range

Source: Image taken from FAU-Lecture Slides "Praktikum Representation Learning" (Bergler, Christlein)

Analog/Digital (A/D) Conversion – Sampling

- Transforming a continuous-time signal $f : \mathbb{R} \to \mathbb{R}$ to a discrete-time signal $x : \mathbb{Z} \to \mathbb{R}$

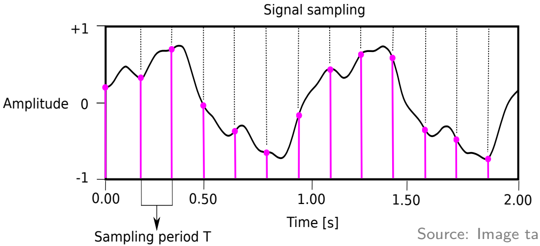


Source: Image taken from "Dissertation Christian Bergler"

Acoustic Signal Processing

Analog/Digital (A/D) Conversion – Sampling

- Transforming a continuous-time signal $f : \mathbb{R} \rightarrow \mathbb{R}$ to a discrete-time signal $x : \mathbb{Z} \rightarrow \mathbb{R}$
- Equidistant sampling: $x(n) := f(n \cdot T) \rightarrow x(n) = \text{sample}$

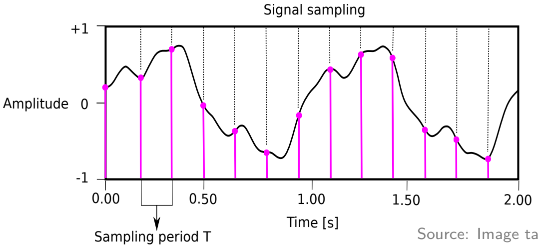


Source: Image taken from "Dissertation Christian Bergler"

Acoustic Signal Processing

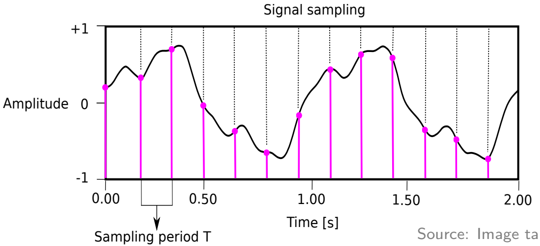
Analog/Digital (A/D) Conversion – Sampling

- Transforming a continuous-time signal $f : \mathbb{R} \rightarrow \mathbb{R}$ to a discrete-time signal $x : \mathbb{Z} \rightarrow \mathbb{R}$
- Equidistant sampling: $x(n) := f(n \cdot T) \rightarrow x(n) = \text{sample}$
- Sampling period T, with $F_s = 1/T$ as the sampling rate



Source: Image taken from "Dissertation Christian Bergler"

- Transforming a continuous-time signal $f : \mathbb{R} \rightarrow \mathbb{R}$ to a discrete-time signal $x : \mathbb{Z} \rightarrow \mathbb{R}$
- Equidistant sampling: $x(n) := f(n \cdot T) \rightarrow x(n) =$ sample
- Sampling period T, with $F_s = 1/T$ as the sampling rate
- Default: lossy conversion $\rightarrow$ discrete $x(n)$ to reconstruct continuous $f(t)$ (Aliasing!)



Source: Image taken from "Dissertation Christian Bergler"

OTH Amberg-Weiden

Analog/Digital (A/D) Conversion – Sampling

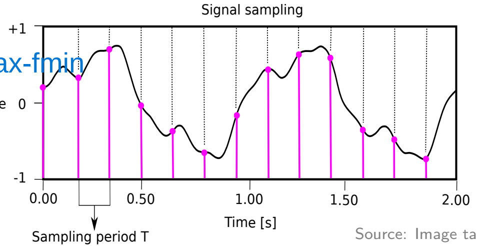
- Transforming a continuous-time signal $f : \mathbb{R} \rightarrow \mathbb{R}$ to a discrete-time signal $x : \mathbb{Z} \rightarrow \mathbb{R}$
- Equidistant sampling: $x(n) := f(n \cdot T) \rightarrow x(n) = \text{sample}$

n replaces time

- Sampling period T, with $F_s = 1/T$ as the sampling rate
- Default: lossy conversion $\rightarrow$ discrete $x(n)$ to reconstruct continuous $f(t)$ (Aliasing!)
- Solution: Nyquist-Shannon Sampling Theorem, facilitates a perfect and lossless signal reconstruction, using: $F_s > 2 \cdot (f_{max} - f_{min})$ (Nyquist-Frequency $\Omega = \frac{F_s}{2}$)

sampling rate in 2 times fmax-fmin

usually fmin is zero



humans are able to hear 22khz so we sample it with 44khz which can be devided by 2

Source: Image taken from "Dissertation Christian Bergler"

Acoustic Signal Processing

Analog/Digital (A/D) Conversion – Nyquist-Shannon Sampling Theorem

- Let $f(x)$ be a band-limited function in the frequency range $[-B_x, B_x]$.
- Then $f(x)$ is determined completely by the samples

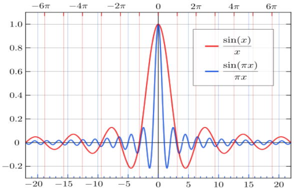
$$f_k = f(k \cdot \Delta x), \quad k = 0, \pm 1, \pm 2, \ldots$$

if the following constraint holds for the sampling interval Δx:

$$\Delta x \leq \frac{1}{2B_x} = \frac{1}{f_{sample}}, \text{with: } f_{sample} > 2 \cdot B_x$$

- The original signal $f(x)$ can be reconstructed precisely using the following interpolation:

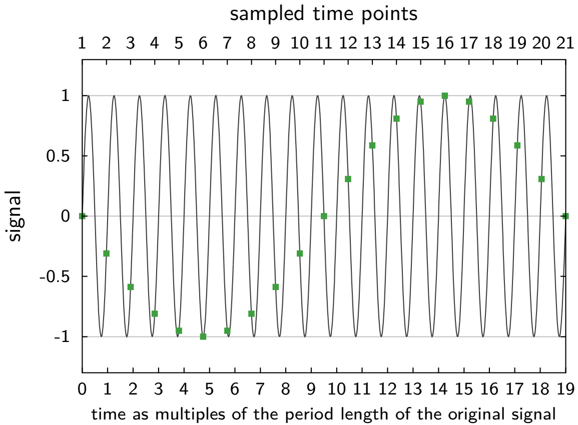
$$f(x) = \sum_{k=-\infty}^{\infty} f_k \cdot sinc\left(2\pi B_x (x - k\Delta x)\right)$$



Source: Image taken from https://en.wikipedia.org/wiki/Sinc_function, FAU-Lecture Slides “Praktikum Representation Learning” (Bergler, Christlein)

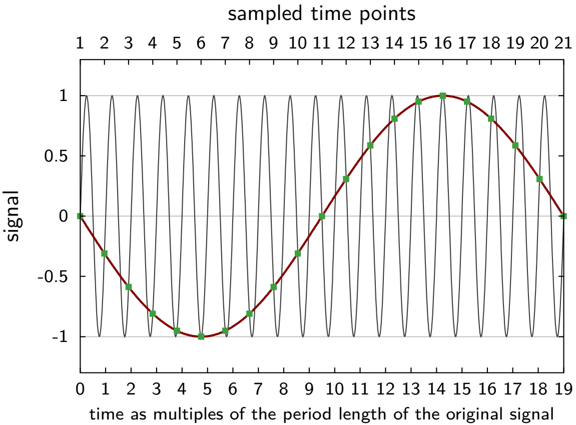
OTH Amberg-Weiden

Impact of the Sampling Theorem – Undersampling (Aliasing!)



Source: FAU-Lecture Slides "Praktikum Representation Learning" (Bergler, Christlein)

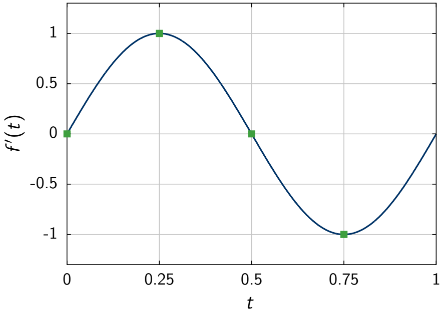
Impact of the Sampling Theorem – Undersampling (Aliasing!)



