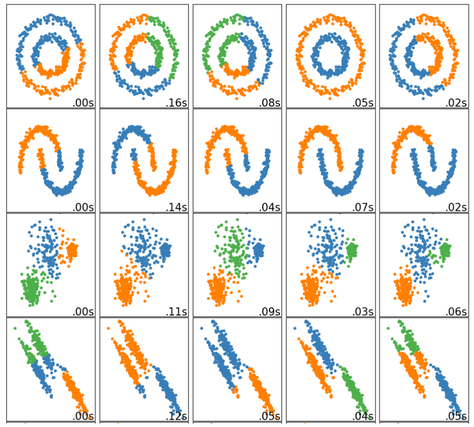
Self-Supervised Learning

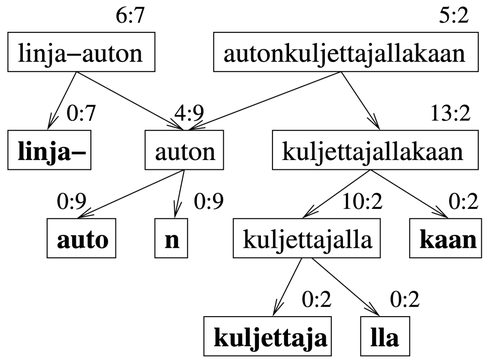
DSCC 251/451: Machine Learning with Limited Data

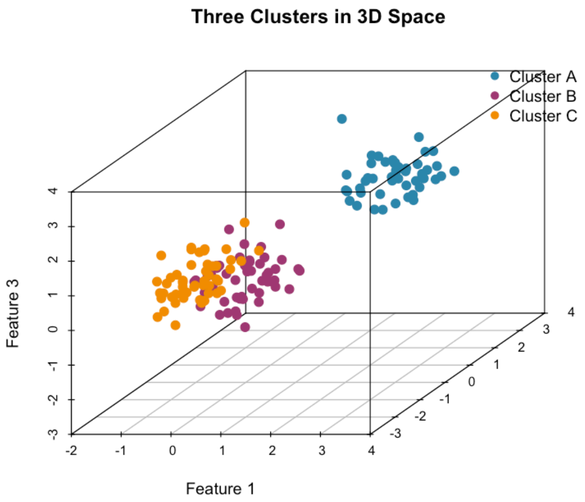
C.M. Downey

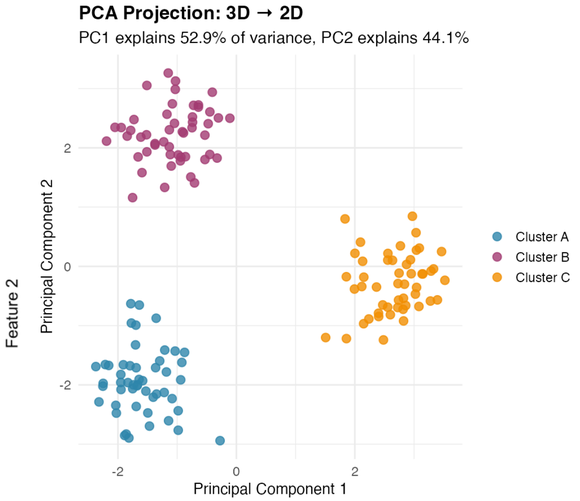
Spring 2026

Last week: Unsupervised Learning



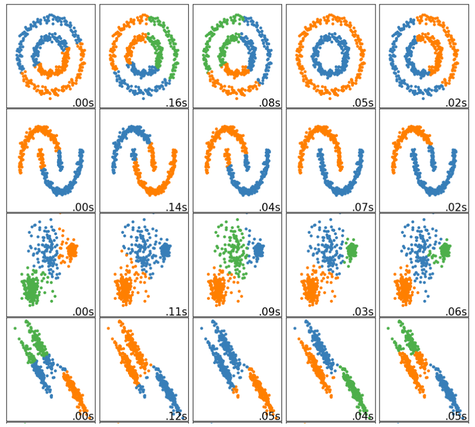


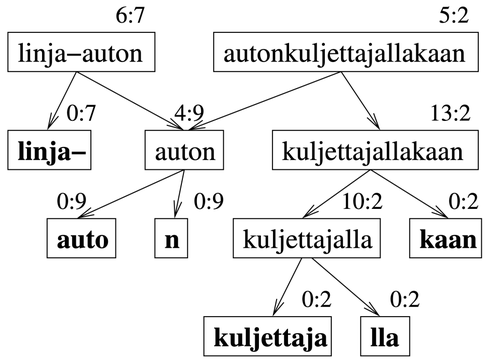


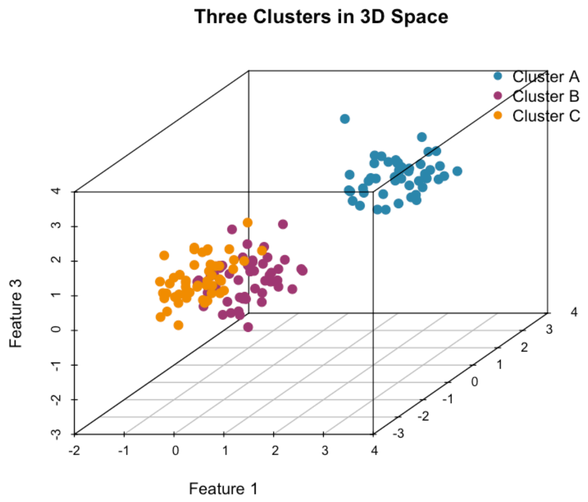


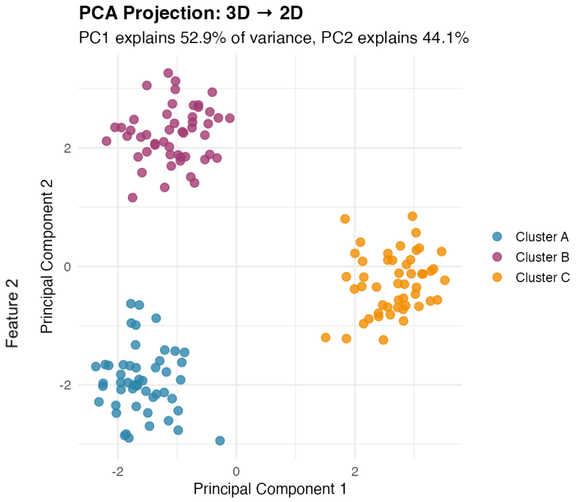
Last week: Unsupervised Learning

- Techniques to **find structure** in **raw data** (i.e. **without labels**)



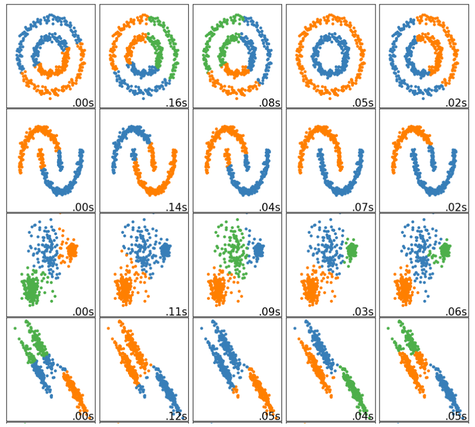


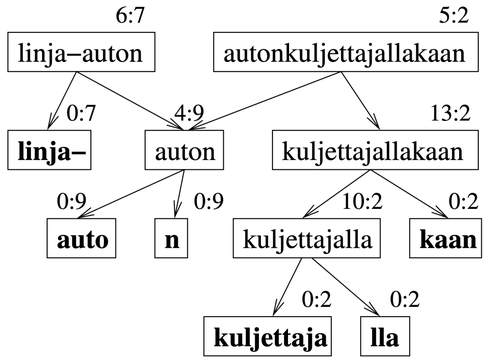


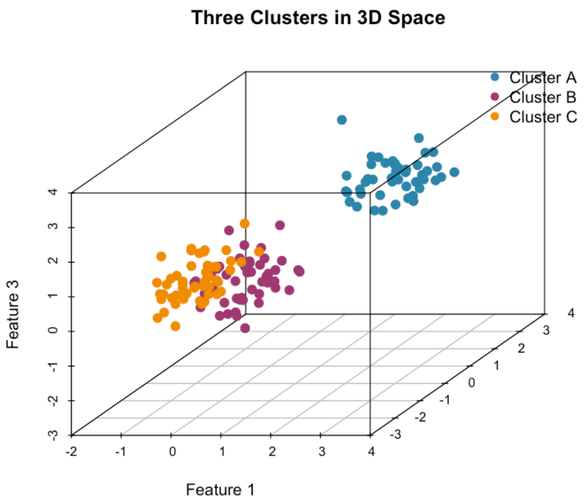


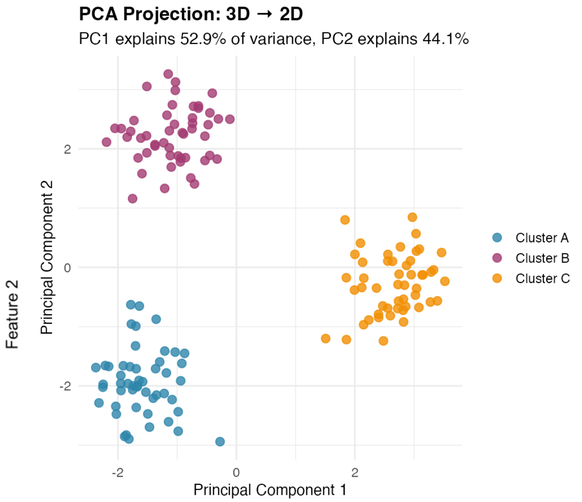
Last week: Unsupervised Learning

- Techniques to **find structure** in **raw data** (i.e. **without labels**)
- Usually requires optimizing a **surrogate objective**
  - E.g. cluster coherence, MDL, variance explanation, reconstruction



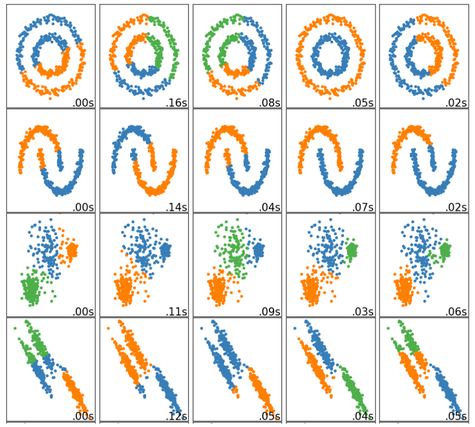


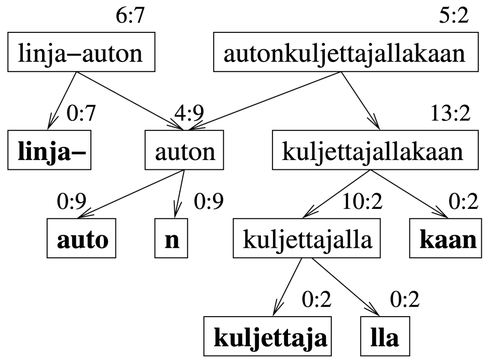


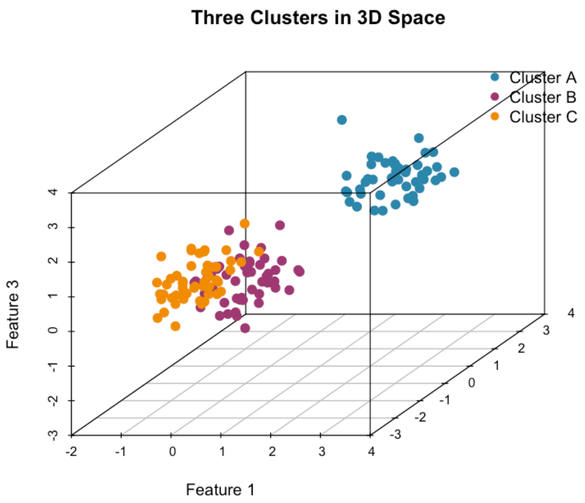


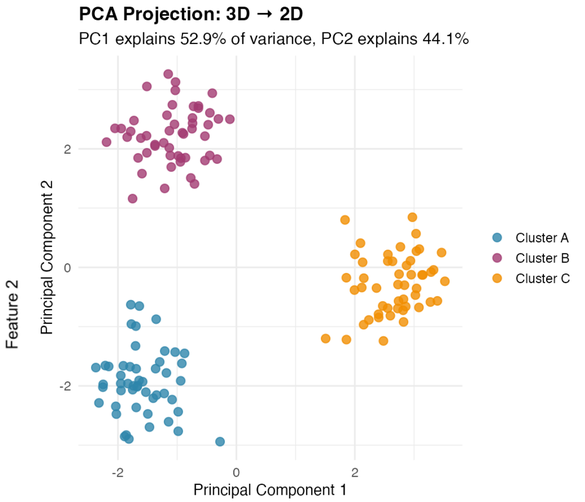
Last week: Unsupervised Learning

- Techniques to **find structure** in **raw data** (i.e. **without labels**)
- Usually requires optimizing a **surrogate objective**
  - E.g. cluster coherence, MDL, variance explanation, reconstruction
- Self-supervised learning is a **type of unsupervised learning** (fits these criteria)

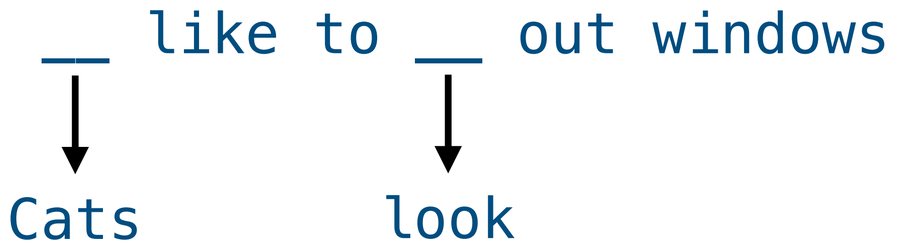


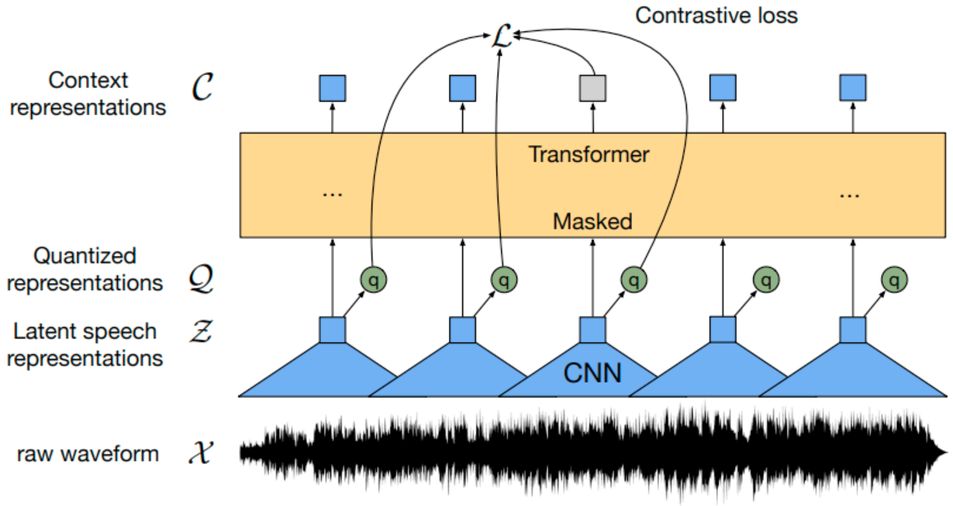






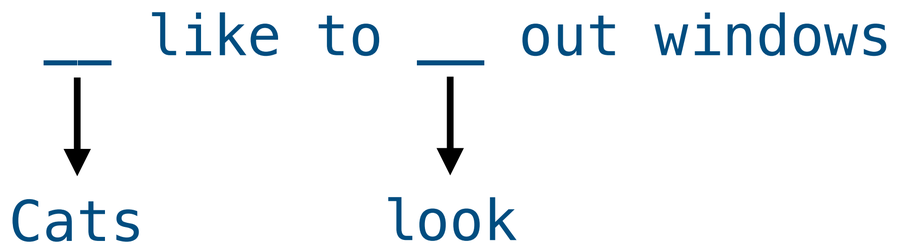
Self-Supervision

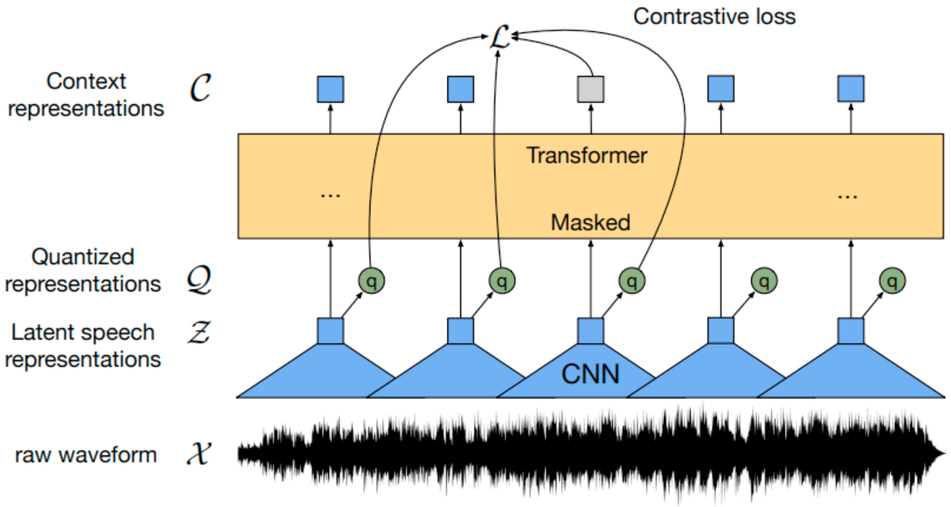




Self-Supervision

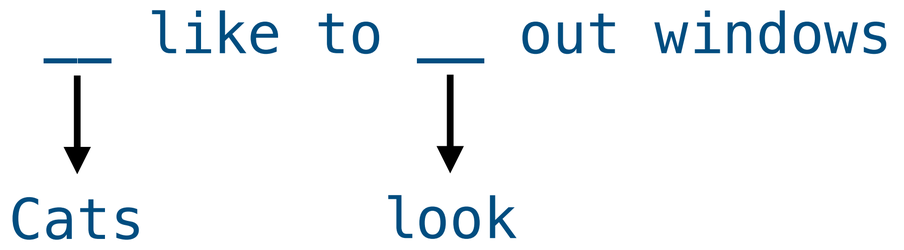
- Idea: create a **supervised learning task** from the **raw data alone**
  - Ex: learn to **predict or reconstruct part of the data**
  - Or classify **similar vs. dissimilar** examples (**Contrastive Learning**)

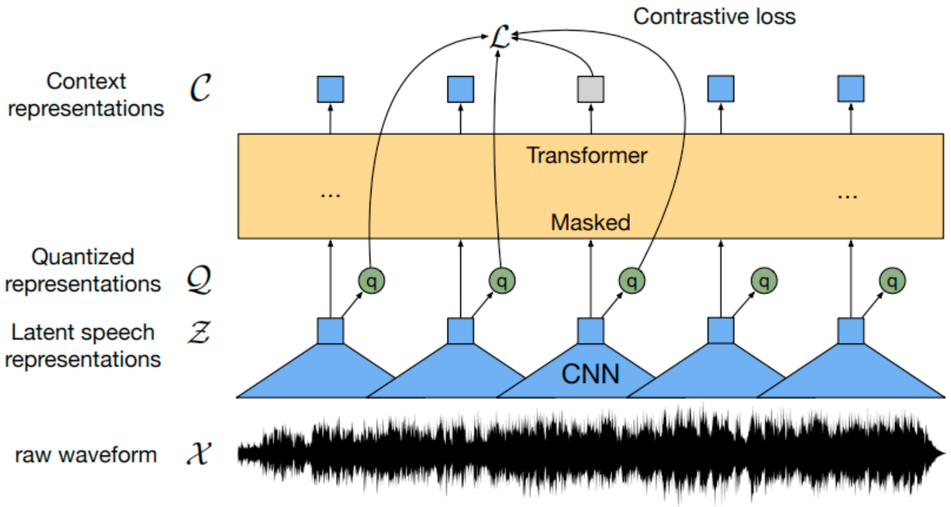




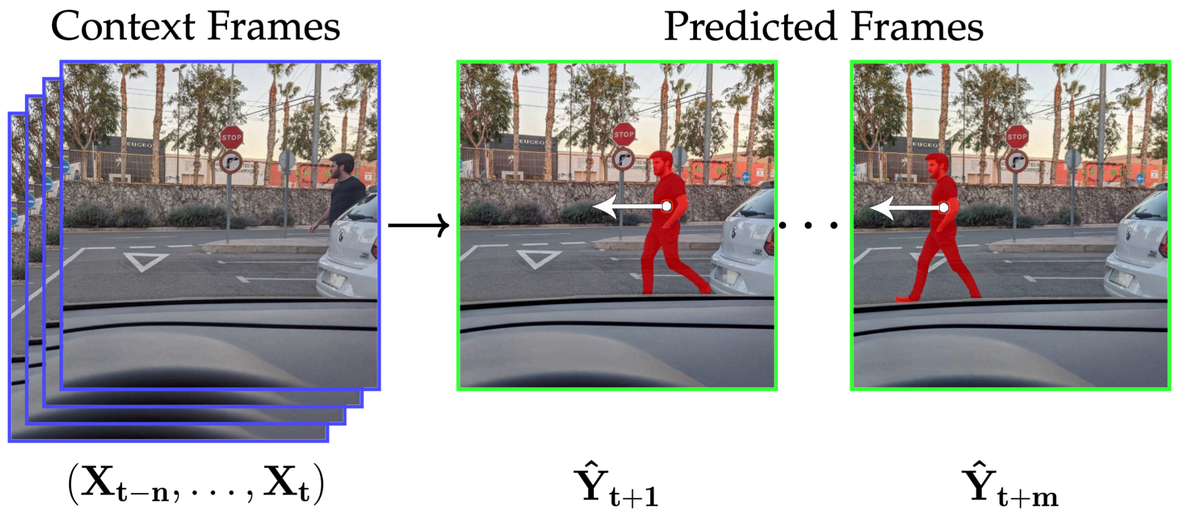
Self-Supervision

- Idea: create a **supervised learning task** from the **raw data alone**
  - Ex: learn to **predict or reconstruct part of the data**
  - Or classify **similar vs. dissimilar** examples (**Contrastive Learning**)
- Hypothesis: forces the model to learn **useful representations** of the data
  - Usually in the context of **Neural Networks**



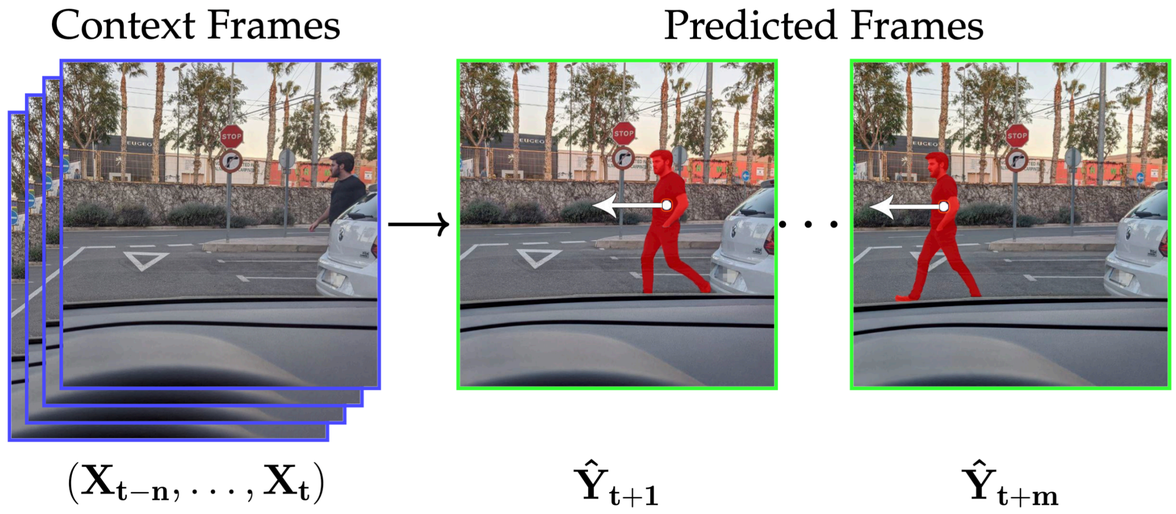


Self-Supervision



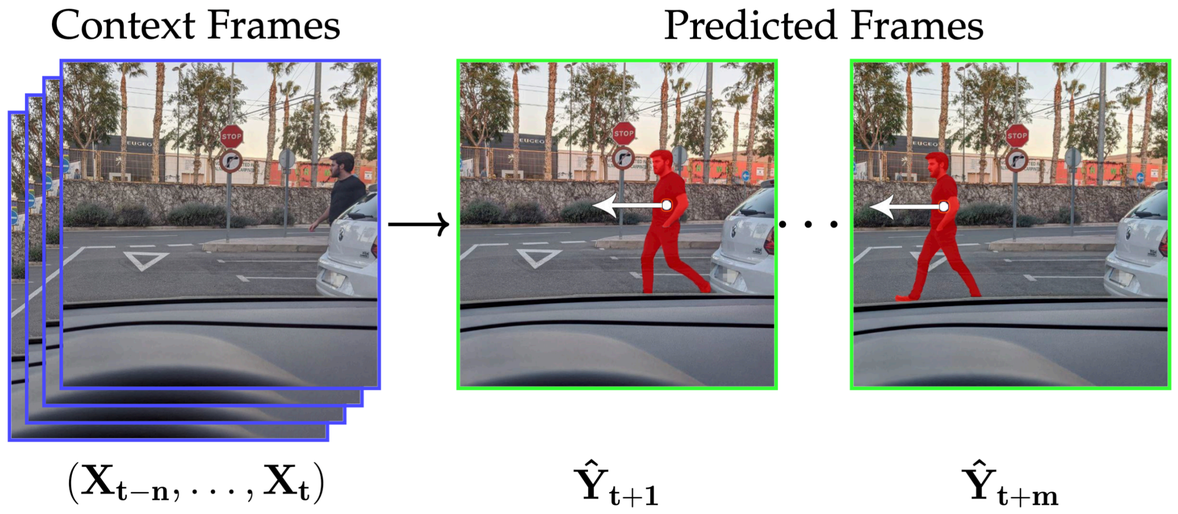
Self-Supervision

- Why should this work? Example:
  video frame prediction



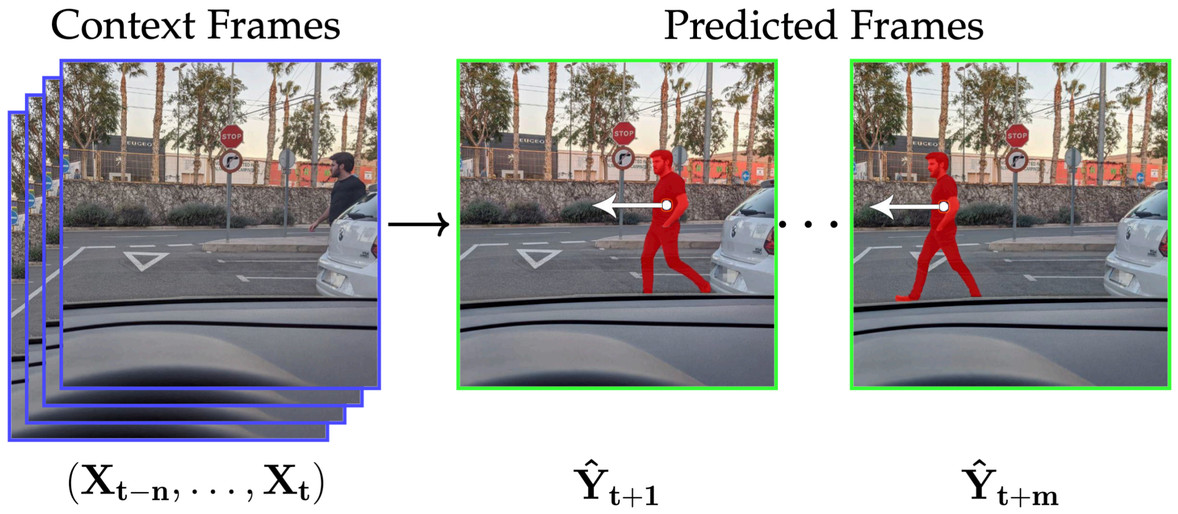
Self-Supervision

- Why should this work? Example: **video frame prediction**
- **What kinds of information** are necessary to successfully predict future frames?
  - Physics, world-knowledge, traffic rules, human behavior, etc. etc.



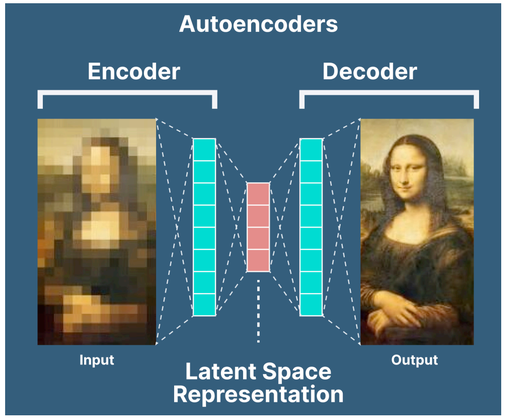
Self-Supervision

- Why should this work? Example: **video frame prediction**
- **What kinds of information** are necessary to successfully predict future frames?
  - Physics, world-knowledge, traffic rules, human behavior, etc. etc.
- The raw data itself is a **goldmine for rich information!**



Self-Supervised vs. Unsupervised

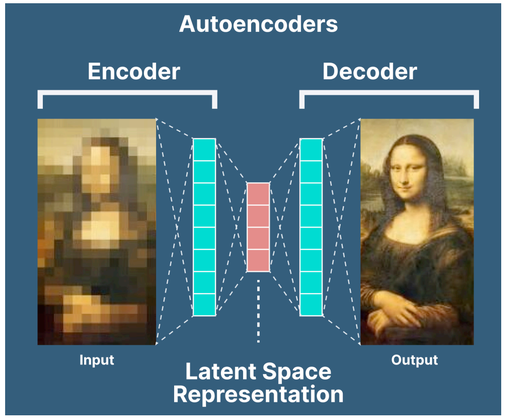
Method	Human Labels?	Constructed Labels?	Self-supervised?
K-means	No	No	No
PCA	No	No	No
Autoencoder	No	Sort of (input = target)	Maybe?
Language Modeling	No	Yes (next word)	Yes
Contrastive Learning	No	Yes (pairs)	Yes



Self-Supervised vs. Unsupervised

- SSL is a **subtype** of unsupervised learning
  - All SSL is unsupervised, but **not all unsupervised learning** is SSL

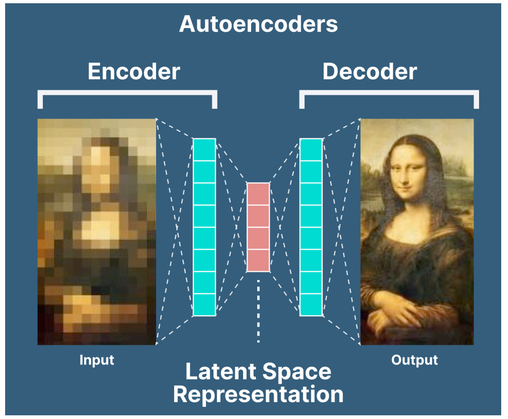
Method	Human Labels?	Constructed Labels?	Self-supervised?
K-means	No	No	No
PCA	No	No	No
Autoencoder	No	Sort of (input = target)	Maybe?
Language Modeling	No	Yes (next word)	Yes
Contrastive Learning	No	Yes (pairs)	Yes



Self-Supervised vs. Unsupervised

- SSL is a **subtype** of unsupervised learning
  - All SSL is unsupervised, but **not all unsupervised learning** is SSL
- Distinguishing feature: SSL constructs **supervisory signal from the data itself**

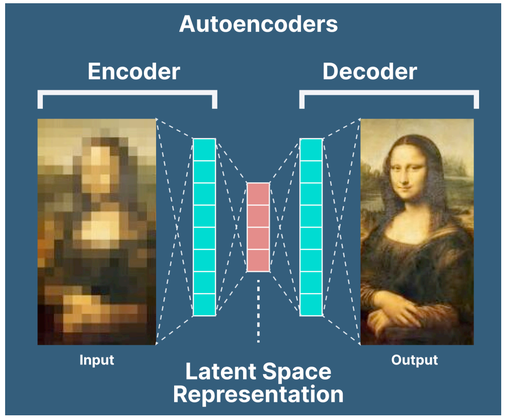
Method	Human Labels?	Constructed Labels?	Self-supervised?
K-means	No	No	No
PCA	No	No	No
Autoencoder	No	Sort of (input = target)	Maybe?
Language Modeling	No	Yes (next word)	Yes
Contrastive Learning	No	Yes (pairs)	Yes



Self-Supervised vs. Unsupervised

- SSL is a **subtype** of unsupervised learning
  - All SSL is unsupervised, but **not all unsupervised learning** is SSL
- Distinguishing feature: SSL constructs **supervisory signal from the data itself**
- **Blurred line: autoencoders**
  - The input **is** the target. Does that count?
  - **Denoising AEs** look even more like SSL

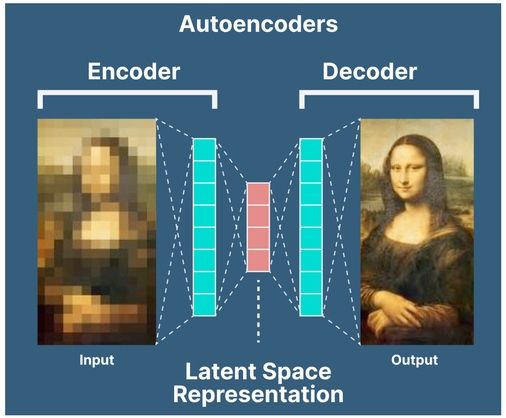
Method	Human Labels?	Constructed Labels?	Self-supervised?
K-means	No	No	No
PCA	No	No	No
Autoencoder	No	Sort of (input = target)	Maybe?
Language Modeling	No	Yes (next word)	Yes
Contrastive Learning	No	Yes (pairs)	Yes



Self-Supervised vs. Unsupervised

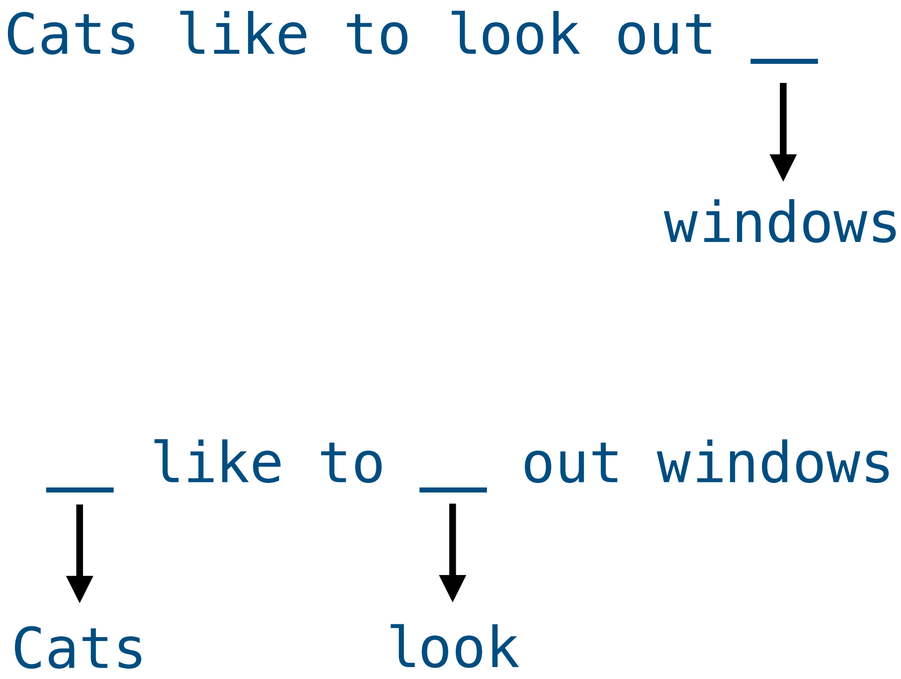
- SSL is a **subtype** of unsupervised learning
  - All SSL is unsupervised, but **not all unsupervised learning** is SSL
- Distinguishing feature: SSL constructs **supervisory signal from the data itself**
- **Blurred line: autoencoders**
  - The input **is** the target. Does that count?
  - **Denoising AEs** look even more like SSL
- In practice: terms are **often used interchangeably** (don't get hung up on it)

Method	Human Labels?	Constructed Labels?	Self-supervised?
K-means	No	No	No
PCA	No	No	No
Autoencoder	No	Sort of (input = target)	Maybe?
Language Modeling	No	Yes (next word)	Yes
Contrastive Learning	No	Yes (pairs)	Yes



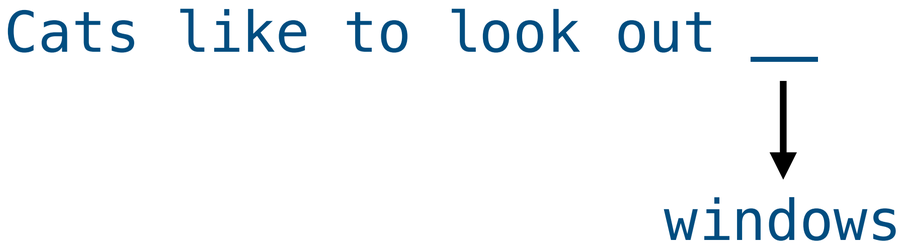
Language Modeling

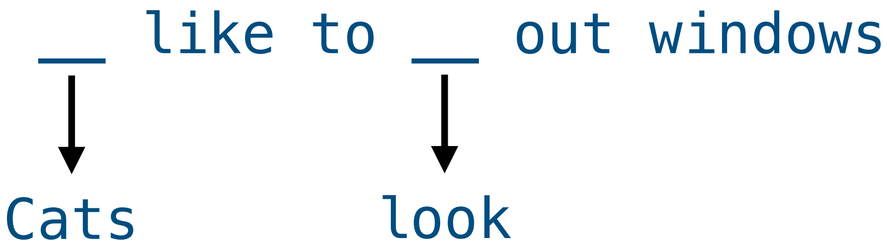
What is Language Modeling?



What is Language Modeling?

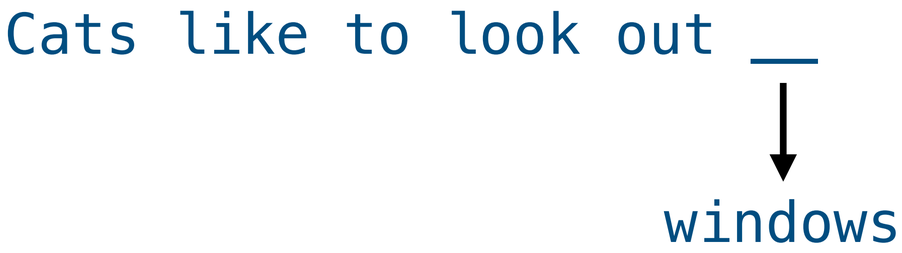
- Simple idea: **predict the next word** based on the preceding words
  - **Also called:** "Autoregressive" LM, "Causal" LM, "Next-Word Prediction"
  - Another variant called **Masked Language Modeling**: predict the **missing word** instead

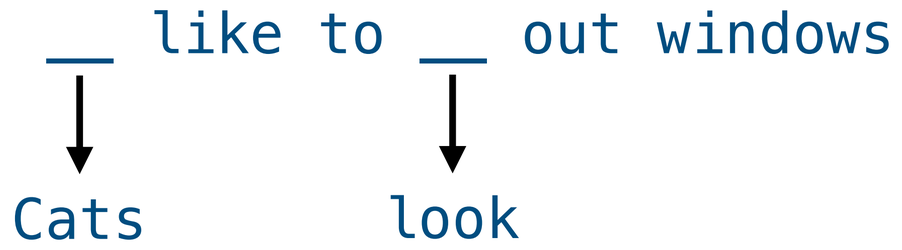




What is Language Modeling?

- Simple idea: **predict the next word** based on the preceding words
  - **Also called:** "Autoregressive" LM, "Causal" LM, "Next-Word Prediction"
  - Another variant called **Masked Language Modeling**: predict the **missing word** instead
- Training data: **any raw text!**
  - Each word is a **training example**
  - A book → millions of examples
  - The internet → trillions+
  - **No human labels required**





Why do Language Modeling?

What word comes next?

The ____

Why do Language Modeling?

What word comes next?

The ____

class
woman
axon
green
colorless
great
⋮

Why do Language Modeling?

What word comes next?