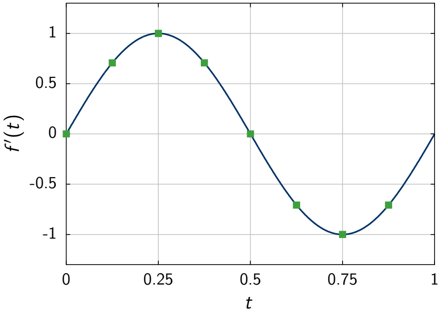
Source: FAU-Lecture Slides "Praktikum Representation Learning" (Bergler, Christlein)

Impact of the Sampling Theorem – Oversampling

- Avoids aliasing, improves resolution, reduces noise
- Higher sampling rates lead to larger amounts of data!



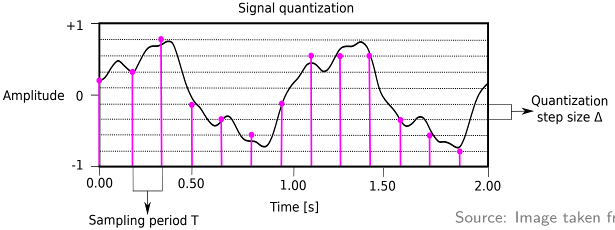
(a) $f_s = 4 \cdot f_{max}$



(b) $f_s = 8 \cdot f_{max}$

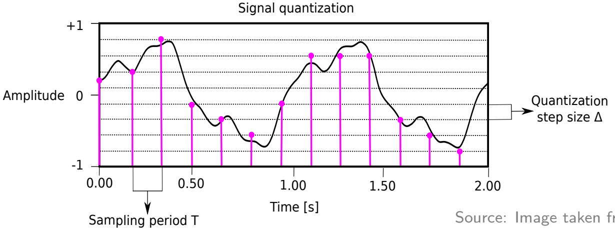
Source: FAU-Lecture Slides "Praktikum Representation Learning" (Bergler, Christlein)

- Sampling is the discretization in time ($f : \mathbb{R} \rightarrow \mathbb{R}$ to $x : \mathbb{Z} \rightarrow \mathbb{R}$)



Source: Image taken from "Dissertation Christian Bergler"

- Sampling is the discretization in time ($f : \mathbb{R} \rightarrow \mathbb{R}$ to $x : \mathbb{Z} \rightarrow \mathbb{R}$)
- Quantization is the discretization process of the continuous amplitude values $a \in \mathbb{R}$ converted via Q: $\mathbb{R} \rightarrow \Gamma$, with the discrete set $\Gamma \subset \mathbb{R}$

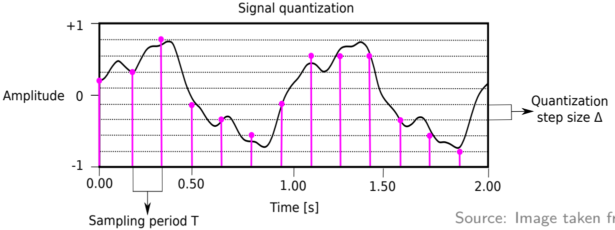


Source: Image taken from "Dissertation Christian Bergler"

Acoustic Signal Processing

Analog/Digital (A/D) Conversion – Quantization

- Sampling is the discretization in time ($f : \mathbb{R} \rightarrow \mathbb{R}$ to $x : \mathbb{Z} \rightarrow \mathbb{R}$)
- Quantization is the discretization process of the continuous amplitude values $a \in \mathbb{R}$ converted via Q: $\mathbb{R} \rightarrow \Gamma$, with the discrete set $\Gamma \subset \mathbb{R}$
- $Q(a) \in \Gamma$, with $Q(a) := sgn(a) \cdot \Delta \cdot \lfloor \frac{|a|}{\Delta} + \frac{1}{2} \rfloor$, with $a \in \mathbb{R}$, quantization step-size Δ, $sgn(\cdot)$ as the signum function, $\lfloor \cdot \rfloor$ as real number truncation

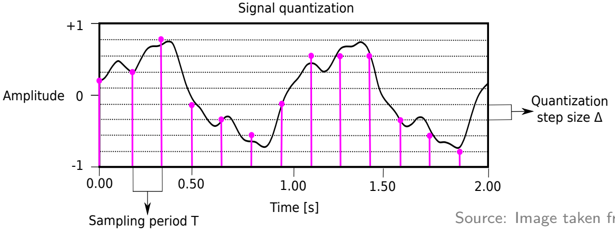


Source: Image taken from "Dissertation Christian Bergler"

Acoustic Signal Processing

Analog/Digital (A/D) Conversion – Quantization

- Sampling is the discretization in time ($f : \mathbb{R} \rightarrow \mathbb{R}$ to $x : \mathbb{Z} \rightarrow \mathbb{R}$)
- Quantization is the discretization process of the continuous amplitude values $a \in \mathbb{R}$ converted via Q: $\mathbb{R} \rightarrow \Gamma$, with the discrete set $\Gamma \subset \mathbb{R}$
- $Q(a) \in \Gamma$, with $Q(a) := sgn(a) \cdot \Delta \cdot \lfloor \frac{|a|}{\Delta} + \frac{1}{2} \rfloor$, with $a \in \mathbb{R}$, quantization step-size Δ, $sgn(\cdot)$ as the signum function, $\lfloor \cdot \rfloor$ as real number truncation
- Lossy operation: different amplitudes $a \in \mathbb{R}$ are mapped to the same discrete value $Q(a)$, known as Quantization Error!

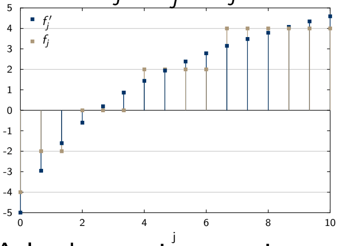


Source: Image taken from "Dissertation Christian Bergler"

Analog/Digital (A/D) Conversion – Quantization

- Quantization error n_j between real value f'_j and discretized value f_j:

$$n_j = f'_j - f_j$$

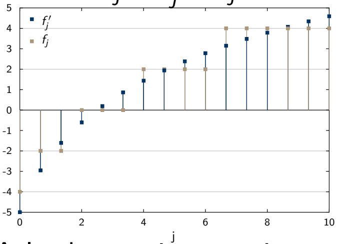


- Smaller quantization steps Δ lead to an increase in resolution, less errors, and significantly higher number of required bits for encoding

Source: Image taken from FAU-Lecture Slides "Praktikum Representation Learning" (Bergler, Christlein)

- Quantization error n_j between real value f_j' and discretized value f_j:

$$n_j = f_j' - f_j$$



- Smaller quantization steps Δ lead to an increase in resolution, less errors, and significantly higher number of required bits for encoding
- Usually impossible to reconstruct the original analog waveform, however, fulfilling the Nyquist-Shannon Theorem with adequate sampling rates, together with a sufficiently high quantization resolution, reconstructs the original perceptually free!

Source: Image taken from FAU-Lecture Slides "Praktikum Representation Learning" (Bergler, Christlein)

Acoustic Signal Processing

Analog/Digital (A/D) Conversion – Discrete Fourier Transform

- The Fourier analysis can be considered as the inverse process of decomposing an analog and periodic (stationary) audio signal $f(t)$ into its weighted components of superimposed elementary and periodic sinusoidal functions: $f(t) \rightarrow DFT \rightarrow \hat{f}(\omega)$

Source: FAU-Lecture Slides "Praktikum Representation Learning" (Bergler, Christlein) & "Dissertation Christian Bergler"

- The Fourier analysis can be considered as the inverse process of decomposing an analog and periodic (stationary) audio signal $f(t)$ into its weighted components of superimposed elementary and periodic sinusoidal functions: $f(t) \to DFT \to \hat{f}(\omega)$
- Identify all underlying and important frequency components ω of a given signal $f(t)$

Source: FAU-Lecture Slides “Praktikum Representation Learning” (Bergler, Christlein) & “Dissertation Christian Bergler”

Acoustic Signal Processing

Analog/Digital (A/D) Conversion – Discrete Fourier Transform

- The Fourier analysis can be considered as the inverse process of decomposing an analog and periodic (stationary) audio signal $f(t)$ into its weighted components of superimposed elementary and periodic sinusoidal functions: $f(t) \rightarrow DFT \rightarrow \hat{f}(\omega)$
- Identify all underlying and important frequency components ω of a given signal $f(t)$
- Continuous Fourier Transform:

$$\hat{f}(\omega) = \int_{t \in \mathbb{R}} f(t) e^{-2\pi i \omega t} dt = \int_{t \in \mathbb{R}} f(t) \cos(-2\pi\omega t) dt + i \int_{t \in \mathbb{R}} f(t) \sin(-2\pi\omega t) dt$$

- However: Digitized signals need a discrete version $\rightarrow$ Discrete Fourier Transform (DFT)!

Source: FAU-Lecture Slides “Praktikum Representation Learning” (Bergler, Christlein) & “Dissertation Christian Bergler”

- The Fourier analysis can be considered as the inverse process of decomposing an analog and periodic (stationary) audio signal $f(t)$ into its weighted components of superimposed elementary and periodic sinusoidal functions: $f(t) \rightarrow DFT \rightarrow \hat{f}(\omega)$
- Identify all underlying and important frequency components ω of a given signal $f(t)$
- Continuous Fourier Transform:

$$\hat{f}(\omega) = \int_{t \in \mathbb{R}} f(t) e^{-2\pi i \omega t} dt = \int_{t \in \mathbb{R}} f(t) \cos(-2\pi \omega t) dt + i \int_{t \in \mathbb{R}} f(t) \sin(-2\pi \omega t) dt$$

- However: Digitized signals need a discrete version → Discrete Fourier Transform (DFT)!

$$X[k] \stackrel{\text{def}}{=} \sum_{n=0}^{N-1} x[n] \cdot \mathrm{e}^{-\frac{2\pi i n k}{N}} = \sum_{n=0}^{N-1} x[n] \cdot \mathrm{W}_N^{nk}, \text{ with } W_N = \mathrm{e}^{-\frac{2\pi i}{N}}$$

Source: FAU-Lecture Slides “Praktikum Representation Learning” (Bergler, Christlein) & “Dissertation Christian Bergler”

Acoustic Signal Processing

Analog/Digital (A/D) Conversion – Discrete Fourier Transform

- The Fourier analysis can be considered as the inverse process of decomposing an analog and periodic (stationary) audio signal $f(t)$ into its weighted components of superimposed elementary and periodic sinusoidal functions: $f(t) \rightarrow DFT \rightarrow \hat{f}(\omega)$
- Identify all underlying and important frequency components ω of a given signal $f(t)$
- Continuous Fourier Transform:

$$\hat{f}(\omega) = \int_{t \in \mathbb{R}} f(t) e^{-2\pi i \omega t} dt = \int_{t \in \mathbb{R}} f(t) \cos(-2\pi\omega t) dt + i \int_{t \in \mathbb{R}} f(t) \sin(-2\pi\omega t) dt$$

- However: Digitized signals need a discrete version $\rightarrow$ Discrete Fourier Transform (DFT)!

$$X[k] \stackrel{\text{def}}{=} \sum_{n=0}^{N-1} x[n] \cdot \mathrm{e}^{-\frac{2\pi i n k}{N}} = \sum_{n=0}^{N-1} x[n] \cdot \mathrm{W}_N^{nk}, \text{ with } W_N = \mathrm{e}^{-\frac{2\pi i}{N}}$$

with $X[k] \in \mathbb{C}$ (Re/Im), n = sample index, $K \in \mathbb{N}$ = fixed number of frequency bins, $k \in \mathbb{N}$ = frequency index, typical $K = N$, frequency resolution $\omega = \frac{k}{N=K}$

Source: FAU-Lecture Slides "Praktikum Representation Learning" (Bergler, Christlein) & "Dissertation Christian Bergler"

DFT – Matrix-Vector Product

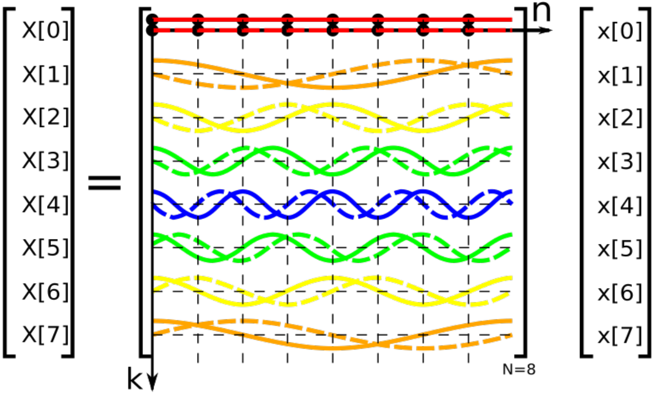
$$X[k] \overset{\text{def}}{=} \sum_{n=0}^{N-1} x[n] \cdot \mathrm{e}^{-\frac{2\pi ink}{N}} = \sum_{n=0}^{N-1} x[n] \cdot \mathrm{W}_N^{nk}, \text{ with } W_N = \mathrm{e}^{-\frac{2\pi i}{N}}$$

$$\begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ X[3] \\ \vdots \\ X[N-1] \end{bmatrix}_{N\times 1} = \begin{bmatrix} 1 & 1 & 1 & 1 & \dots & 1 \\ 1 & W_N^1 & W_N^2 & W_N^3 & \dots & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & W_N^6 & \dots & W_N^{2(N-1)} \\ 1 & W_N^3 & W_N^6 & W_N^9 & \dots & W_N^{3(N-1)} \\ \vdots & & & & & \\ 1 & W_N^{(N-1)} & W_N^{2(N-1)} & W_N^{3(N-1)} & \dots & W_N^{(N-1)(N-1)} \end{bmatrix}_{N\times N} \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \\ \vdots \\ x[N-1] \end{bmatrix}_{N\times 1}$$

Source: FAU-Lecture Slides “Praktikum Representation Learning” (Bergler, Christlein) & “Dissertation Christian Bergler”

Acoustic Signal Processing

Analog/Digital (A/D) Conversion – Discrete Fourier Transform

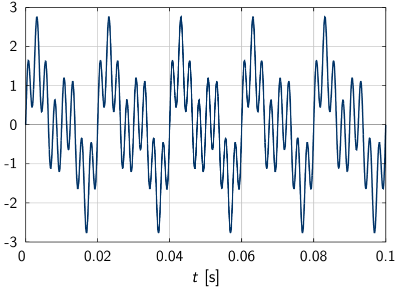


- Each row refers to a particular frequency component $W_N^{nk} = e^{-\frac{2\pi ink}{N}} = e^{-2\pi i\omega n}$
- Diagonal symmetric due to complex conjugate elements (negative frequencies), which are discarded, so only the first $\lfloor N/2+1 \rfloor$ frequency bins are required

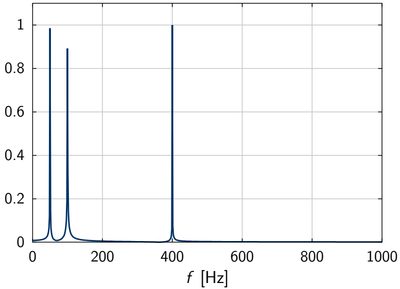
Source: Image taken from https://en.wikipedia.org/wiki/DFT_matrix

DFT – Periodic Signal

- Example: Summation of three different sinusodial function frequencies
- $f_1 = 50\,\text{Hz},\ f_2 = 100\,\text{Hz},\ f_3 = 400\,\text{Hz}$
- $f(t) = \sin(2\pi \cdot 50 \cdot t) + \sin(2\pi \cdot 100 \cdot t) + \sin(2\pi \cdot 400 \cdot t)$



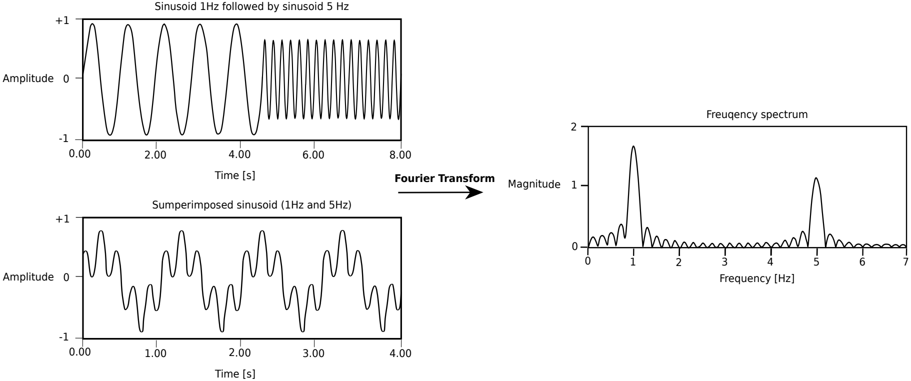
(a) waveform



(b) Fourier spectrum (amplitudes)

Source: FAU-Lecture Slides "Praktikum Representation Learning" (Bergler, Christlein)

DFT – Non-Periodic Signal



- Same DFT-output for both input signals (→ DFT looses information about the temporal occurrence of a certain frequency) → How to handle real-world signals???