The ____

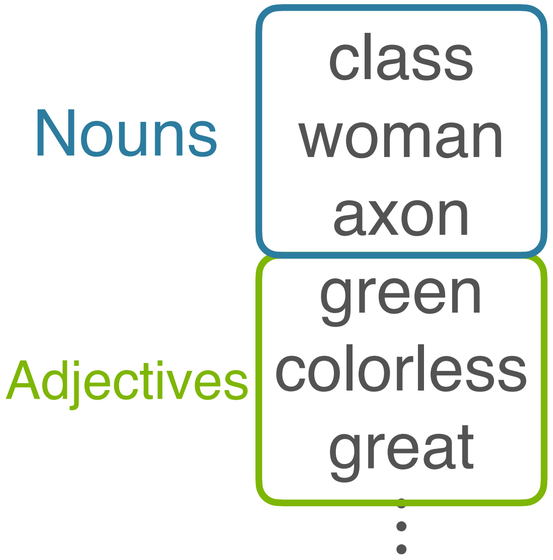
Nouns

class
woman
axon
green
colorless
great
⋮

Why do Language Modeling?

What word comes next?

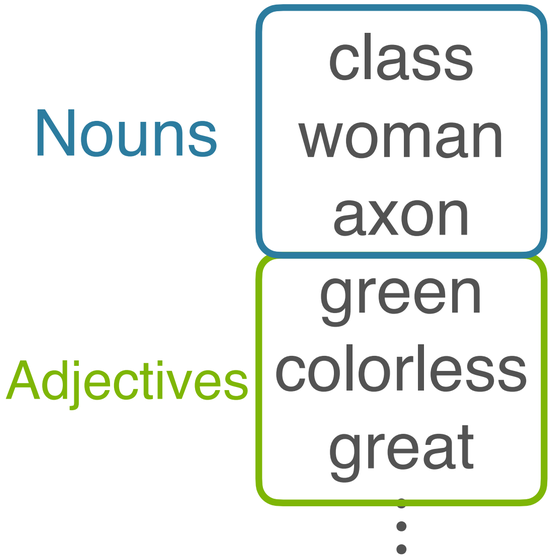
The ____



Why do Language Modeling?

What word comes next?

The ____



We can predict which **Parts of Speech** are likely!

Why do Language Modeling?

What word comes next?

The ____

class
woman
axon
green
colorless
great
⋮

Why do Language Modeling?

What word comes next?

The calico ____

Why do Language Modeling?

What word comes next?

The calico _____

cat
coat
fur
hair
beans
critter
⋮

Why do Language Modeling?

What word comes next?

The calico _____

cat
coat
fur
hair
beans
critter
⋮

Sometimes a **single word** will be almost certain

Why do Language Modeling?

What word comes next?

The calico _____

cat
coat
fur
hair
beans
critter
⋮

Why do Language Modeling?

What word comes next?

The calico cat ____

Why do Language Modeling?

What word comes next?

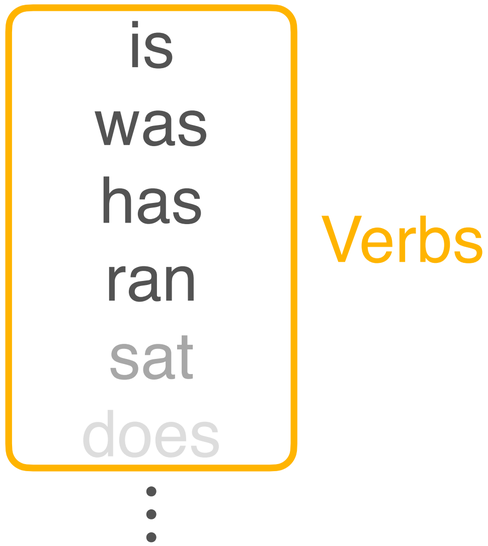
The calico cat _____

is
was
has
ran
sat
does
⋮

Why do Language Modeling?

What word comes next?

The calico cat _____



Why do Language Modeling?

What word comes next?

The calico cat _____

is
was
has
ran
sat
does
⋮

Why do Language Modeling?

What word comes next?

The calico cat sits ____

Why do Language Modeling?

What word comes next?

The calico cat sits _____

on
in
with
still
sat
does
⋮

Why do Language Modeling?

What word comes next?

The calico cat sits _____

Prepositions

Why do Language Modeling?

What word comes next?

The calico cat sits _____

on
in
with
still
sat
does
⋮

Why do Language Modeling?

What word comes next?

The calico cat sits on _____

Why do Language Modeling?

What word comes next?

The calico cat sits on _____

a
the
my
her
me
its
⋮

Why do Language Modeling?

What word comes next?

The calico cat sits on the ____

Why do Language Modeling?

What word comes next?

The calico cat sits on the ____

chair
ledge
window
mat
high
soft
⋮

Why do Language Modeling?

What word comes next?

The calico cat sits on the _____

places a cat might like

Why do Language Modeling?

What word comes next?

The calico cat sits on the ____

chair
ledge
window
mat
high
soft
⋮

Why do Language Modeling?

What word comes next?

The calico cat sits on the sunny ____

Why do Language Modeling?

What word comes next?

The calico cat sits on the sunny ____

window
patio
porch
spot
ledge
roof
⋮

Why do Language Modeling?

What word comes next?

The calico cat sits on the sunny ____

potentially sunny

Why do Language Modeling?

What word comes next?

The calico cat sits on the sunny ____

window
patio
porch
spot
ledge
roof
⋮

Why do Language Modeling?

What word comes next?

The calico cat sits on the sunny ledge ____

Why do Language Modeling?

What word comes next?

The calico cat sits on the sunny ledge ____

.
!
</s>
every
and
all
⋮

Why do Language Modeling?

What word comes next?

The calico cat sits on the sunny ledge lately

Why do Language Modeling?

Why do Language Modeling?

- To make better predictions, an LM encodes **grammatical, semantic, and "real-world" knowledge** at every step
  - We'll see this makes them great **general-purpose models**
  - Also **interesting to Linguists** for the types of language structure they encode

Why do Language Modeling?

- To make better predictions, an LM encodes **grammatical, semantic, and "real-world" knowledge** at every step
  - We'll see this makes them great **general-purpose models**
  - Also **interesting to Linguists** for the types of language structure they encode
- Making good predictions allows them to **generate new language**
  - This has caused the explosion of **Generative AI** and **Chatbots**

Why do Language Modeling?

- To make better predictions, an LM encodes **grammatical, semantic, and "real-world" knowledge** at every step
  - We'll see this makes them great **general-purpose models**
  - Also **interesting to Linguists** for the types of language structure they encode
- Making good predictions allows them to **generate new language**
  - This has caused the explosion of **Generative AI** and **Chatbots**
- This bears out the **"Self-supervision Hypothesis"**
  - The "simple" next-word prediction task forces the model to **learn useful representations of the data**

Language Modeling (Technical Definition)

Language Modeling (Technical Definition)

- A Language Model makes a **probabilistic prediction** about a **missing component** of a **sequence of symbols**

Language Modeling (Technical Definition)

- A Language Model makes a **probabilistic prediction** about a **missing component** of a **sequence of symbols**
  - "probabilistic": assigns a **probability** to **each possible prediction**

Language Modeling (Technical Definition)

- A Language Model makes a **probabilistic prediction** about a **missing component** of a **sequence of symbols**
  - "probabilistic": assigns a **probability** to **each possible prediction**
  - "missing component": usually this is the **next symbol in the sequence**, given a certain prefix

Language Modeling (Technical Definition)

- A Language Model makes a **probabilistic prediction** about a **missing component** of a **sequence of symbols**
  - "probabilistic": assigns a **probability** to **each possible prediction**
  - "missing component": usually this is the **next symbol in the sequence**, given a certain prefix
    - BUT: some LMs predict a **missing word**, rather than the next one

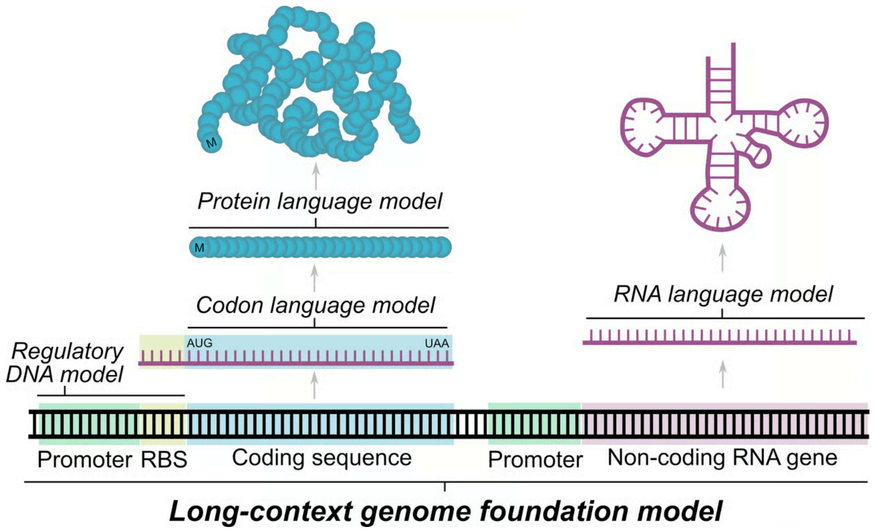
Language Modeling (Technical Definition)

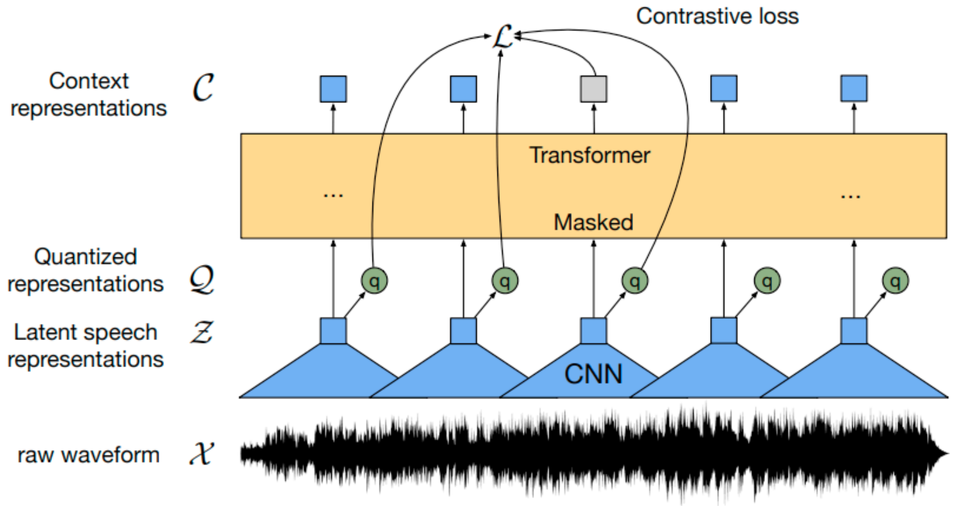
- A Language Model makes a **probabilistic prediction** about a **missing component** of a **sequence of symbols**
  - "probabilistic": assigns a **probability** to **each possible prediction**
  - "missing component": usually this is the **next symbol in the sequence**, given a certain prefix
    - BUT: some LMs predict a **missing word**, rather than the next one
  - "sequence of symbols": any **ordered sequence** of **discrete units**

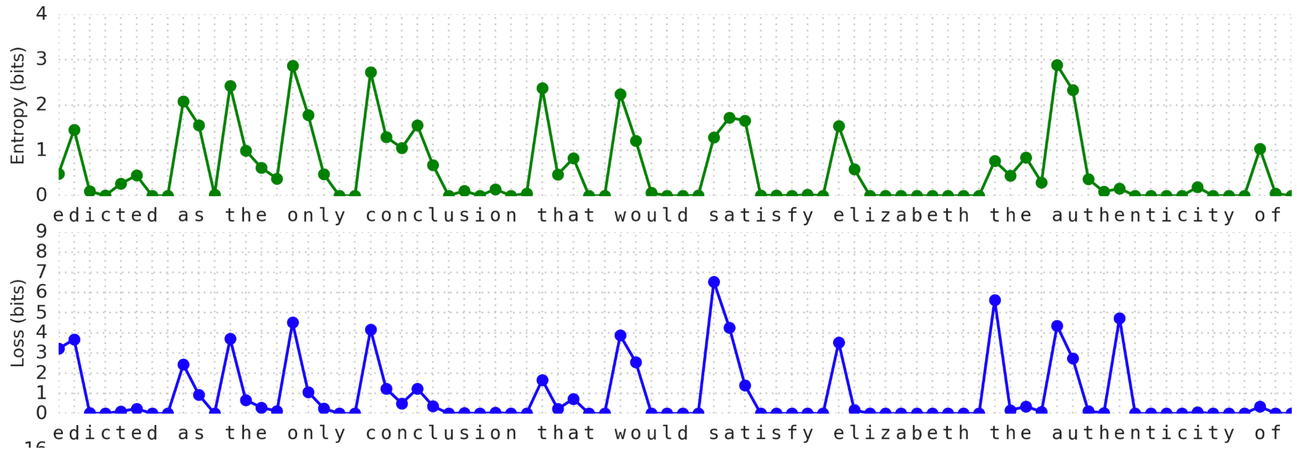
Language Modeling (Technical Definition)

- A Language Model makes a **probabilistic prediction** about a **missing component** of a **sequence of symbols**
  - "probabilistic": assigns a **probability** to **each possible prediction**
  - "missing component": usually this is the **next symbol in the sequence**, given a certain prefix
    - BUT: some LMs predict a **missing word**, rather than the next one
  - "sequence of symbols": any **ordered sequence** of **discrete units**
    - Often words, but sometimes **characters, "sub-words", sound units, DNA**