Source: FAU-Lecture Slides “Praktikum Representation Learning” (Bergler, Christlein) & “Dissertation Christian Bergler”

Analog/Digital (A/D) Conversion – DFT – Periodic VS. Non-Periodic Signal

- DFT is only meaningful/reasonable for periodic (stationary) signals

Source: "Dissertation Christian Bergler"

Analog/Digital (A/D) Conversion – DFT – Periodic VS. Non-Periodic Signal

- DFT is only meaningful/reasonable for periodic (stationary) signals
- However, real-world signals are mostly time varying, characterized by a changing and non-stationary/-periodic frequency information $\rightarrow$ No periodicity over longer periods (e.g. speech, music, ...)

Source: "Dissertation Christian Bergler"

Analog/Digital (A/D) Conversion – DFT – Periodic VS. Non-Periodic Signal

- DFT is only meaningful/reasonable for periodic (stationary) signals
- However, real-world signals are mostly time varying, characterized by a changing and non-stationary/-periodic frequency information $\rightarrow$ No periodicity over longer periods (e.g. speech, music, ...)
- Fourier transform across the whole temporal context extracts the entire frequency information, however, without maintaining the knowledge at which specific point in time a certain frequency exactly occurs

Source: “Dissertation Christian Bergler”

Analog/Digital (A/D) Conversion – DFT – Periodic VS. Non-Periodic Signal

- DFT is only meaningful/reasonable for periodic (stationary) signals
- However, real-world signals are mostly time varying, characterized by a changing and non-stationary/-periodic frequency information $\rightarrow$ No periodicity over longer periods (e.g. speech, music, ...)
- Fourier transform across the whole temporal context extracts the entire frequency information, however, without maintaining the knowledge at which specific point in time a certain frequency exactly occurs

Solution

- Consider very short time segments (windows) and step-by-step compute DFT

Source: "Dissertation Christian Bergler"

- DFT is only meaningful/reasonable for periodic (stationary) signals
- However, real-world signals are mostly time varying, characterized by a changing and non-stationary/-periodic frequency information → No periodicity over longer periods (e.g. speech, music, ...)
- Fourier transform across the whole temporal context extracts the entire frequency information, however, without maintaining the knowledge at which specific point in time a certain frequency exactly occurs

Solution

- Consider very short time segments (windows) and step-by-step compute DFT
- Short-time analysis maintains both: time information when frequencies appear and periodicity, since short temporal segments of real signals are quasi-stationary (implicite assumption: periodic continuation of the window)

Source: "Dissertation Christian Bergler"

Acoustic Signal Processing

Analog/Digital (A/D) Conversion – DFT – Periodic VS. Non-Periodic Signal

- DFT is only meaningful/reasonable for periodic (stationary) signals
- However, real-world signals are mostly time varying, characterized by a changing and non-stationary/-periodic frequency information $\rightarrow$ No periodicity over longer periods (e.g. speech, music, ...)
- Fourier transform across the whole temporal context extracts the entire frequency information, however, without maintaining the knowledge at which specific point in time a certain frequency exactly occurs

Solution

- Consider very short time segments (windows) and step-by-step compute DFT
- Short-time analysis maintains both: time information when frequencies appear and periodicity, since short temporal segments of real signals are quasi-stationary (implicite assumption: periodic continuation of the window)
- Algorithmen: Short-Time Fourier Transform (STFT)

Source: “Dissertation Christian Bergler”

The STFT Algorithm

1. Extraction of a short-time audio excerpt from the original input signal according to the chosen FFT window-size N

The STFT Algorithm

1. Extraction of a short-time audio excerpt from the original input signal according to the chosen FFT window-size N
2. Selection of a proper window function w, very common: Hamming window ($\epsilon = 0.54$) and/or Hanning ($\epsilon = 0.5$) window ($+$ many others)

$$w(n) := \begin{cases} \epsilon - (1-\epsilon) \cdot \cos(\frac{2\pi n}{N}) & \text{with } 0 \leq n \leq N-1 \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

OTH Amberg-Weiden

The STFT Algorithm

1. Extraction of a short-time audio excerpt from the original input signal according to the chosen FFT window-size N
2. Selection of a proper window function w, very common: Hamming window ($\epsilon = 0.54$) and/or Hanning ($\epsilon = 0.5$) window (+ many others)

$$w(n) := \begin{cases} \epsilon - (1-\epsilon) \cdot \cos(\frac{2\pi n}{N}) & \text{with } 0 \leq n \leq N-1 \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

3. Apply the window function w to the actual signal x to obtain the final windowed signal

The STFT Algorithm

1. Extraction of a short-time audio excerpt from the original input signal according to the chosen FFT window-size N
2. Selection of a proper window function w, very common: Hamming window ($\epsilon = 0.54$) and/or Hanning ($\epsilon = 0.5$) window ($+$ many others)

$$w(n) := \begin{cases} \epsilon - (1-\epsilon) \cdot \cos(\frac{2\pi n}{N}) & \text{with } 0 \leq n \leq N-1 \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

3. Apply the window function w to the actual signal x to obtain the final windowed signal
4. Computation of the DFT from the windowed signal, representing the entire frequency information (spectrum slide) for a given time bin of the overall final spectrogram

$$\mathcal{X}(m,k) := \sum_{n=0}^{N-1} x(n+mH)w(n)e^{\frac{-2\pi ikn}{N}} \tag{2}$$

The STFT Algorithm

1. Extraction of a short-time audio excerpt from the original input signal according to the chosen FFT window-size N
2. Selection of a proper window function w, very common: Hamming window ($\epsilon = 0.54$) and/or Hanning ($\epsilon = 0.5$) window (+ many others)

$$w(n) := \begin{cases} \epsilon - (1-\epsilon) \cdot \cos(\frac{2\pi n}{N}) & \text{with } 0 \leq n \leq N-1 \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

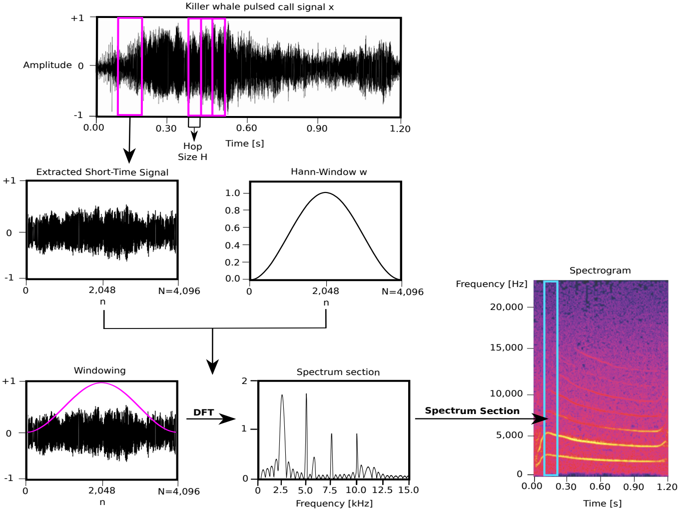
3. Apply the window function w to the actual signal x to obtain the final windowed signal
4. Computation of the DFT from the windowed signal, representing the entire frequency information (spectrum slide) for a given time bin of the overall final spectrogram

$$\mathcal{X}(m,k) := \sum_{n=0}^{N-1} x(n+mH)w(n)e^{\frac{-2\pi ikn}{N}} \tag{2}$$

5. Move the analysis window by the chosen hop-size H and repeat the entire procedure

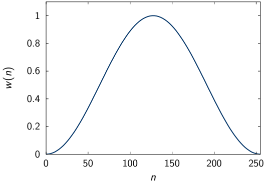
Acoustic Signal Processing

Analog/Digital (A/D) Conversion – Short-Time Fourier Transform (STFT)

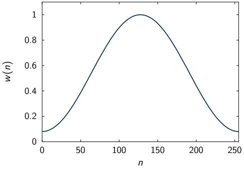


Source: "Dissertation Christian Bergler"