Language Models Without Words



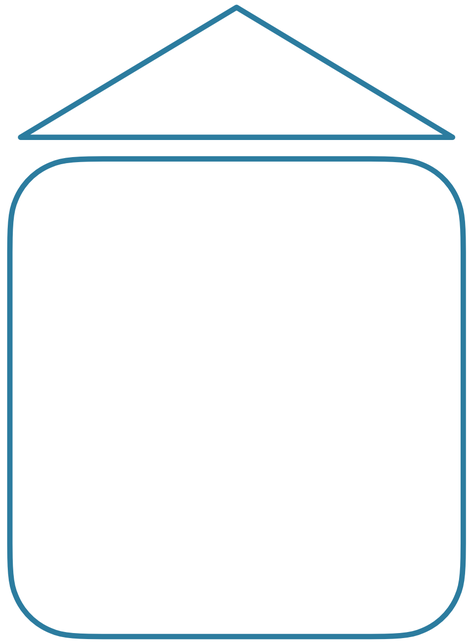




(Language Model) Pre-Training

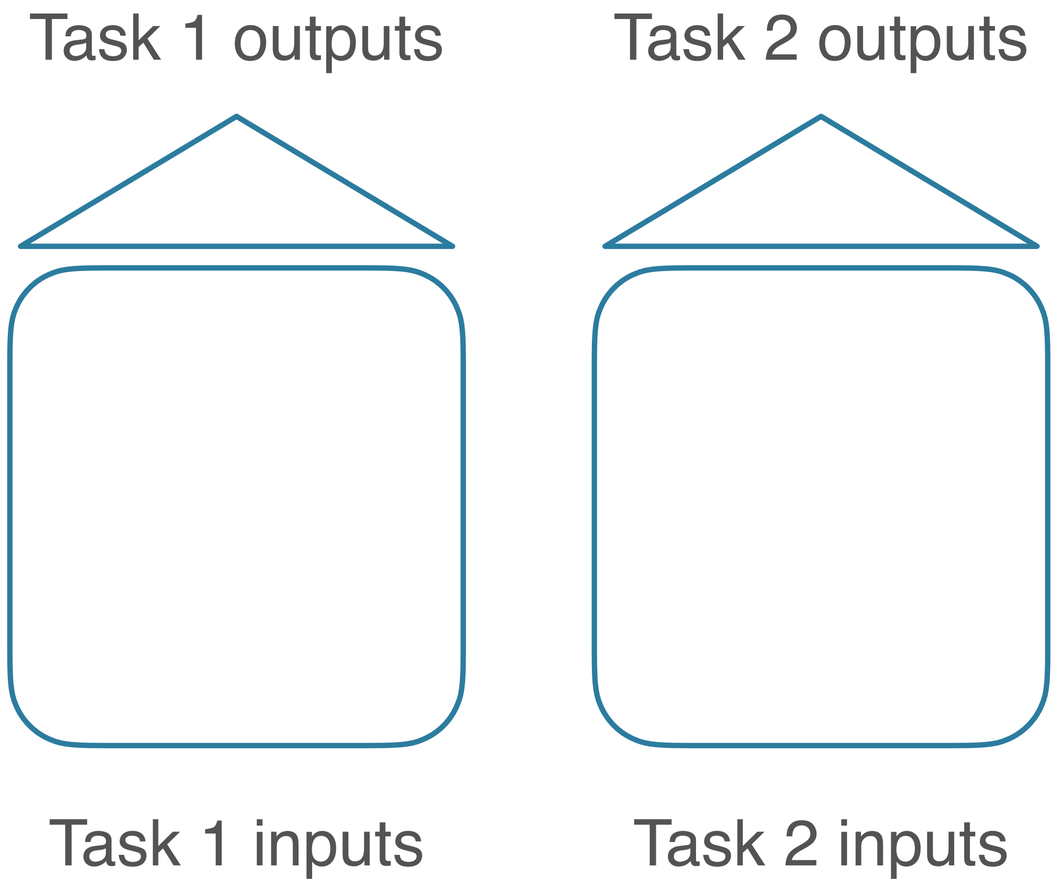
Traditional Learning

Task 1 outputs

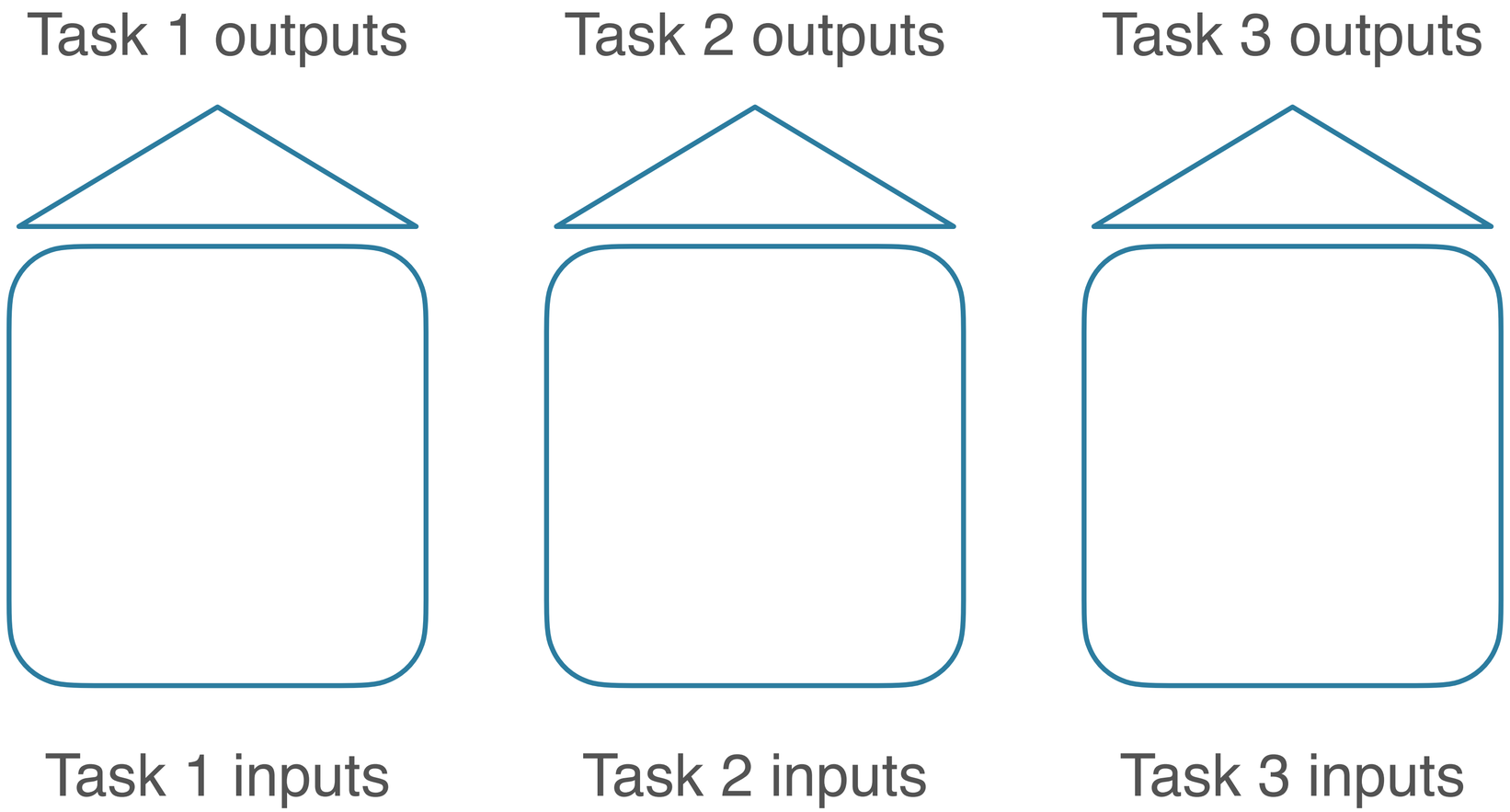


Task 1 inputs

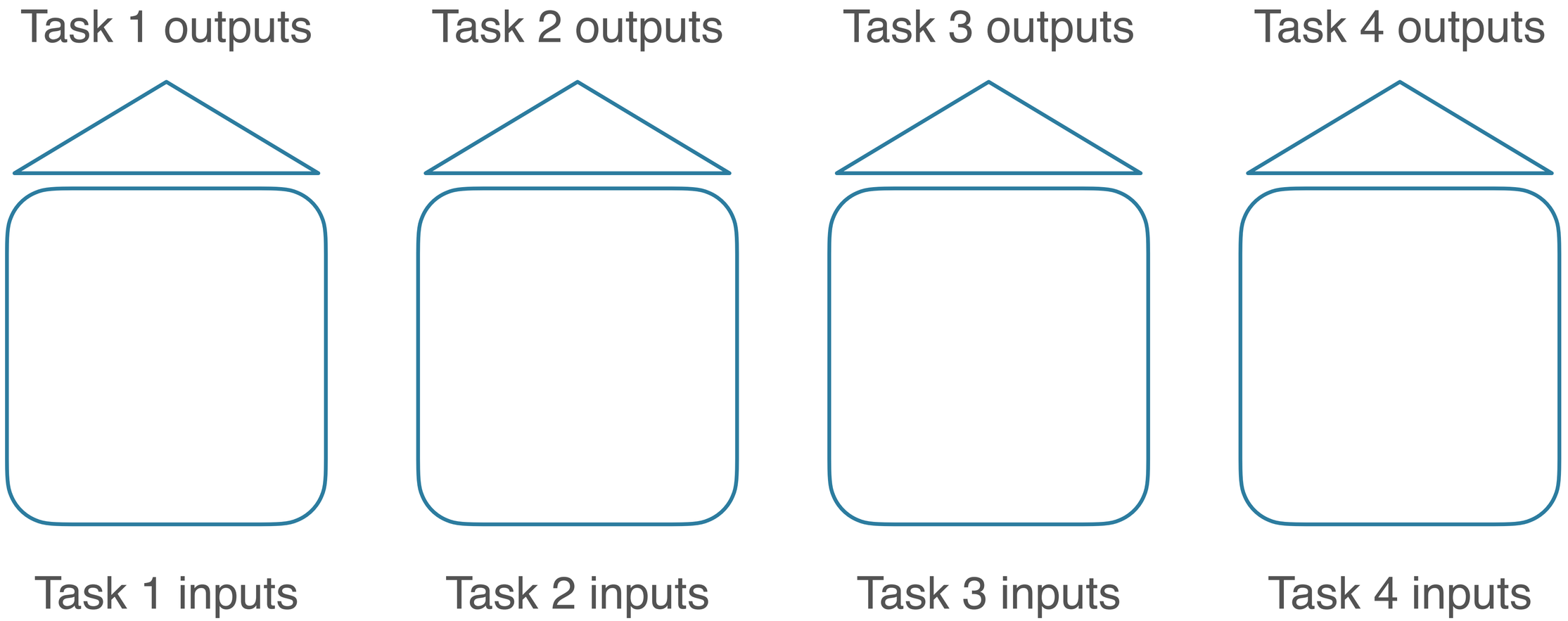
Traditional Learning



Traditional Learning

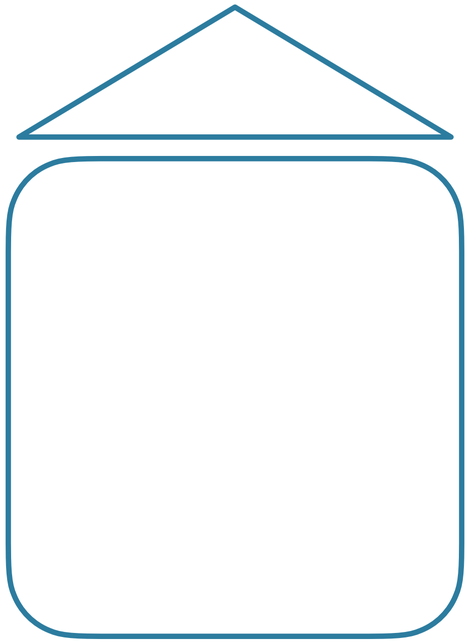


Traditional Learning



Pre-training / Representation Learning

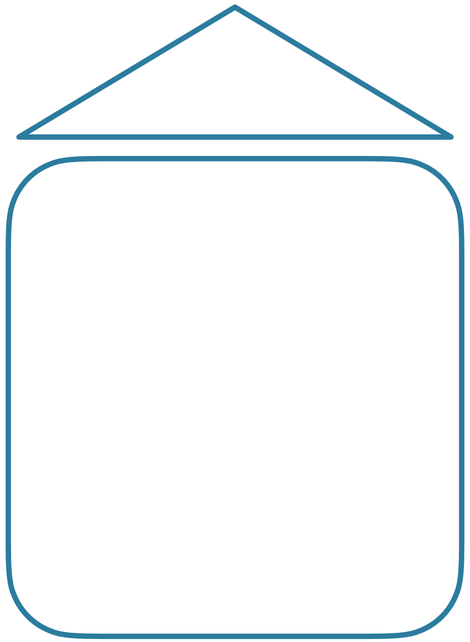
“pre-training” task outputs



“pre-training” task inputs

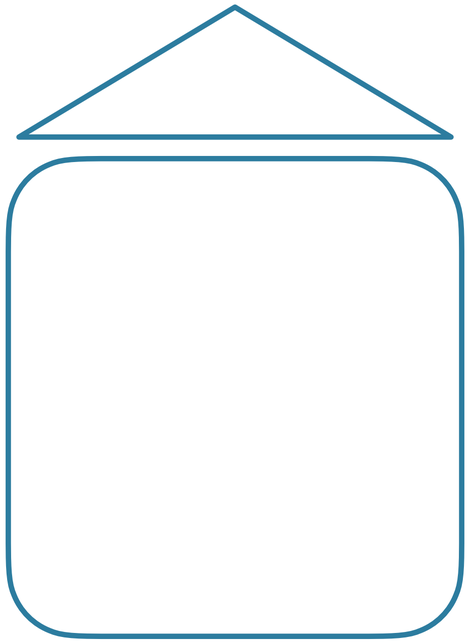
Pre-training / Representation Learning

“pre-training” task outputs



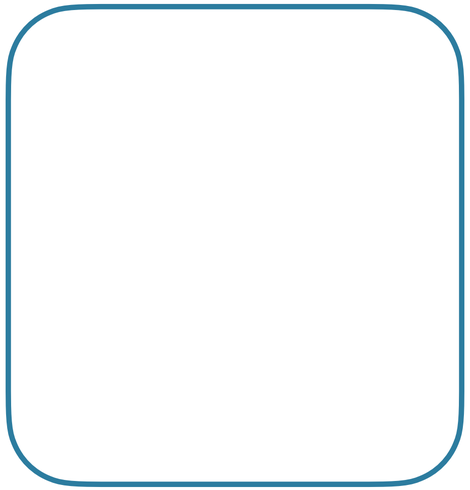
Pre-training / Representation Learning

“pre-training” task outputs



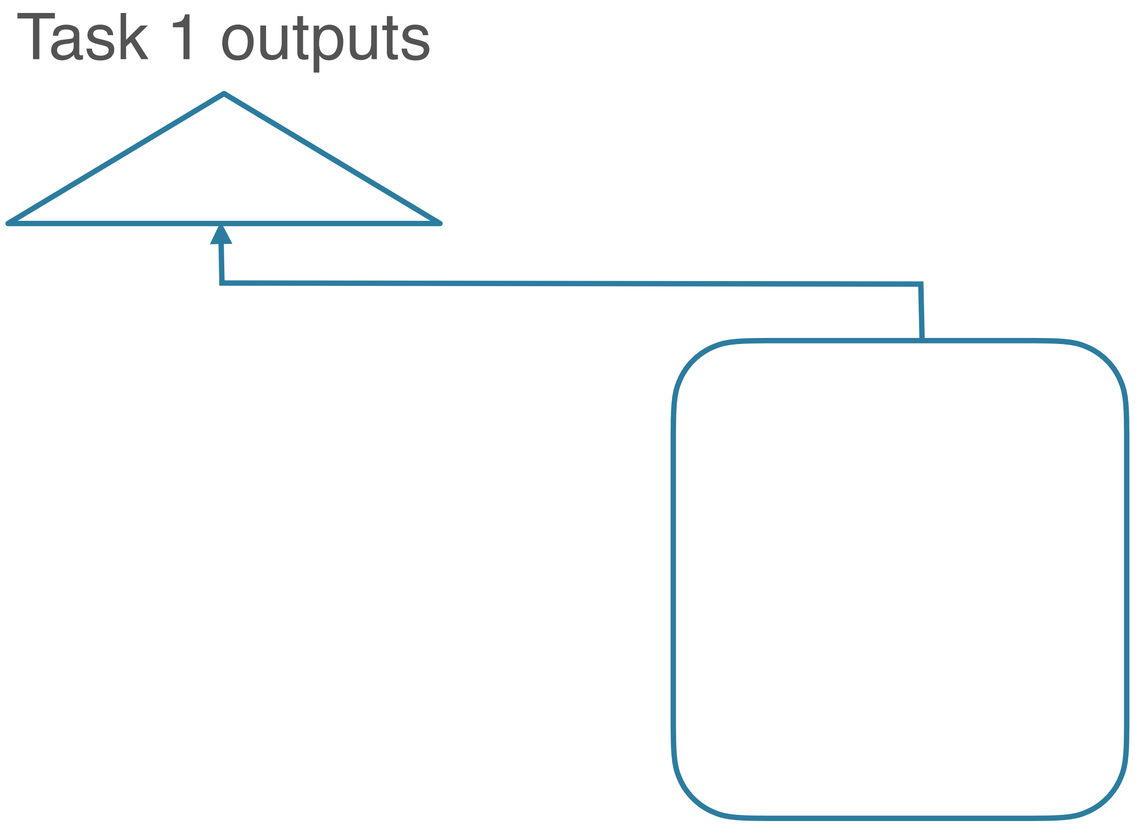
Task 1 inputs

Pre-training / Representation Learning

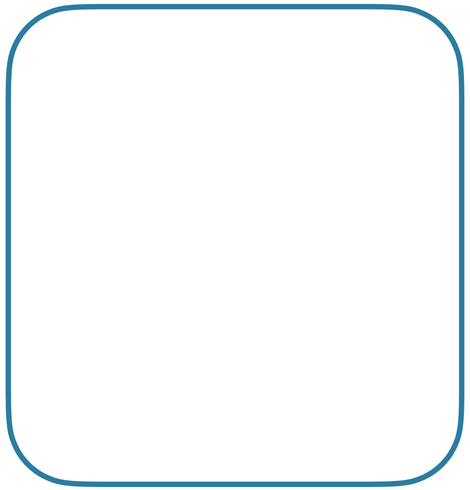


Task 1 inputs

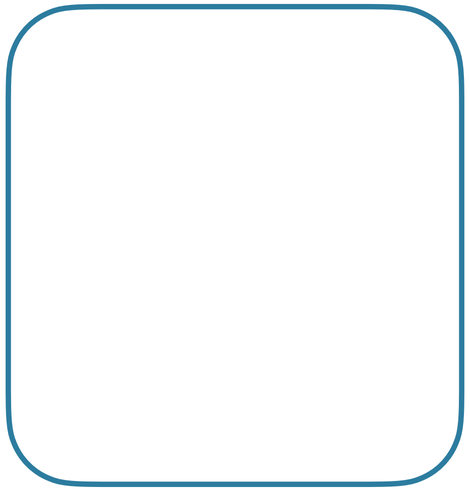
Pre-training / Representation Learning



Pre-training / Representation Learning

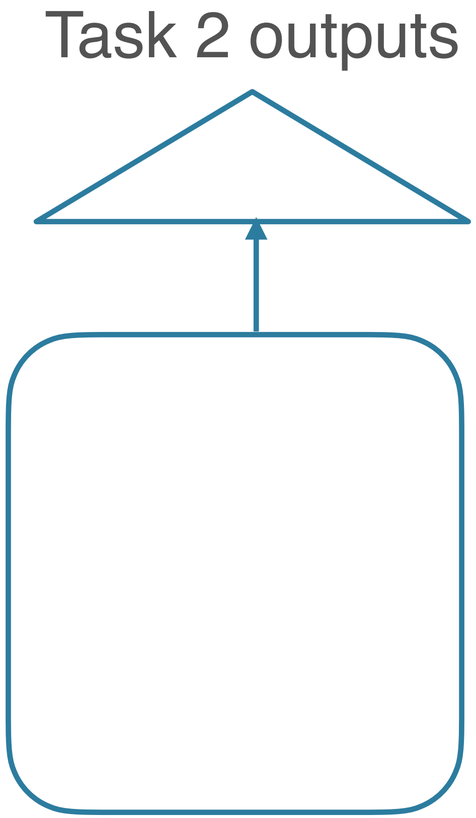


Pre-training / Representation Learning

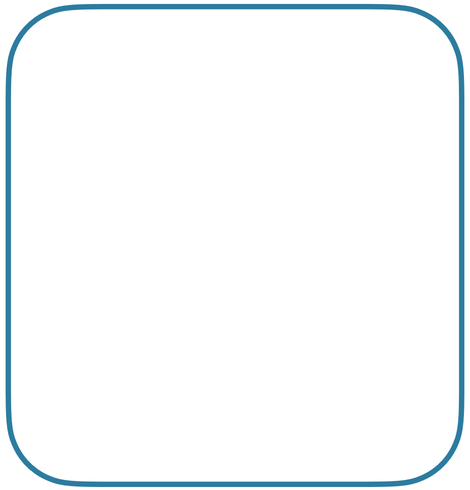


Task 2 inputs

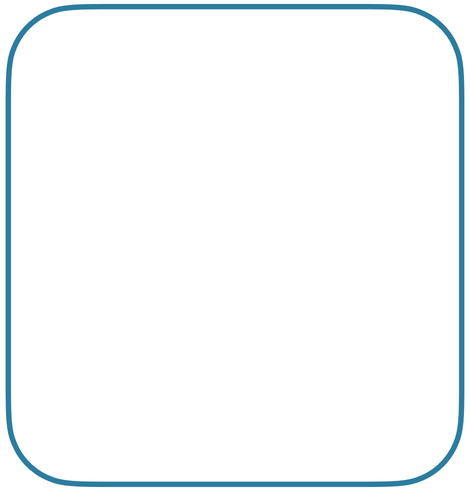
Pre-training / Representation Learning



Pre-training / Representation Learning

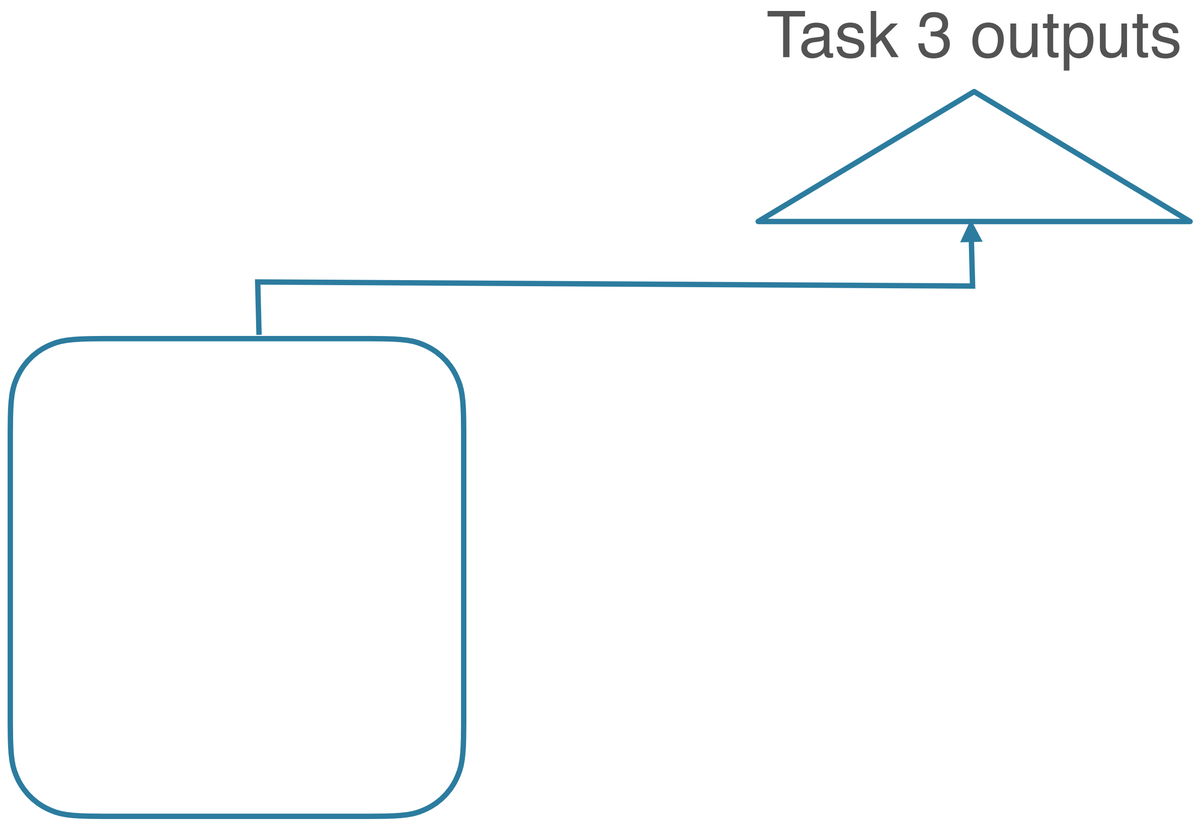


Pre-training / Representation Learning

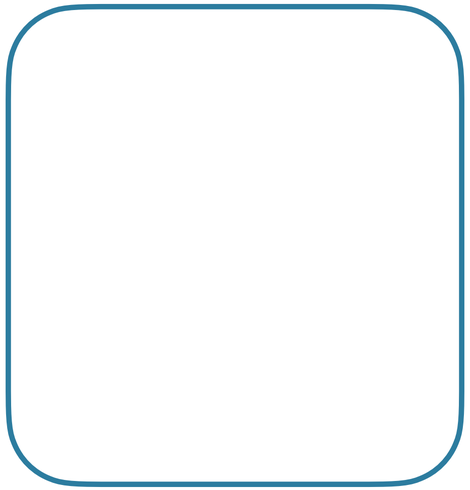


Task 3 inputs

Pre-training / Representation Learning



Pre-training / Representation Learning



Pre-training + Fine-tuning

Pre-training + Fine-tuning

- Step 1: **pre-train** a model on a “general” task
  - Questions: which task for pre-training?
  - Goal: produce **general-purpose representations** of the input (“representation learning”), that will be useful when “transferred” to a more specific task.

Pre-training + Fine-tuning

- Step 1: **pre-train** a model on a “general” task
  - Questions: which task for pre-training?
  - Goal: produce **general-purpose representations** of the input (“representation learning”), that will be useful when “transferred” to a more specific task.
- Step 2: **fine-tune** that model on the main task
  - Replace the “head” of the model with some **task-specific layers**
  - Run supervised training with the resulting model

Where to transfer *from*?

Where to transfer *from*?

- Goal: find a task that will build **general-purpose** and **transferable representations**

Where to transfer *from*?

- Goal: find a task that will build **general-purpose** and **transferable representations**
- Possibilities:

Where to transfer *from*?

- Goal: find a task that will build **general-purpose** and **transferable representations**
- Possibilities:
  - Constituency or dependency parsing

Where to transfer *from*?

- Goal: find a task that will build **general-purpose** and **transferable representations**
- Possibilities:
  - Constituency or dependency parsing
  - Semantic parsing

Where to transfer *from*?

- Goal: find a task that will build **general-purpose** and **transferable representations**
- Possibilities:
  - Constituency or dependency parsing
  - Semantic parsing
  - Machine translation

Where to transfer *from*?

- Goal: find a task that will build **general-purpose** and **transferable representations**
- Possibilities:
  - Constituency or dependency parsing
  - Semantic parsing
  - Machine translation
  - QA

Where to transfer *from*?

- Goal: find a task that will build **general-purpose** and **transferable representations**
- Possibilities:
  - Constituency or dependency parsing
  - Semantic parsing
  - Machine translation
  - QA
  - …

Where to transfer *from*?

- Goal: find a task that will build **general-purpose** and **transferable representations**
- Possibilities:
  - Constituency or dependency parsing
  - Semantic parsing
  - Machine translation
  - QA
  - …
- Scalability issue: all require **expensive annotation**

Language Modeling Advantages

Language Modeling Advantages

- A good language model should produce good **general-purpose** and **transferable** representations

Language Modeling Advantages

- A good language model should produce good **general-purpose** and **transferable** representations
- **Linguistic knowledge**
  - The bicycles, even though old, were in good shape because ____ …
  - The bicycle, even though old, was in good shape because ____ …

Language Modeling Advantages

- A good language model should produce good **general-purpose** and **transferable** representations
- **Linguistic knowledge**
  - The bicycles, even though old, were in good shape because ____ …
  - The bicycle, even though old, was in good shape because ____ …
- **World knowledge**
  - The University of Washington was founded in _____
  - Seattle had a huge population boom as a launching point for expeditions to _____

Language Model Pre-training

Language Model Pre-training

- **Pre-train** a large language model on a large amount of **raw text**
  - **Millions-to-trillions** of training datapoints, for "free" or **relatively cheap**

Language Model Pre-training

- **Pre-train** a large language model on a large amount of **raw text**
  - **Millions-to-trillions** of training datapoints, for "free" or **relatively cheap**
- **Fine-tune** a small model on top of the LM for the **task** you care about
  - Supervised task data: probably only **hundreds-to-thousands** of datapoints, and also **expensive to curate**

Language Model Pre-training

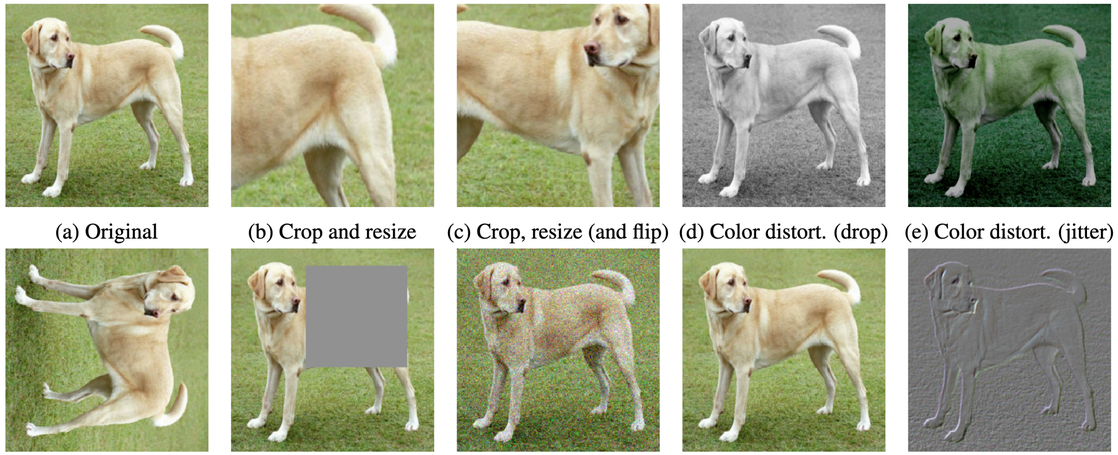
- **Pre-train** a large language model on a large amount of **raw text**
  - **Millions-to-trillions** of training datapoints, for "free" or **relatively cheap**
- **Fine-tune** a small model on top of the LM for the **task** you care about
  - Supervised task data: probably only **hundreds-to-thousands** of datapoints, and also **expensive to curate**
- Most of the **useful learning** is accomplished during **self-supervised pre-training**

Language Model Pre-training

- **Pre-train** a large language model on a large amount of **raw text**
  - **Millions-to-trillions** of training datapoints, for "free" or **relatively cheap**
- **Fine-tune** a small model on top of the LM for the **task** you care about
  - Supervised task data: probably only **hundreds-to-thousands** of datapoints, and also **expensive to curate**
- Most of the **useful learning** is accomplished during **self-supervised pre-training**
- **All mainstream Language Models** are built on SSL
  - Modern chatbots add a few "extra ingredients" on top

(Image) Contrastive Learning

Contrastive Learning



(a) Original (b) Crop and resize (c) Crop, resize (and flip) (d) Color distort. (drop) (e) Color distort. (jitter)

(f) Rotate $\{90°, 180°, 270°\}$ (g) Cutout (h) Gaussian noise (i) Gaussian blur (j) Sobel filtering

Figure 4. Illustrations of the studied data augmentation operators. Each augmentation can transform data stochastically with some internal parameters (e.g. rotation degree, noise level). Note that we *only* test these operators in ablation, the *augmentation policy used to train our models* only includes *random crop (with flip and resize)*, *color distortion*, and *Gaussian blur*. (Original image cc-by: Von.grzanka)

Contrastive Learning

- So far: SSL = **predicting a missing piece of data**
  - Natural for language because of its **intrinsic sequential structure**

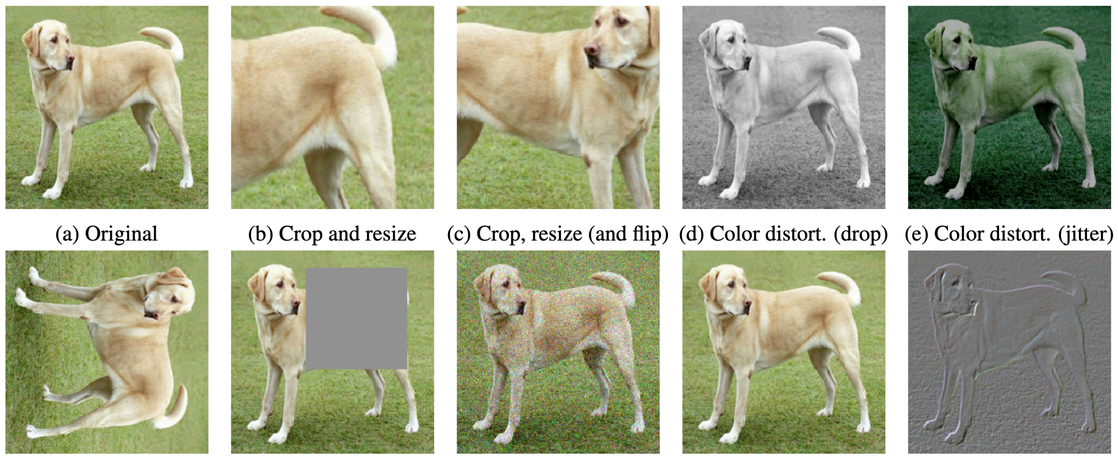


Figure 4. Illustrations of the studied data augmentation operators. Each augmentation can transform data stochastically with some internal parameters (e.g. rotation degree, noise level). Note that we *only* test these operators in ablation, the *augmentation policy used to train our models* only includes *random crop (with flip and resize)*, *color distortion*, and *Gaussian blur*. (Original image cc-by: Von.grzanka)

Contrastive Learning

- So far: SSL = **predicting a missing piece of data**
  - Natural for language because of its **intrinsic sequential structure**
- What about for **image data?**
  - No real "natural ordering"
  - Might be able to **in-fill missing patches**, but models tend to take shortcuts

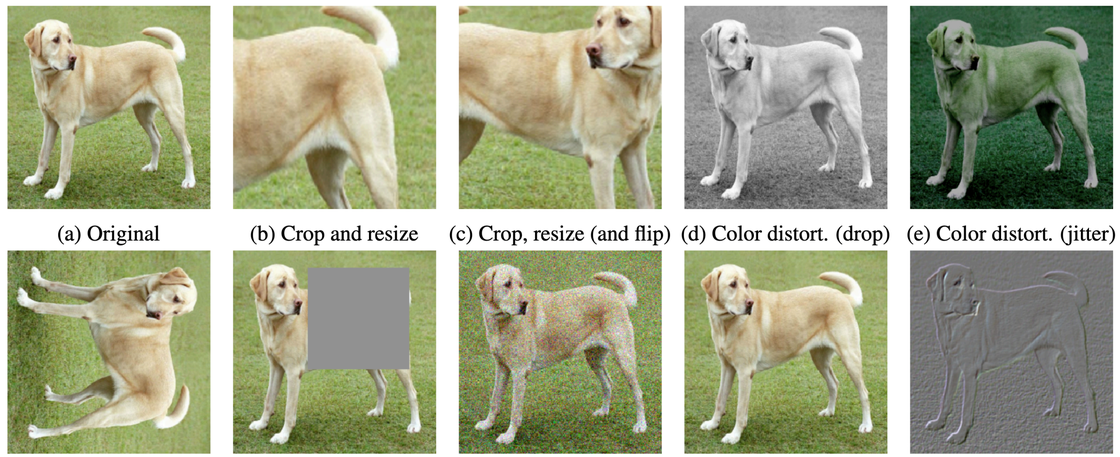


Figure 4. Illustrations of the studied data augmentation operators. Each augmentation can transform data stochastically with some internal parameters (e.g. rotation degree, noise level). Note that we *only* test these operators in ablation, the *augmentation policy used to train our models* only includes *random crop (with flip and resize)*, *color distortion*, and *Gaussian blur*. (Original image cc-by: Von.grzanka)

Contrastive Learning

- So far: SSL = **predicting a missing piece of data**
  - Natural for language because of its **intrinsic sequential structure**
- What about for **image data?**
  - No real "natural ordering"
  - Might be able to **in-fill missing patches**, but models tend to take shortcuts
- Prominent approach: **Contrastive Learning**
  - Apply a **global transformation** to an image
  - Train the model to **classify transformed images as similar, and different images as dissimilar**

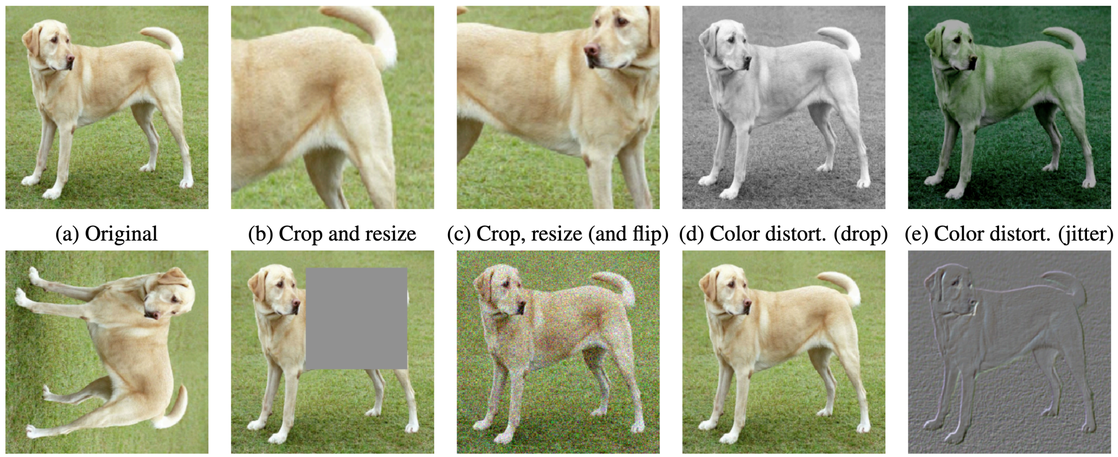


Figure 4. Illustrations of the studied data augmentation operators. Each augmentation can transform data stochastically with some internal parameters (e.g. rotation degree, noise level). Note that we *only* test these operators in ablation, the *augmentation policy used to train our models* only includes *random crop (with flip and resize)*, *color distortion*, and *Gaussian blur*. (Original image cc-by: Von.grzanka)

SimCLR Recipe

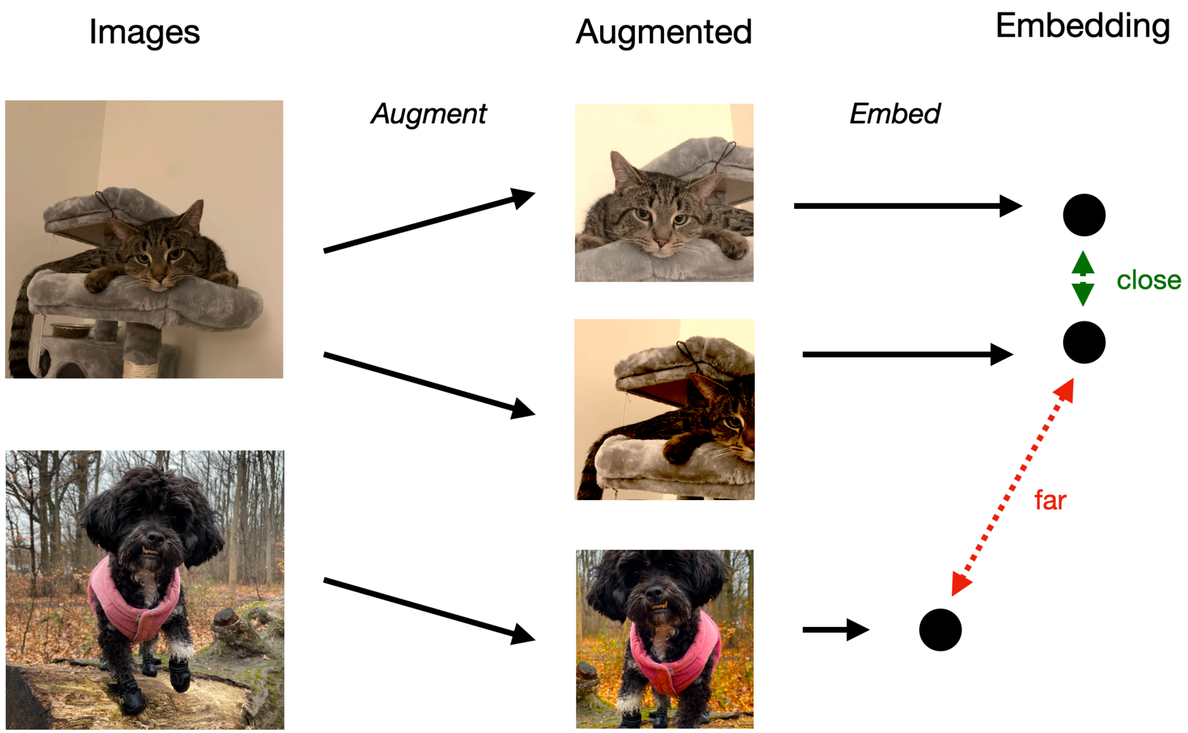


Figure source

SimCLR Recipe

- SimCLR: a conceptually clear contrastive approach from 2020

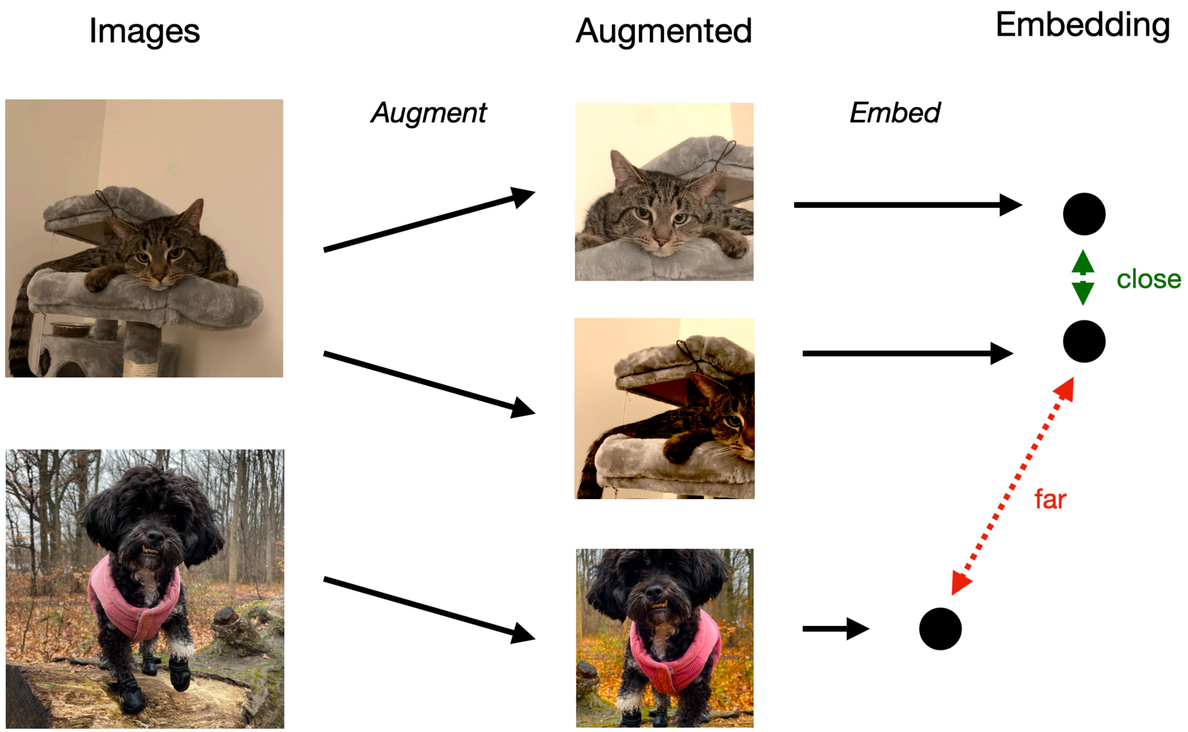


Figure source

SimCLR Recipe

- SimCLR: a conceptually clear contrastive approach from 2020
- For each image in the dataset:

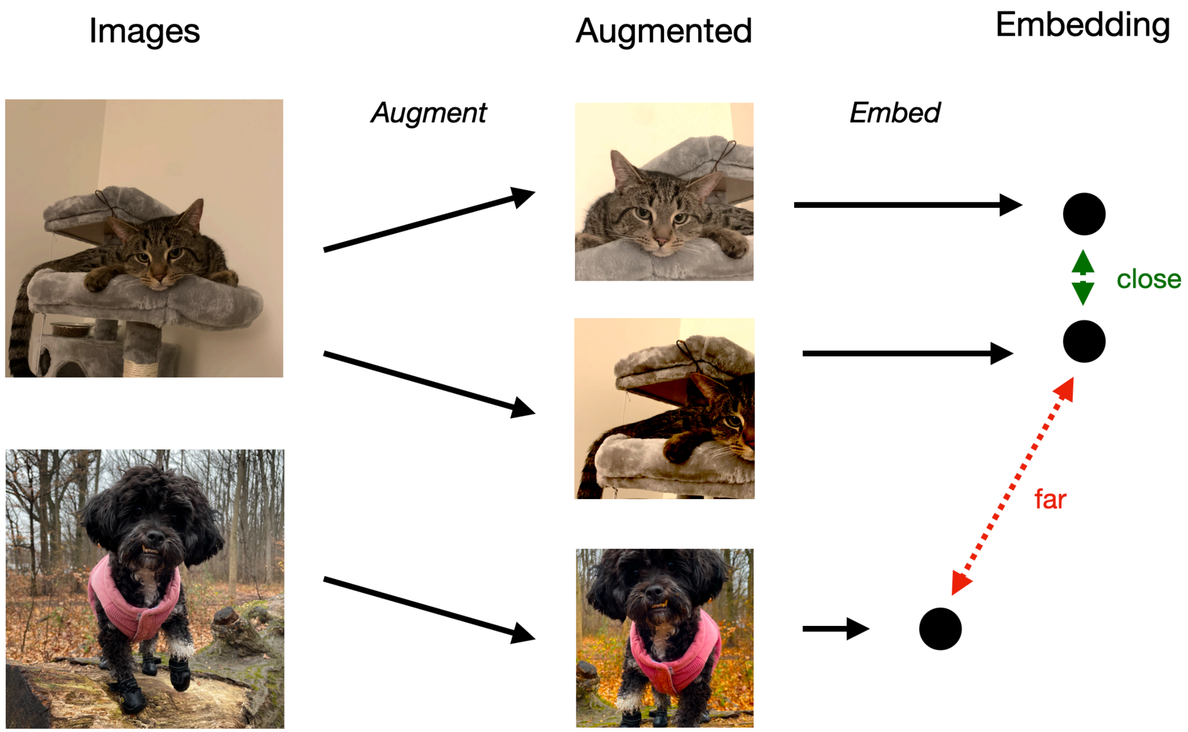


Figure source

SimCLR Recipe

- SimCLR: a conceptually clear contrastive approach from 2020
- For each image in the dataset:
  - Create **random augmentations** (crop, flip, blur, color change, etc.)