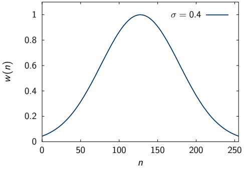
Window Function w



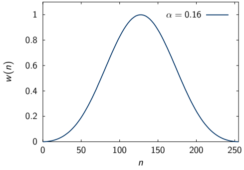
(a) Hann



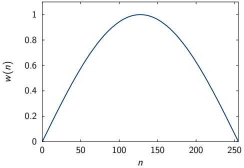
(b) Hamming



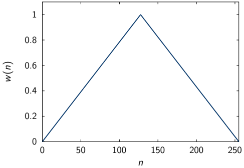
(c) Gauss



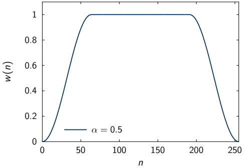
(d) Blackman



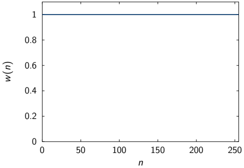
(e) Cosine



(f) Bartlett



(g) Tukey

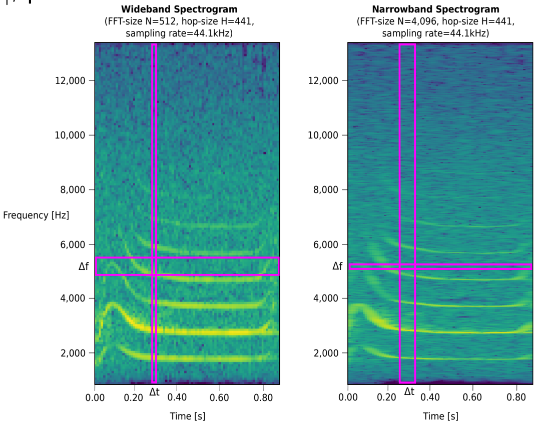


(h) Rectangle

Source: FAU-Lecture Slides “Praktikum Representation Learning” (Bergler, Christlein)

Analog/Digital (A/D) Conversion – Spectrogram (Narrow vs. Wide)

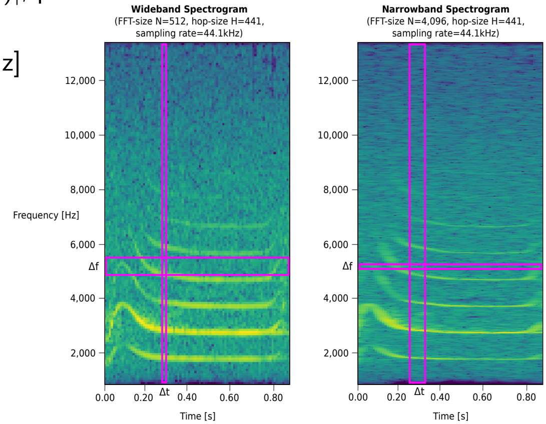
- Spectrogram: complex $X(k,m)$, magnitude $|X(k,m)|$, power $|X(k,m)|^2$, log-scaled, with specific filter-banks



Source: FAU-Lecture Slides "Praktikum Representation Learning" (Bergler, Christlein) & "Dissertation Christian Bergler"

Analog/Digital (A/D) Conversion – Spectrogram (Narrow vs. Wide)

- Spectrogram: complex $X(k, m)$, magnitude $|X(k, m)|$, power $|X(k, m)|^2$, log-scaled, with specific filter-banks
- Max-Frequency (Nyquist-Shannon): $f_{max} = f_{sr}/2$ [Hz]

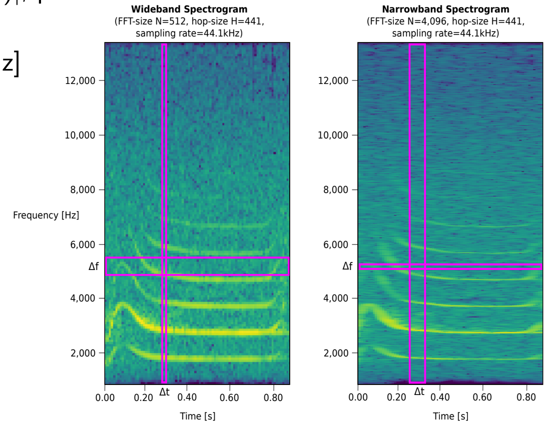


Source: FAU-Lecture Slides “Praktikum Representation Learning” (Bergler, Christlein) & “Dissertation Christian Bergler”

Acoustic Signal Processing

Analog/Digital (A/D) Conversion – Spectrogram (Narrow vs. Wide)

- Spectrogram: complex $X(k,m)$, magnitude $|X(k,m)|$, power $|X(k,m)|^2$, log-scaled, with specific filter-banks
- Max-Frequency (Nyquist-Shannon): $f_{max} = f_{sr}/2$ [Hz]
- Frequency resolution (f_{bins} = fft-size):
  $f_{resolution} = f_{sr}/\text{fft-size}$ [Hz]
  $f_{resolution} = f_{max}/(\text{fft-size}/2)$ [Hz]

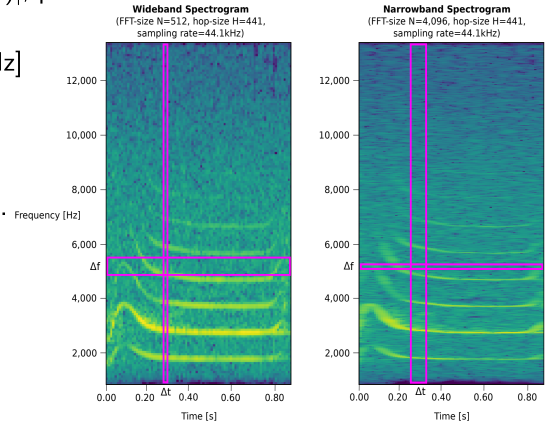


Source: FAU-Lecture Slides “Praktikum Representation Learning” (Bergler, Christlein) & “Dissertation Christian Bergler”

Acoustic Signal Processing

Analog/Digital (A/D) Conversion – Spectrogram (Narrow vs. Wide)

- Spectrogram: complex $X(k,m)$, magnitude $|X(k,m)|$, power $|X(k,m)|^2$, log-scaled, with specific filter-banks
- Max-Frequency (Nyquist-Shannon): $f_{max} = f_{sr}/2$ [Hz]
- Frequency resolution (f_{bins} = fft-size):
  $f_{resolution} = f_{sr}/\text{fft-size}$ [Hz]
  $f_{resolution} = f_{max}/(\text{fft-size}/2)$ [Hz]
- Uncertainty Principle: time resolution ($\Delta t = \frac{N}{f_{sr}}$) vs. frequency resolution ($\Delta f = \frac{f_{sr}}{N(=f_{bins})} = \frac{f_{max}}{N/2}$)

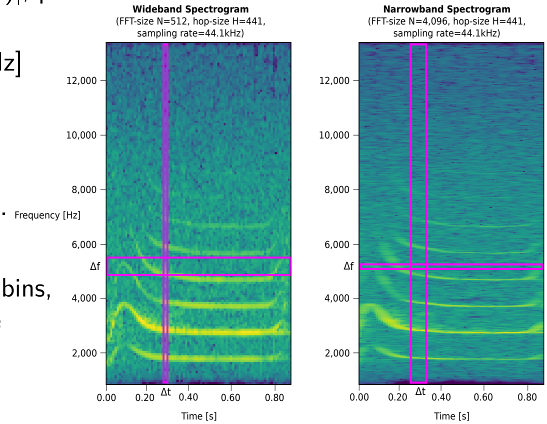


Source: FAU-Lecture Slides "Praktikum Representation Learning" (Bergler, Christlein) & "Dissertation Christian Bergler"

Acoustic Signal Processing

Analog/Digital (A/D) Conversion – Spectrogram (Narrow vs. Wide)

- Spectrogram: complex $X(k, m)$, magnitude $|X(k, m)|$, power $|X(k, m)|^2$, log-scaled, with specific filter-banks
- Max-Frequency (Nyquist-Shannon): $f_{max} = f_{sr}/2$ [Hz]
- Frequency resolution (f_{bins} = fft-size):
  $f_{resolution} = f_{sr}/\text{fft-size}$ [Hz]
  $f_{resolution} = f_{max}/(\text{fft-size}/2)$ [Hz]
- Uncertainty Principle: time resolution ($\Delta t = \frac{N}{f_{sr}}$) vs. frequency resolution ($\Delta f = \frac{f_{sr}}{N(=f_{bins})} = \frac{f_{max}}{N/2}$)
- Wideband: shorter time windows N, less frequency bins, wider frequency ranges in a single bin, less accurate frequency but more precise time information