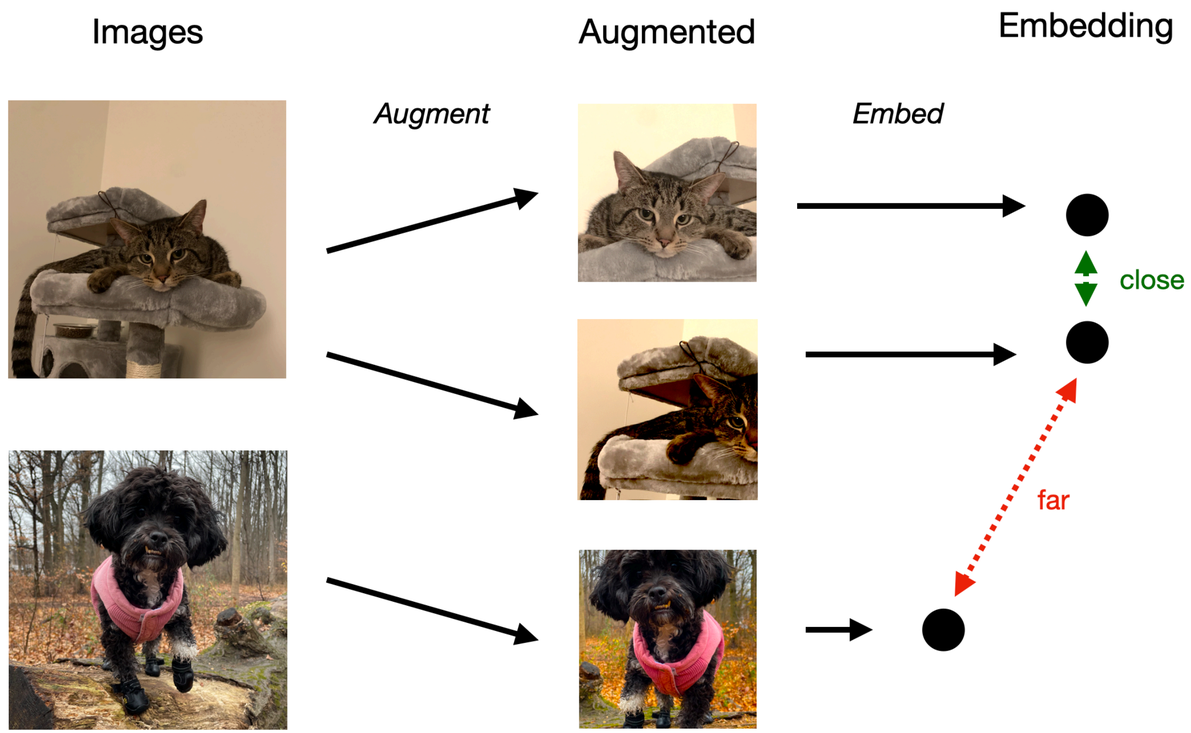


Figure source

SimCLR Recipe

- SimCLR: a conceptually clear contrastive approach from 2020
- For each image in the dataset:
  - Create **random augmentations** (crop, flip, blur, color change, etc.)
  - Pass original and augments through the **same encoder model**

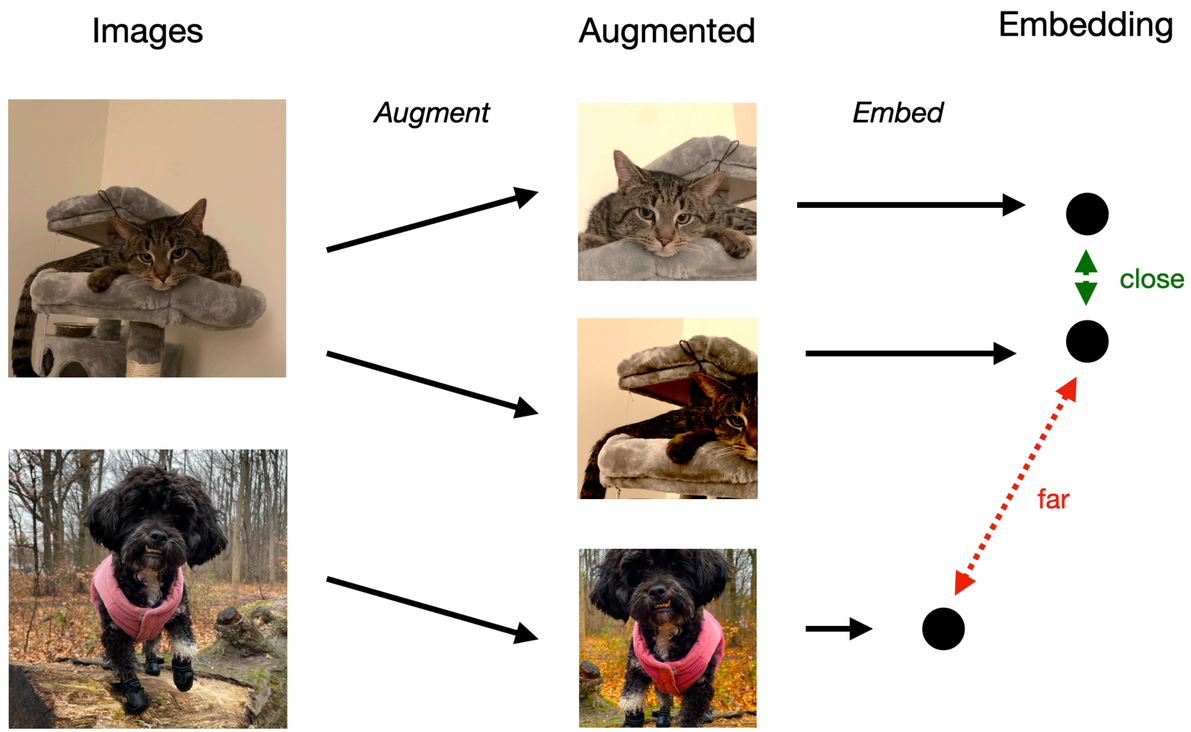


Figure source

SimCLR Recipe

- SimCLR: a conceptually clear contrastive approach from 2020
- For each image in the dataset:
  - Create **random augmentations** (crop, flip, blur, color change, etc.)
  - Pass original and augments through the **same encoder model**
  - Train the embeddings for the **original** and **augments** to be **similar**

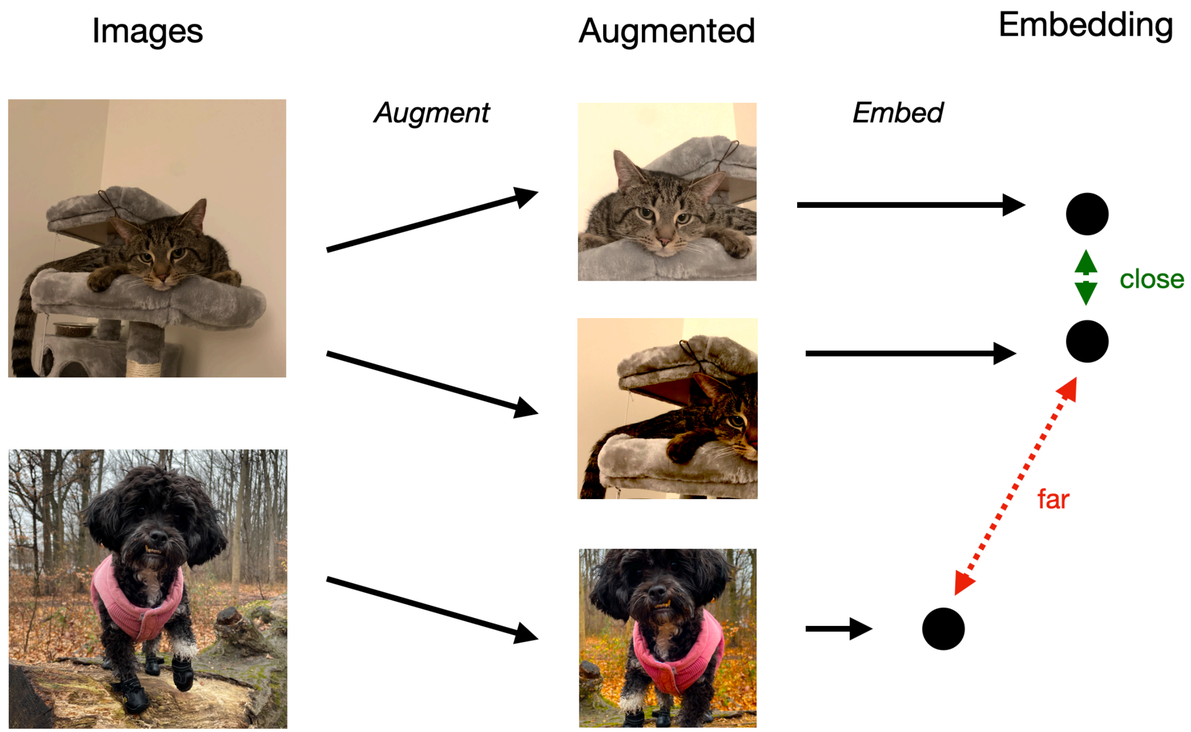


Figure source

SimCLR Recipe

- SimCLR: a conceptually clear contrastive approach from 2020
- For each image in the dataset:
  - Create **random augmentations** (crop, flip, blur, color change, etc.)
  - Pass original and augments through the **same encoder model**
  - Train the embeddings for the **original** and **augments** to be **similar**
  - ...and to be **dissimilar** to other images in the batch

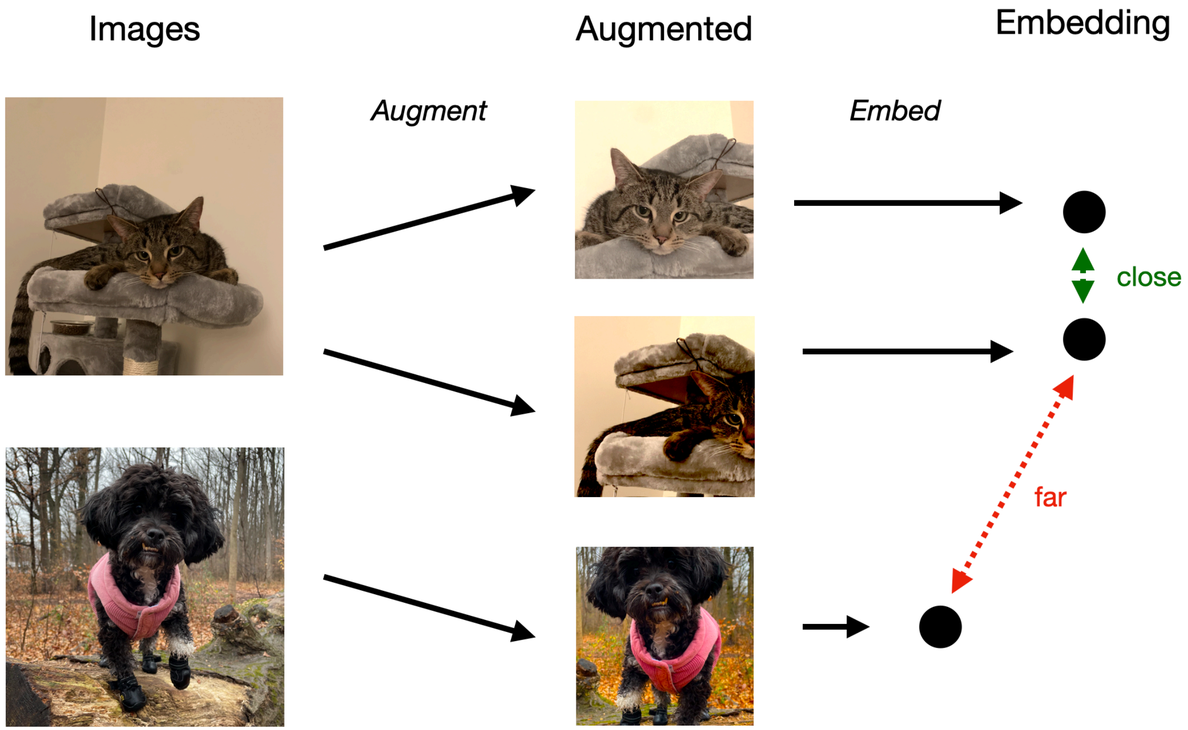


Figure source

Contrastive Learning Intuition

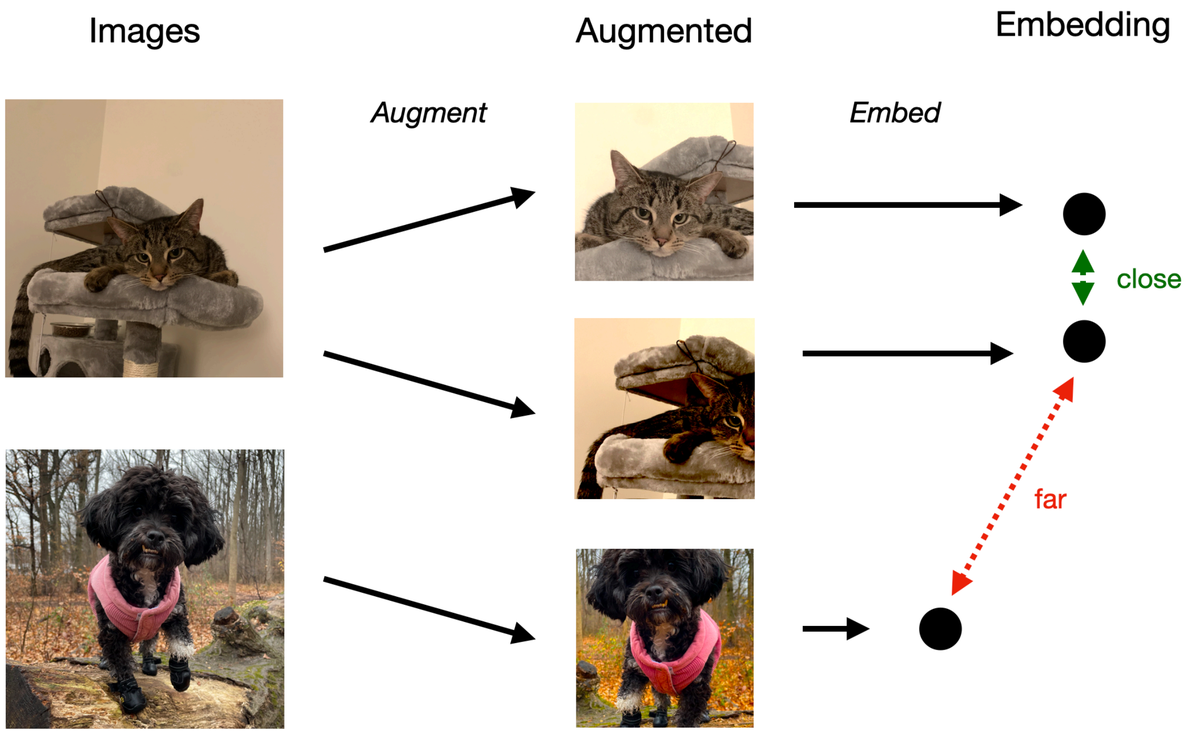


Figure source

Contrastive Learning Intuition

- Model learns aspects of the image that are **invariant** to noise
  - E.g. a blurred cat is still a cat

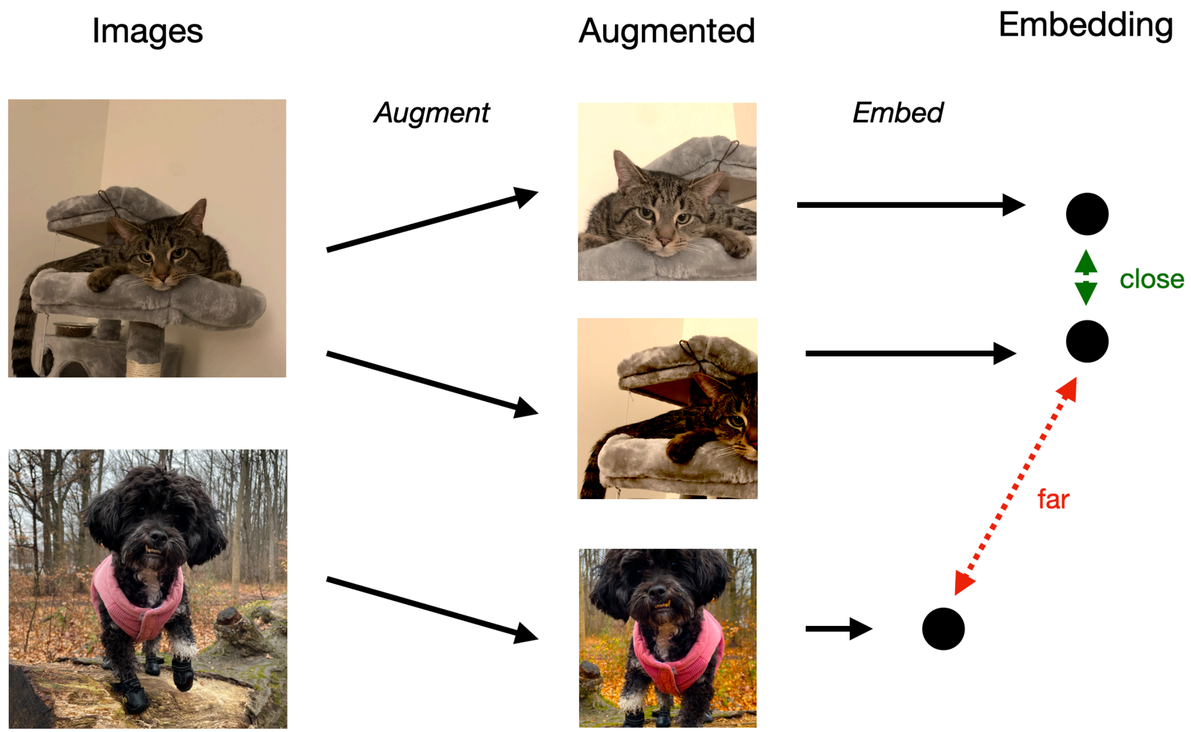


Figure source

Contrastive Learning Intuition

- Model learns aspects of the image that are **invariant** to noise
  - E.g. a blurred cat is still a cat
- Forces it to focus on **semantic aspects** of the image, rather than irrelevant details
  - This is built-in **inductive bias** for what the model should and shouldn't care about

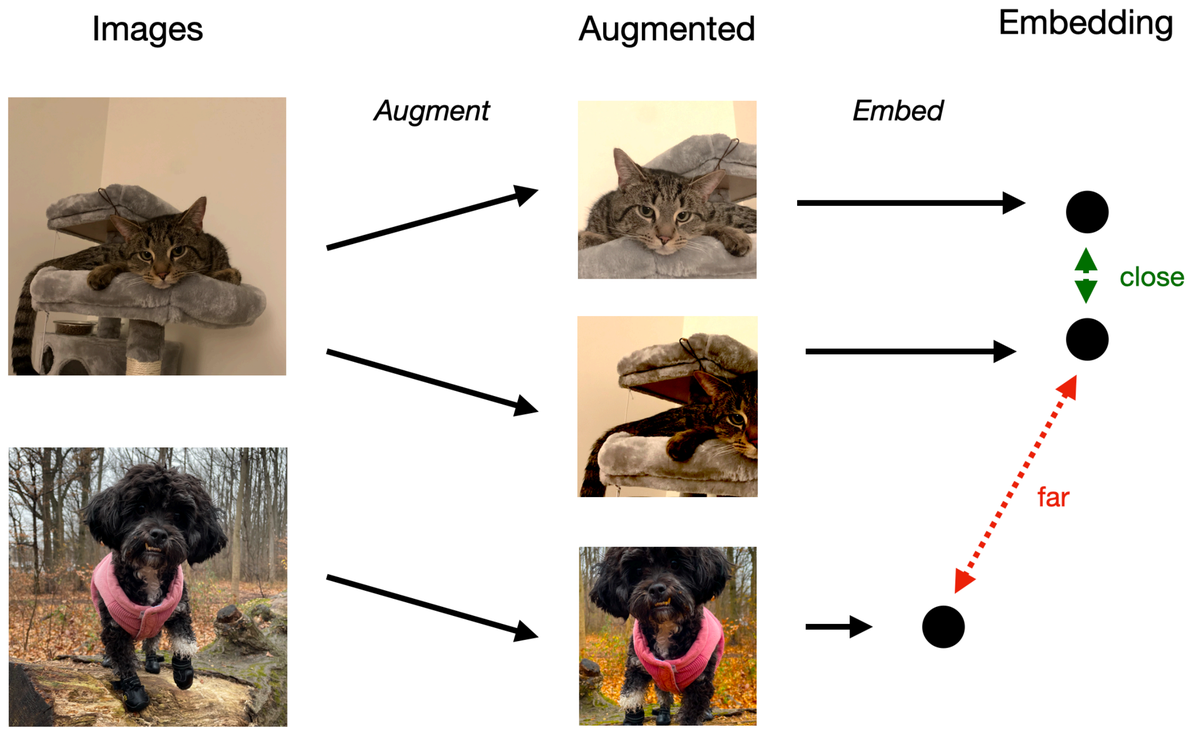


Figure source

Contrastive Learning Intuition

- Model learns aspects of the image that are **invariant** to noise
  - E.g. a blurred cat is still a cat
- Forces it to focus on **semantic aspects** of the image, rather than irrelevant details
  - This is built-in **inductive bias** for what the model should and shouldn't care about
- Similar images are **attracted** in vector space, while dissimilar ones are **repelled**

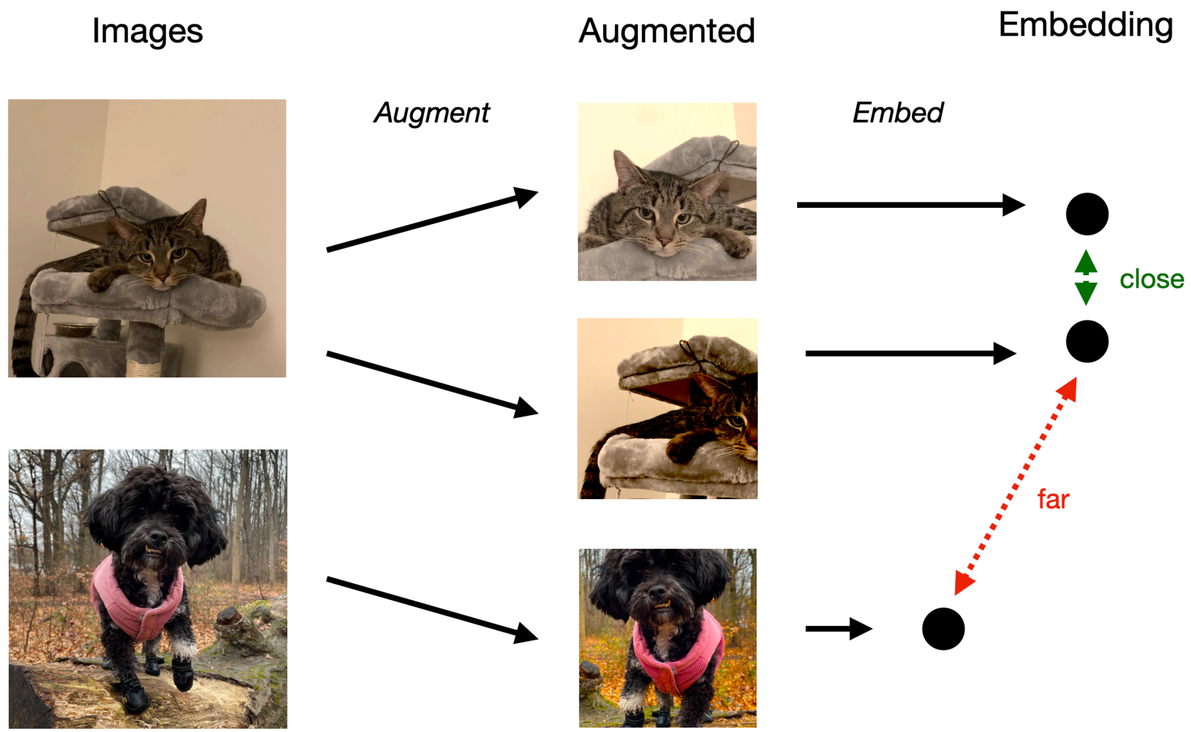


Figure source

Contrastive Loss

$$\mathscr{L} = -\log \frac{\exp(\text{sim}(anchor, positive)/\tau)}{\sum_i \exp(\text{sim}(anchor, candidate_i)/\tau)}$$

Contrastive Loss

- Many CL models use a loss term **similar to the one shown here**
  - In jargon, this is called **InfoNCE**
  - Very close to the general-purpose ML function called **(log) softmax**
  - Intuitively: maximize the similarity to the positive example **at the expense of the negative samples**

$$\mathscr{L} = -\log \frac{\exp(\text{sim}(anchor, positive)/\tau)}{\sum_i \exp(\text{sim}(anchor, candidate_i)/\tau)}$$

Contrastive Loss

- Many CL models use a loss term **similar to the one shown here**
  - In jargon, this is called **InfoNCE**
  - Very close to the general-purpose ML function called **(log) softmax**
  - Intuitively: maximize the similarity to the positive example **at the expense of the negative samples**

similarity of original and augmented image vectors

$$\mathscr{L} = -\log \frac{\exp(\text{sim}(anchor, positive)/\tau)}{\sum_i \exp(\text{sim}(anchor, candidate_i)/\tau)}$$

Contrastive Loss

- Many CL models use a loss term **similar to the one shown here**
  - In jargon, this is called **InfoNCE**
  - Very close to the general-purpose ML function called **(log) softmax**
  - Intuitively: maximize the similarity to the positive example **at the expense of the negative samples**

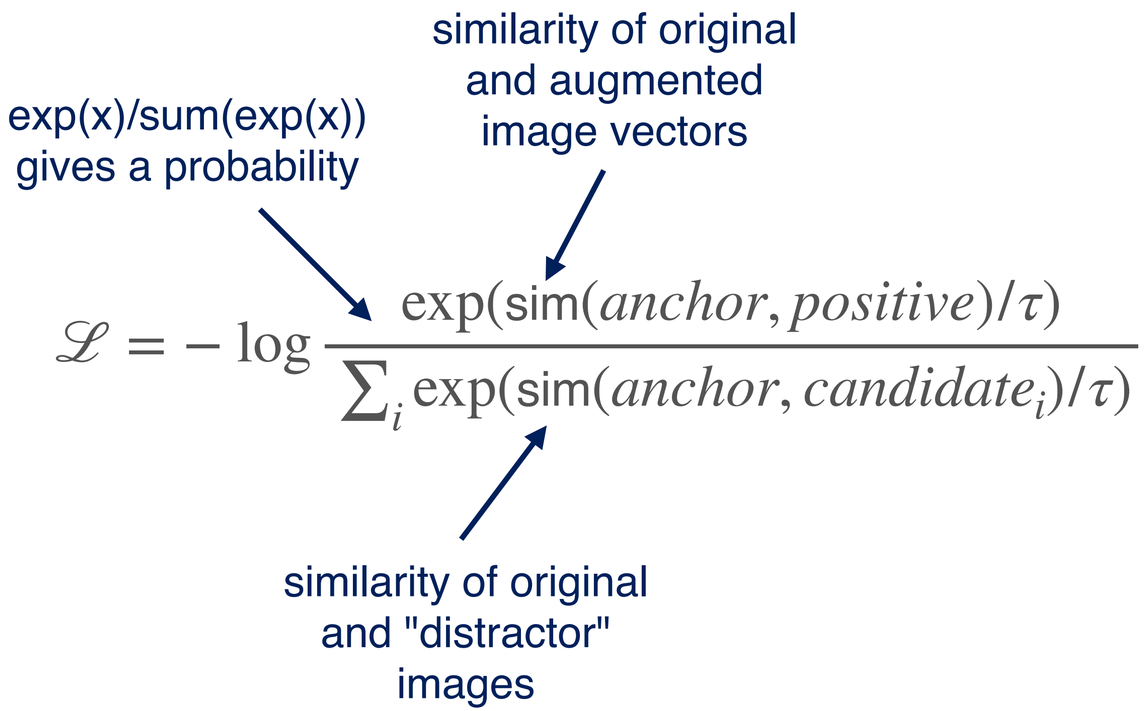
similarity of original and augmented image vectors

$$\mathscr{L} = -\log \frac{\exp(\text{sim}(anchor, positive)/\tau)}{\sum_i \exp(\text{sim}(anchor, candidate_i)/\tau)}$$

similarity of original and "distractor" images

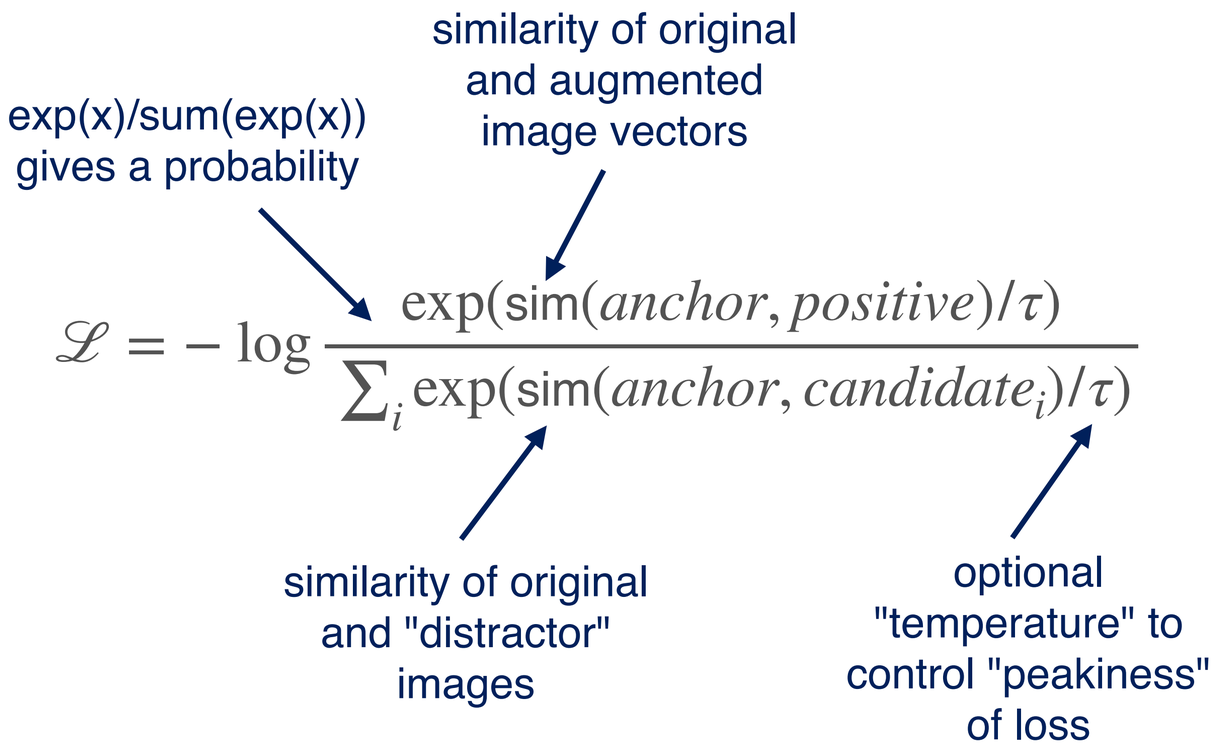
Contrastive Loss

- Many CL models use a loss term **similar to the one shown here**
  - In jargon, this is called **InfoNCE**
  - Very close to the general-purpose ML function called **(log) softmax**
  - Intuitively: maximize the similarity to the positive example **at the expense of the negative samples**



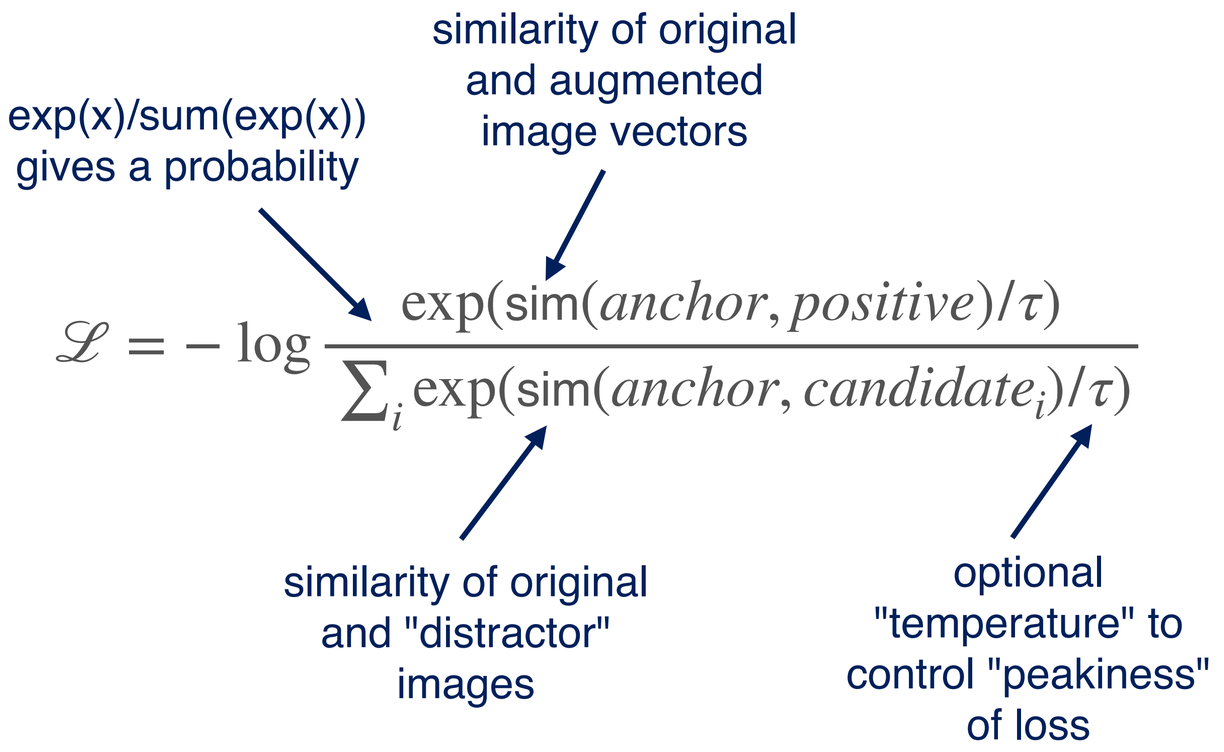
Contrastive Loss

- Many CL models use a loss term **similar to the one shown here**
  - In jargon, this is called **InfoNCE**
  - Very close to the general-purpose ML function called **(log) softmax**
  - Intuitively: maximize the similarity to the positive example **at the expense of the negative samples**



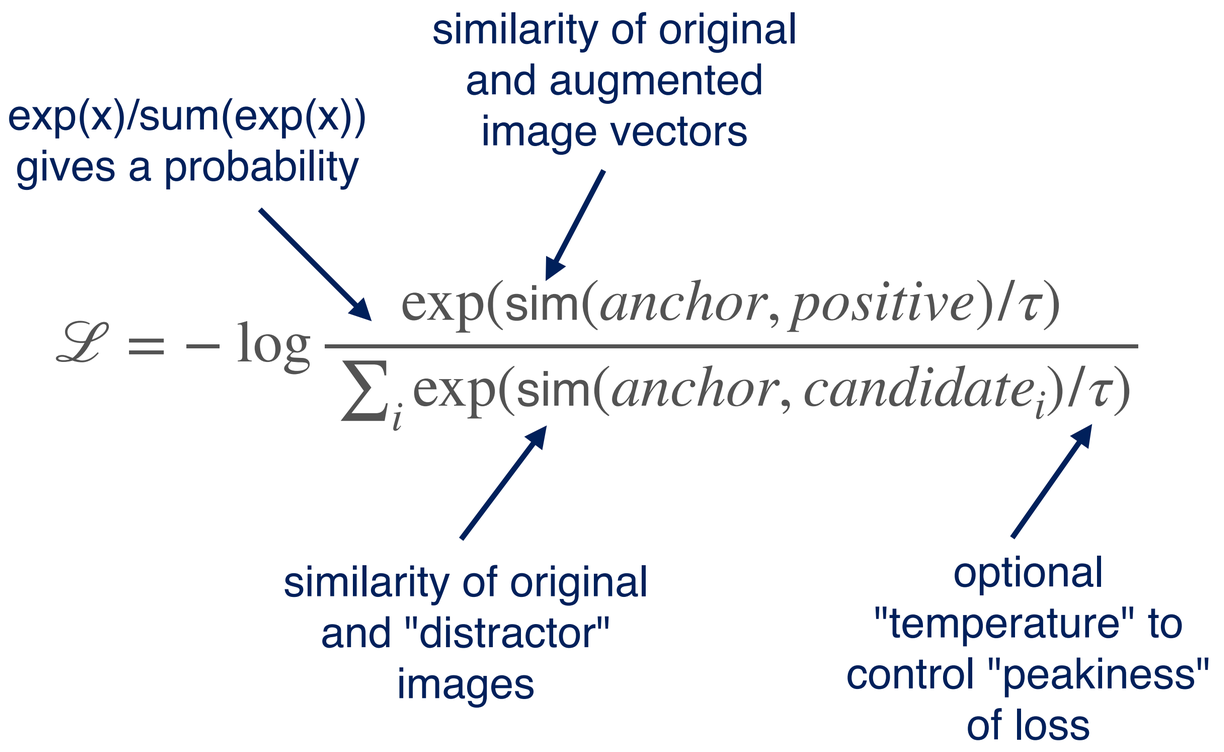
Contrastive Loss

- Many CL models use a loss term **similar to the one shown here**
  - In jargon, this is called **InfoNCE**
  - Very close to the general-purpose ML function called **(log) softmax**
  - Intuitively: maximize the similarity to the positive example **at the expense of the negative samples**
- This is **not** the only CL function out there, just a common one



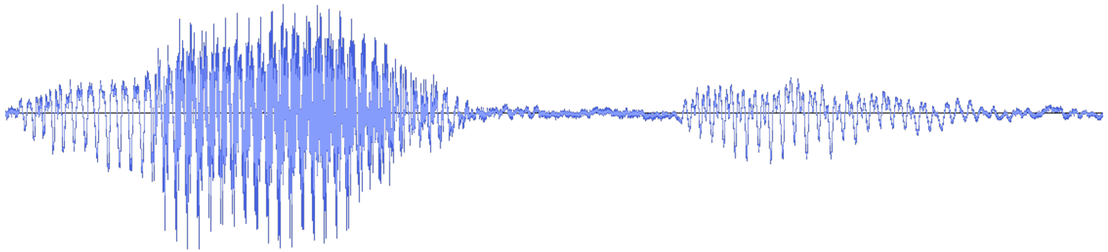
Contrastive Loss

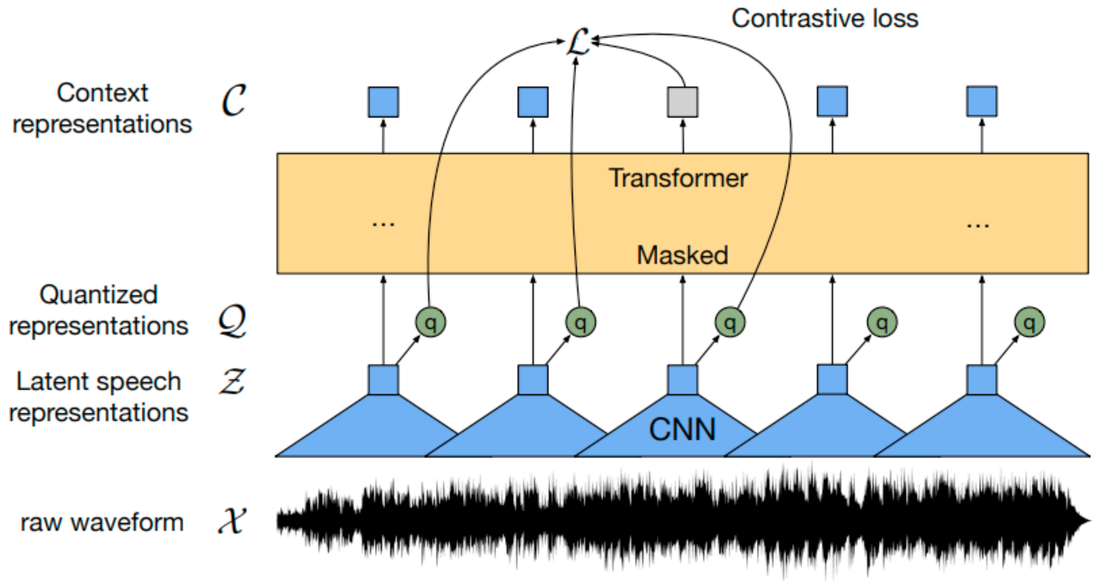
- Many CL models use a loss term **similar to the one shown here**
  - In jargon, this is called **InfoNCE**
  - Very close to the general-purpose ML function called **(log) softmax**
  - Intuitively: maximize the similarity to the positive example **at the expense of the negative samples**
- This is **not** the only CL function out there, just a common one
- Important: this is a **supervised classification formulation** (hence, self-supervised)