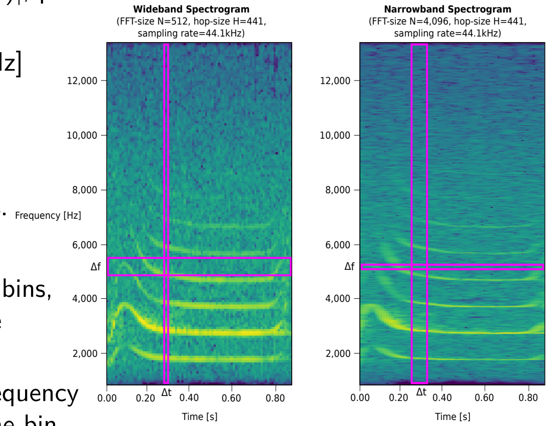


Source: FAU-Lecture Slides “Praktikum Representation Learning” (Bergler, Christlein) & “Dissertation Christian Bergler”

Acoustic Signal Processing

Analog/Digital (A/D) Conversion – Spectrogram (Narrow vs. Wide)

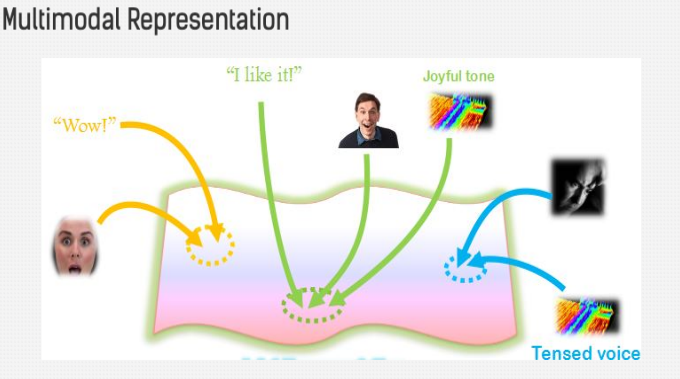
- Spectrogram: complex $X(k,m)$, magnitude $|X(k,m)|$, power $|X(k,m)|^2$, log-scaled, with specific filter-banks
- Max-Frequency (Nyquist-Shannon): $f_{max} = f_{sr}/2$ [Hz]
- Frequency resolution ($f_{bins} =$ fft-size):
  $f_{resolution} = f_{sr}/\text{fft-size}$ [Hz]
  $f_{resolution} = f_{max}/(\text{fft-size}/2)$ [Hz]
- Uncertainty Principle: time resolution ($\Delta t = \frac{N}{f_{sr}}$) vs. frequency resolution ($\Delta f = \frac{f_{sr}}{N(=f_{bins})} = \frac{f_{max}}{N/2}$)
- Wideband: shorter time windows N, less frequency bins, wider frequency ranges in a single bin, less accurate frequency but more precise time information
- Narrowband: larger time windows N, increase in frequency bins, narrows the spectral content represented in one bin, improves frequency resolution deteriorates time resolution



Source: FAU-Lecture Slides “Praktikum Representation Learning” (Bergler, Christlein) & “Dissertation Christian Bergler”

Multimodal Learning

What is it about?

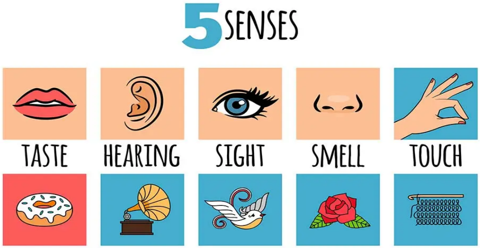


- Multimodal Learning is a learning paradigm in the field of deep learning, using various data modalities, such as text, audio, sensor signals, and images/videos within a single learning concept at the same time!

Source: Image taken from https://vinija.ai/multimodal/challenges/

Multimodal Learning

Why?

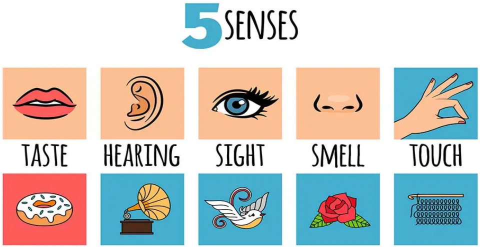


- The core idea of deep learning was to design algorithms trying to mimic the human brain (neurons, layers, deep structures, ...)

Source: Image taken from https://www.kdnuggets.com/2023/03/multimodal-models-explained.html

Multimodal Learning

Why?

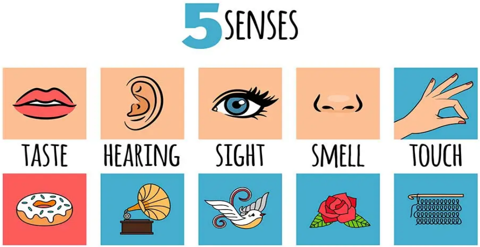


- The core idea of deep learning was to design algorithms trying to mimic the human brain (neurons, layers, deep structures, ...)
- Humans have access to five senses – sight, hearing, touch, taste, and smell – not just to collect information, but also to understand and interpret the environment around us

Source: Image taken from https://www.kdnuggets.com/2023/03/multimodal-models-explained.html

Multimodal Learning

Why?



- The core idea of deep learning was to design algorithms trying to mimic the human brain (neurons, layers, deep structures, ...)
- Humans have access to five senses – sight, hearing, touch, taste, and smell – not just to collect information, but also to understand and interpret the environment around us
- Making use of diverse information sources at the same time it is possible to derive a better and more complete understanding of the underlying purpose/data/task, unlock new and deeper insights

Source: Image taken from https://www.kdnuggets.com/2023/03/multimodal-models-explained.html

Multimodal Learning

Transfer to Deep Learning!

- The integration of multiple modalities allows a model to leverage complementary information, handle missing data from one source by relying on another, provide more comprehensive insights, and improves model generalization!

Source: Image taken from https://618media.com/en/blog/the-multimodal-capabilities-of-chatgpt-4/

Multimodal Learning

Transfer to Deep Learning!

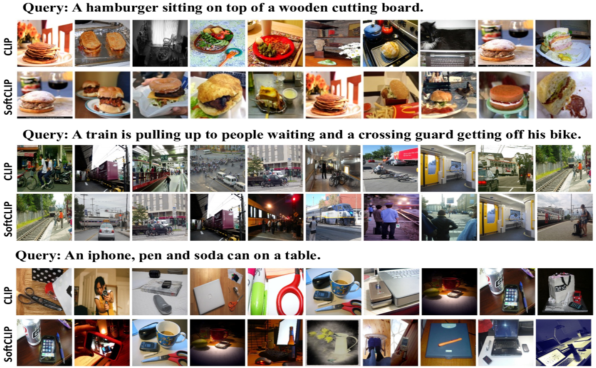
- The integration of multiple modalities allows a model to leverage complementary information, handle missing data from one source by relying on another, provide more comprehensive insights, and improves model generalization!
- Real-world data and applications are very often multimodal!

Source: Image taken from https://618media.com/en/blog/the-multimodal-capabilities-of-chatgpt-4/

Multimodal Learning

Real-World Applications – Computer Vision/NLP

Text-Image VS. Image-Text Retrieval



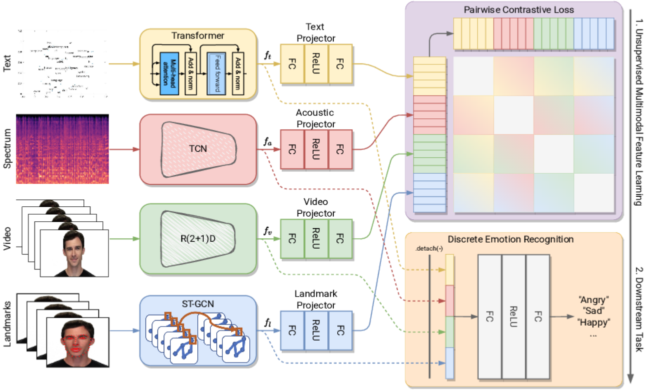
- For a given image, find related text, or vice versa, which is particularly useful in any kind of search engines

Source: Ìmage from Gao et al. "SoftCLIP: Softer Cross-modal Alignment Makes CLIP Stronger"

Multimodal Learning

Real-World Applications – Computer Vision/Audio/NLP

Emotion Recognition



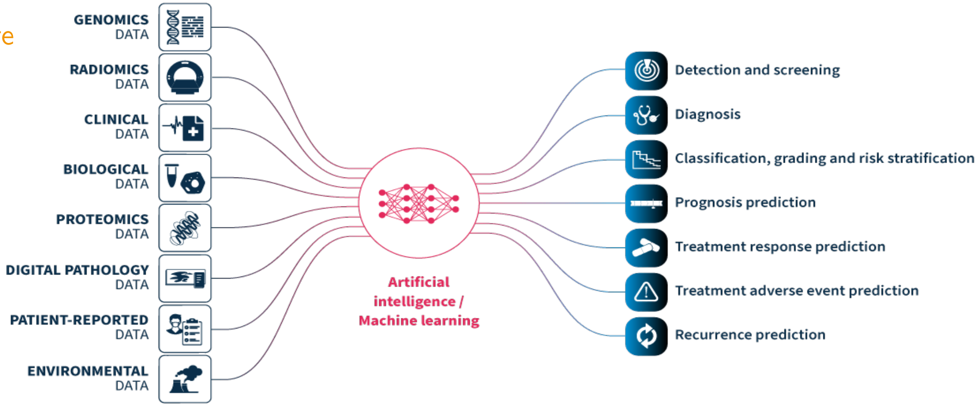
- Modern systems use video (facial expressions), audio (voice tone), and textual data (spoken words) to specifiy emotions of a person more accurately
- Further: LipSync and textual models in ASR Systems (Vision + Speech + Text)

Source: Image from Franceschini et al., "Multimodal Emotion Recognition with Modality-Pairwise Unsupervised Contrastive Loss"

Multimodal Learning

Real-World Applications – Computer Vision/Text/Audio

Healthcare



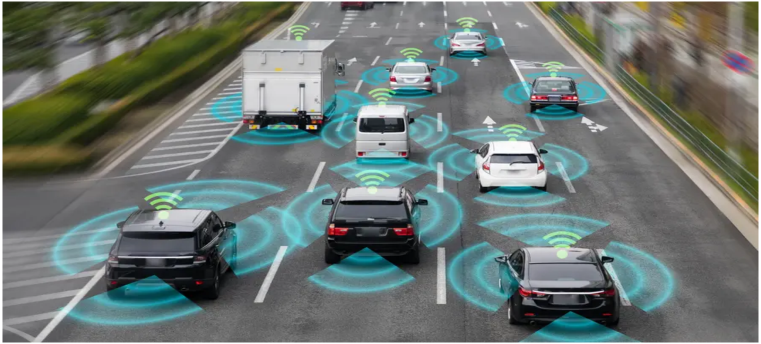
- Using multimodal data such as clinical & patient data, radiomics, biological data, environmental data, etc., in order to gain much deeper insights to the human biology & medical conditions

Source: Image from https://www.sophiagenetics.com/science-hub/the-power-of-multimodal-data-driven-medicine/

Multimodal Learning

Real-World Applications – Computer Vision/Text/Audio

Autonomous Driving



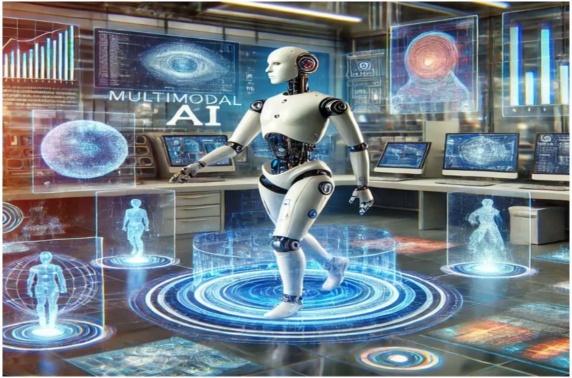
- Multiple sensor data (camera, radar, LIDAR, GPS, ...), together with audio cues (alerts, navigation, status updates, ...), to better understand the environment & to handle the respective traffic situations accordingly

Source: Image from https://www.topgear.com/car%20news/what-are-sae-levels-autonomous-driving-uk

Multimodal Learning

Real-World Applications – Computer Vision/Text/Audio

Human-Robot Interaction (Humanoid Systems)



- Intelligent Humanoid Systems: robots can interpret speech, facial expressions, gestures, and body posture to understand and interact with a human being in a very natural way

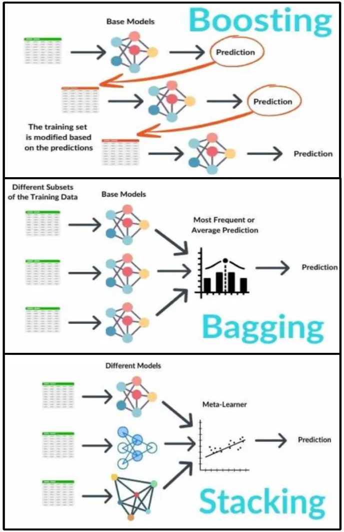
Source: Image from https://squoraishee.medium.com/multimodal-machine-learning-a-deep-dive-bc4f25f6a63b

Multimodal Learning

Combining Models VS. Multimodal Learning

Combining Models

- Combination of several individual (basic) models to improve performance & robustness, following the principle "together we are stronger"
- Boosting – Sequence of models, while each subsequent model corrects the mistakes made by the previous ones (Adaptive Boosting, Gradient Boosting Machines, ...)



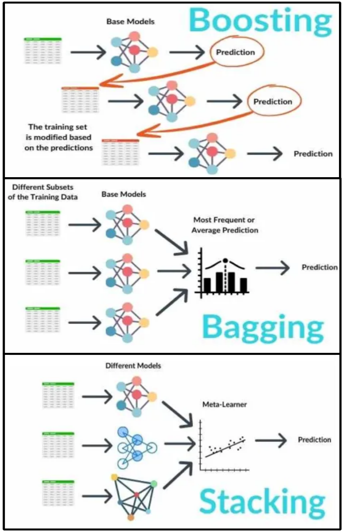
Source: https://spotintelligence.com/2024/03/18/bagging-boosting-stacking/

Multimodal Learning

Combining Models VS. Multimodal Learning

Combining Models

- Combination of several individual (basic) models to improve performance & robustness, following the principle "together we are stronger"
- Boosting – Sequence of models, while each subsequent model corrects the mistakes made by the previous ones (Adaptive Boosting, Gradient Boosting Machines, ...)
- Bagging – Training multiple model instances of the same algorithm w.r.t. different training subsets (final prediction, obtained by averaging or majority vote)



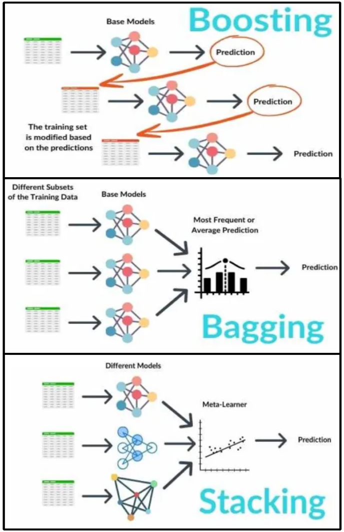
Source: https://spotintelligence.com/2024/03/18/bagging-boosting-stacking/

Multimodal Learning

Combining Models VS. Multimodal Learning

Combining Models

- Combination of several individual (basic) models to improve performance & robustness, following the principle "together we are stronger"
- Boosting – Sequence of models, while each subsequent model corrects the mistakes made by the previous ones (Adaptive Boosting, Gradient Boosting Machines, ...)
- Bagging – Training multiple model instances of the same algorithm w.r.t. different training subsets (final prediction, obtained by averaging or majority vote)
- Stacking – Training multiple diverse base models and combine the predictions using another meta-model, weighting the base-model output for final prediction



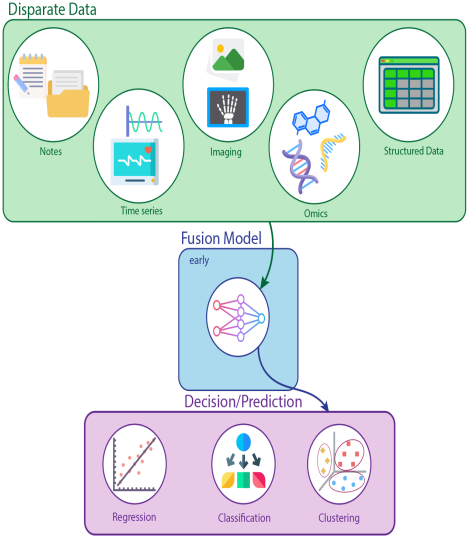
Source: https://spotintelligence.com/2024/03/18/bagging-boosting-stacking/

Multimodal Learning

Combining Models VS. Multimodal Learning

Multimodal Learning

- Combining models: trained independently, while the final prediction is made by combining the outputs, especially helpful when individual models have complementary strengths and weaknesses



Source: https://pub.towardsai.net/multimodal-machine-learning-data-fusion-d1d8776e2cb0

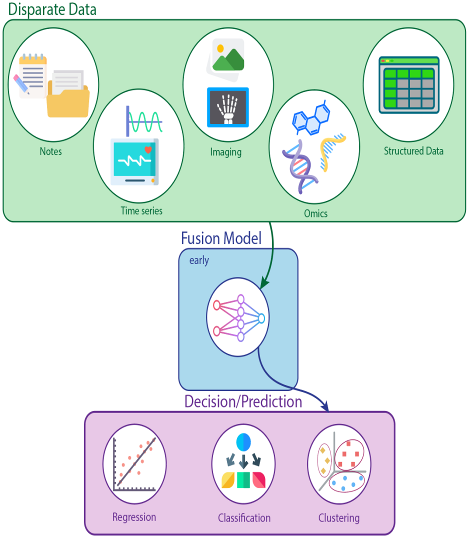
Multimodal Learning

Combining Models VS. Multimodal Learning

OTH Amberg-Weiden

Multimodal Learning

- Combining models: trained independently, while the final prediction is made by combining the outputs, especially helpful when individual models have complementary strengths and weaknesses
- Multimodal: goal is to combine & merge information from various data modalities to improve the performance on a given task
  - Early Fusion (Feature-Level Fusion)
  - Late Fusion (Decision-Level Fusion)
  - Hybrid Fusion (Mixture between Early & Late)
  - Cross-Modal Learning
  - Joint Representation Learning



Source: https://pub.towardsai.net/multimodal-machine-learning-data-fusion-d1d8776e2cb0

Multimodal Learning

Fusion and Learning Principles

General: Fusion-based approaches perform an encoding of different modalities into a common representation space, where individual feature embeddings are fused to create a single modality-invariant feature representation, capturing all the semantic knowledge

Multimodal Learning

Fusion and Learning Principles

General: Fusion-based approaches perform an encoding of different modalities into a common representation space, where individual feature embeddings are fused to create a single modality-invariant feature representation, capturing all the semantic knowledge

Early Fusion

- Early Fusion (decision-level fusion): Separately extracted features from different modalities are combined (concatenation/merging) at the input level
- Pro: Model learns similarities, correlations, interactions between the multimodal features
- Con: High-dimensional feature data $\rightarrow$ Computational complexity

Multimodal Learning

Fusion and Learning Principles

General: Fusion-based approaches perform an encoding of different modalities into a common representation space, where individual feature embeddings are fused to create a single modality-invariant feature representation, capturing all the semantic knowledge

Early Fusion

- Early Fusion (decision-level fusion): Separately extracted features from different modalities are combined (concatenation/merging) at the input level
- Pro: Model learns similarities, correlations, interactions between the multimodal features
- Con: High-dimensional feature data $\rightarrow$ Computational complexity

Late Fusion

- Each data modality is processed separately using stand-alone models, while the final predictions are combined at decision level (averaging, voting, weighted combination, ...)
- Pro: data modality-specific models, easier to handle different data characteristics
- Con: do not learn how data modality-specific features interact

OTH Amberg-Weiden

Hybrid Fusion

- Combines concepts of both – early and late fusion approaches – by merging data at intermediate stages
- Separate feature learning/extraction and downstream fusion at an intermediate layer
- aiming to combine the strengths of both feature- and decision-level fusion (modality-specific features plus cross-modality feature interactions)
- Pro: balance between computational efficiency and ability to learn feature connections
- Con: hybrid fusion strategy requires a careful selection of the fusion point

Multimodal Learning

Fusion and Learning Principles

Hybrid Fusion

- Combines concepts of both – early and late fusion approaches – by merging data at intermediate stages
- Separate feature learning/extraction and downstream fusion at an intermediate layer
- aiming to combine the strengths of both feature- and decision-level fusion (modality-specific features plus cross-modality feature interactions)
- Pro: balance between computational efficiency and ability to learn feature connections
- Con: hybrid fusion strategy requires a careful selection of the fusion point

Cross-Modal Learning

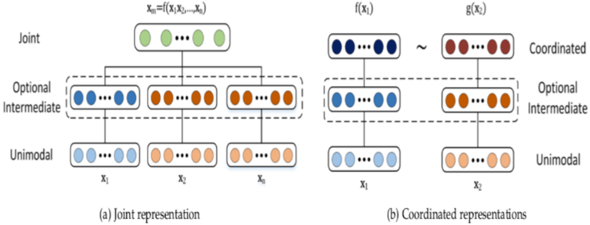
- Learning and transferring information from one data modality to another (e.g. image captioning, text-to-image synthesis, ...)
- Goal: identifying and understanding the connection and correlation between various data modalities ($\rightarrow$ Know-How Transfer!)

Multimodal Learning

Fusion and Learning Principles

Joint Representation Learning

- Unified feature space for all the different modalities
- Fuse information, while capturing & understanding complementary modality information
- Better generalization due to significantly larger knowledge access of multiple modalities



- Coordinated representations projects all the modalities (usually severe differences in characteristics) to its own space, while being coordinated over a constraint

Multimodal Learning

Challenges

- Data Imbalance: problem if a single modality has significantly more data than another data modality

Multimodal Learning

Challenges

- Data Imbalance: problem if a single modality has significantly more data than another data modality
- Data Noise and Quality: impact of low-quality data in one or more modalities on the learning process and how to sum up the important information from multiple data modalities (heterogeneity!) so that complementary content is gonna be eliminated and used as a conglomerate, while filtering out the redundant elements

Multimodal Learning
Challenges

- Data Imbalance: problem if a single modality has significantly more data than another data modality
- Data Noise and Quality: impact of low-quality data in one or more modalities on the learning process and how to sum up the important information from multiple data modalities (heterogeneity!) so that complementary content is gonna be eliminated and used as a conglomerate, while filtering out the redundant elements
- Data Alignment and Synchronization: difficulty in aligning information across different modalities (especially if asynchronously!), next to a best-possible fusion procedure

Multimodal Learning

Challenges

- Data Imbalance: problem if a single modality has significantly more data than another data modality
- Data Noise and Quality: impact of low-quality data in one or more modalities on the learning process and how to sum up the important information from multiple data modalities (heterogeneity!) so that complementary content is gonna be eliminated and used as a conglomerate, while filtering out the redundant elements
- Data Alignment and Synchronization: difficulty in aligning information across different modalities (especially if asynchronously!), next to a best-possible fusion procedure
- Learning Representations: unified representation capturing complementary information w.r.t. all modalities is a problem, because of own structures and characteristics $\rightarrow$ Common feature space hard to specify!

Multimodal Learning
Challenges

- Data Imbalance: problem if a single modality has significantly more data than another data modality
- Data Noise and Quality: impact of low-quality data in one or more modalities on the learning process and how to sum up the important information from multiple data modalities (heterogeneity!) so that complementary content is gonna be eliminated and used as a conglomerate, while filtering out the redundant elements
- Data Alignment and Synchronization: difficulty in aligning information across different modalities (especially if asynchronously!), next to a best-possible fusion procedure
- Learning Representations: unified representation capturing complementary information w.r.t. all modalities is a problem, because of own structures and characteristics $\rightarrow$ Common feature space hard to specify!
- Interpretability: challenges in understanding how decisions are made when combining multiple modalities

Further Questions?

https://www.oth-aw.de/hochschule/ueber-uns/personen/bergler-christian/

Source: https://emekaboris.medium.com/the-intuition-behind-100-days-of-data-science-code-c98402cdc92c