wav2vec: Self-supervised Speech Modeling

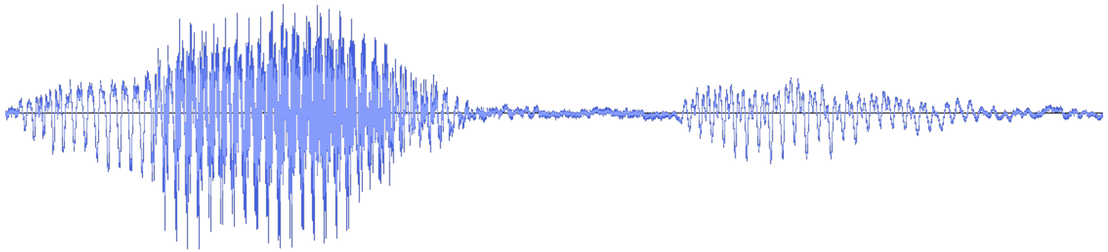
Speech Self-Supervision

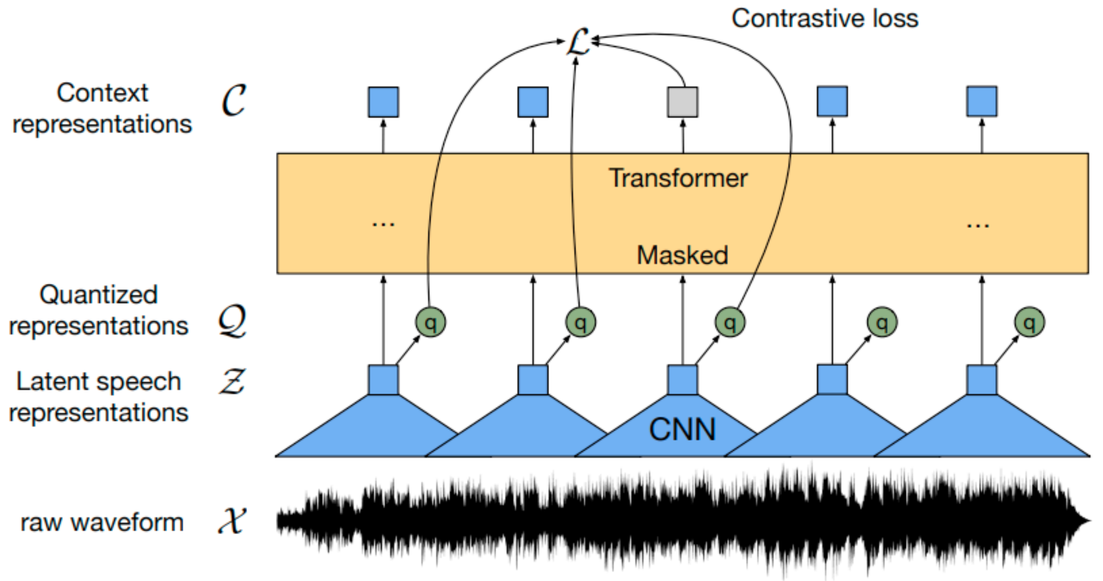




Speech Self-Supervision

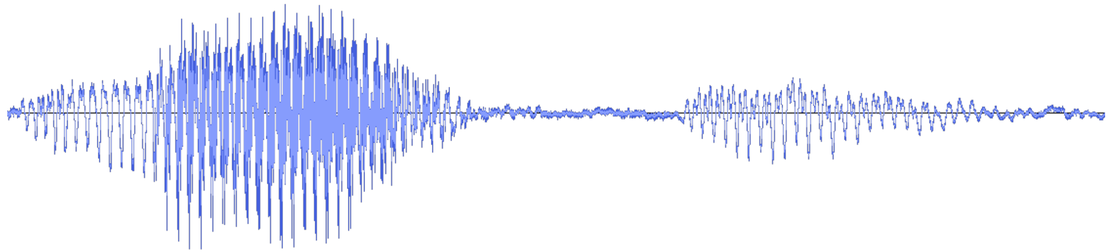
- How can we apply **self-supervision** to speech models?

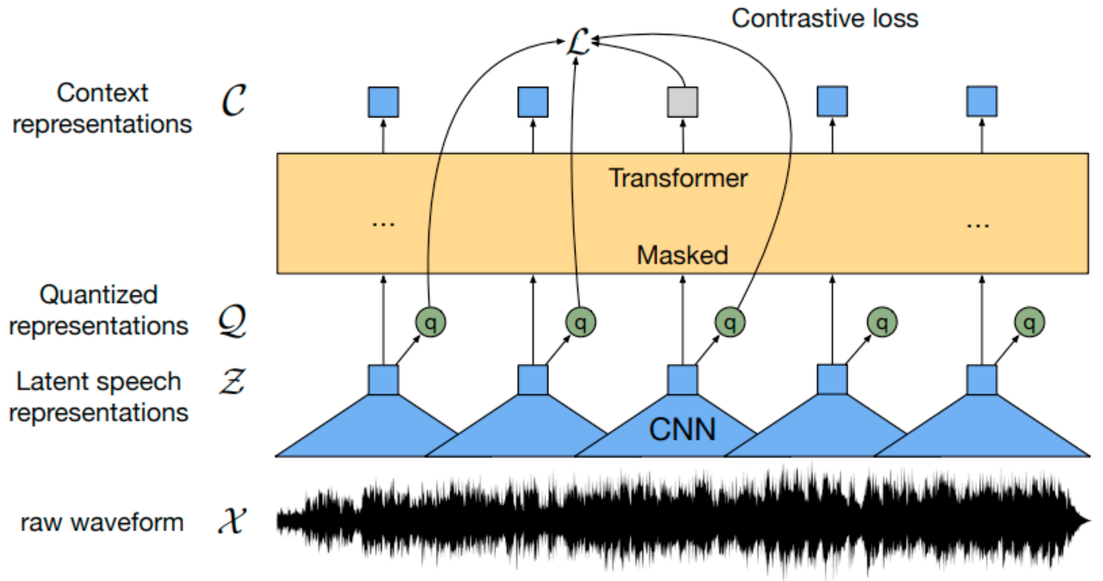




Speech Self-Supervision

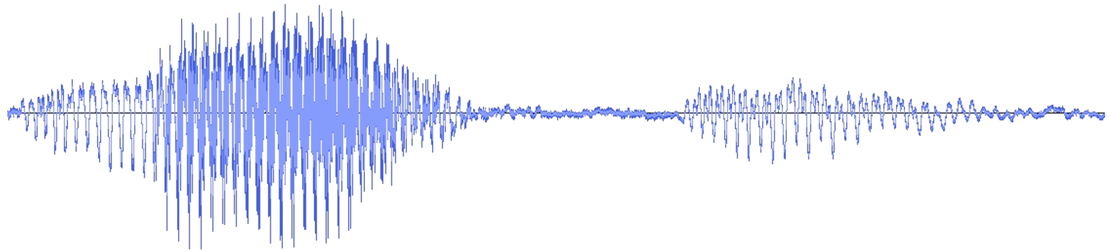
- How can we apply **self-supervision** to speech models?
- Transcribed audio (needed for ASR training) is **rare**

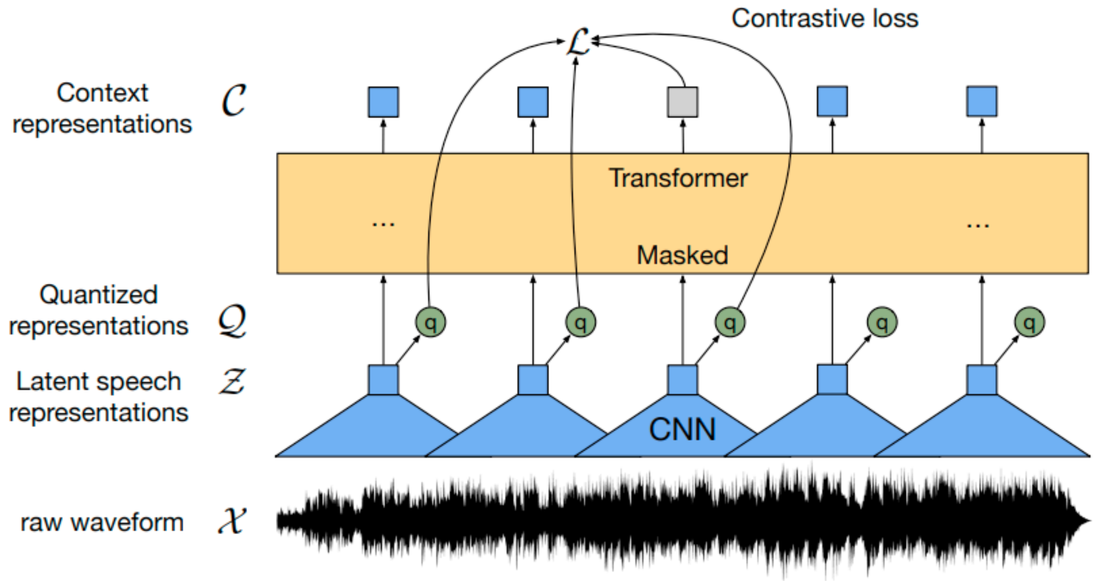




Speech Self-Supervision

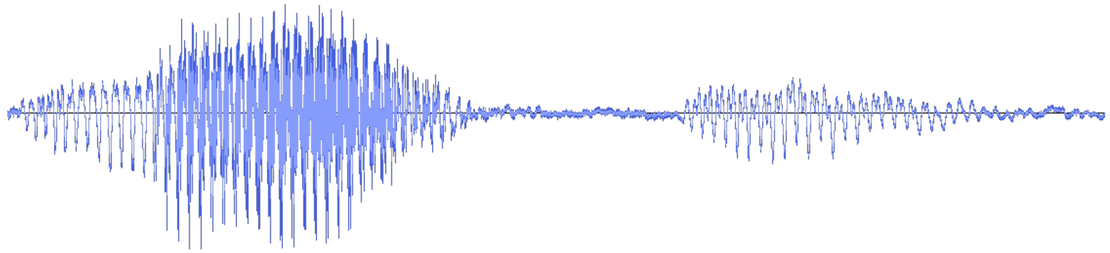
- How can we apply **self-supervision** to speech models?
- Transcribed audio (needed for ASR training) is **rare**
- How can we pre-train on **audio only?**
  - Models like **wav2vec** address this

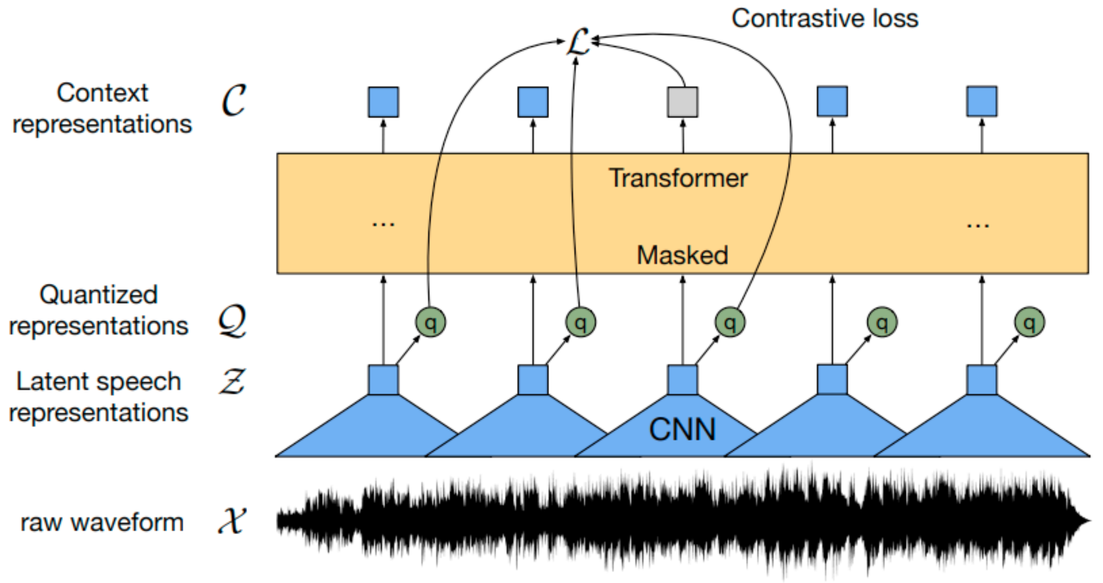




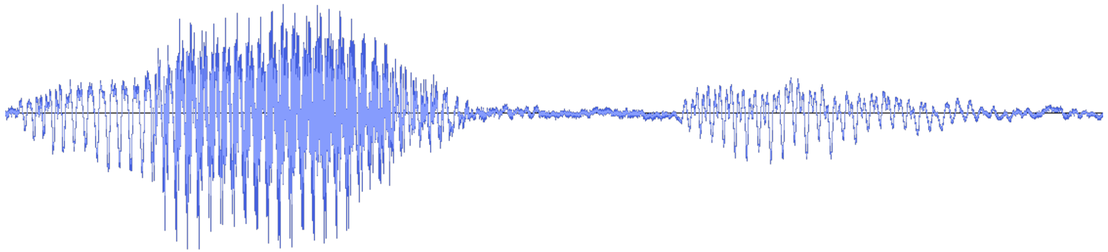
Speech Self-Supervision

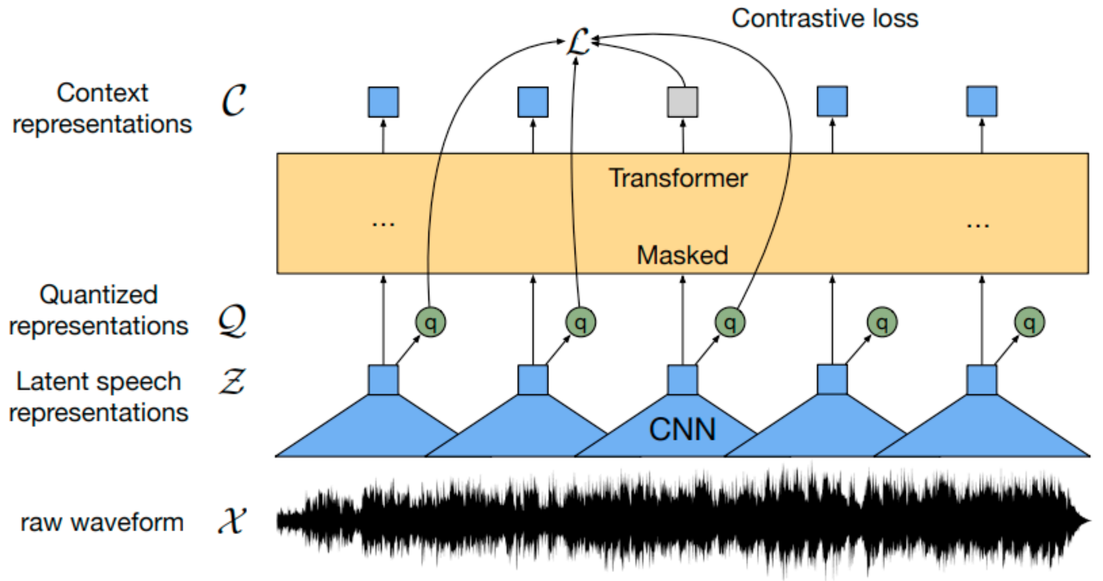
- How can we apply **self-supervision** to speech models?
- Transcribed audio (needed for ASR training) is **rare**
- How can we pre-train on **audio only?**
  - Models like **wav2vec** address this
- Visualizations on next slides from [this helpful blog post]





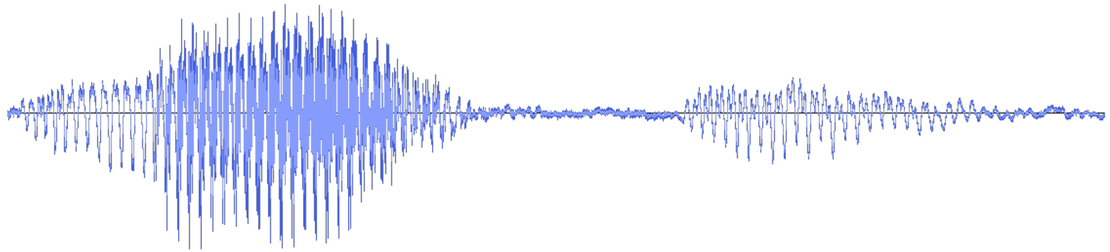
Speech Self-Supervision

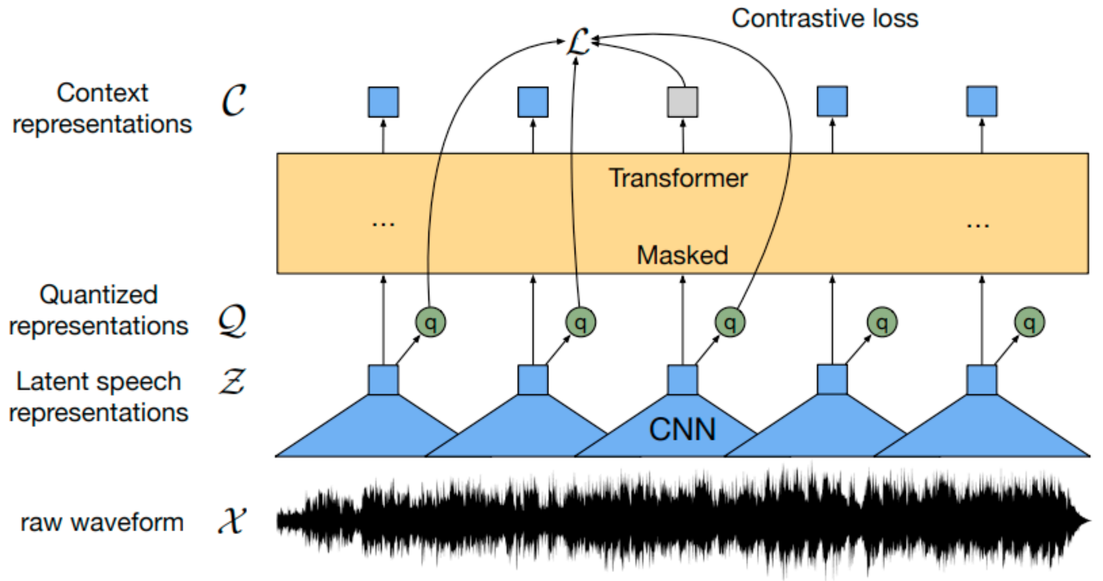




Speech Self-Supervision

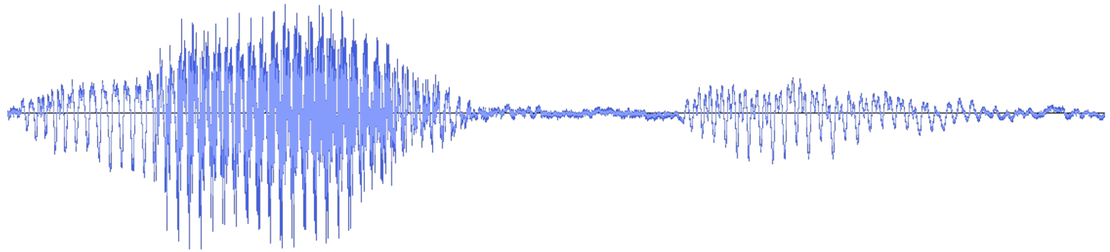
- Audio data is a **continuous waveform** (amplitude across time)
  - What would self-supervision look like? **What should the model predict?**

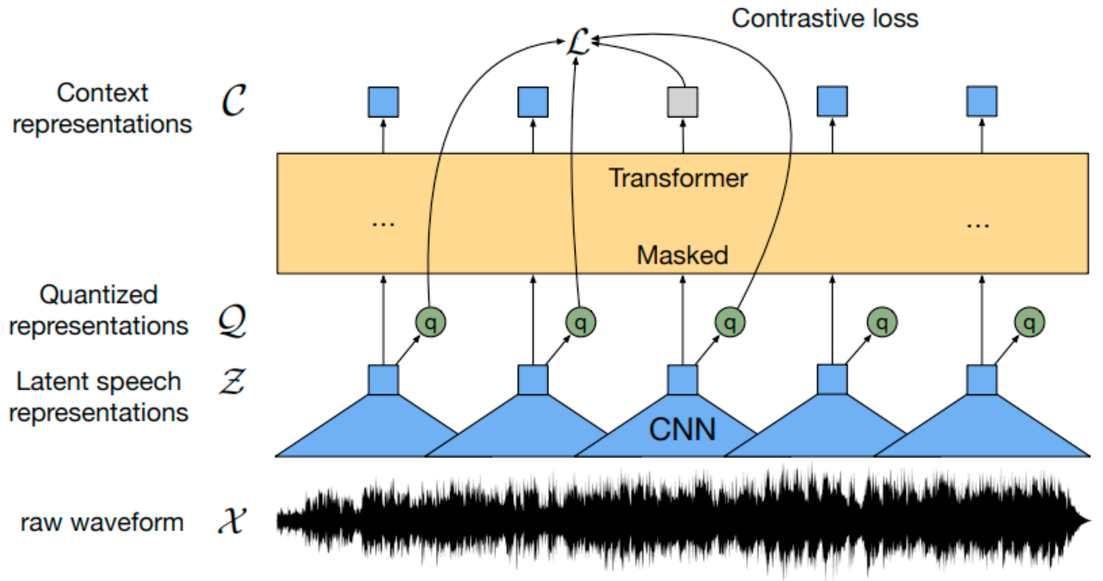




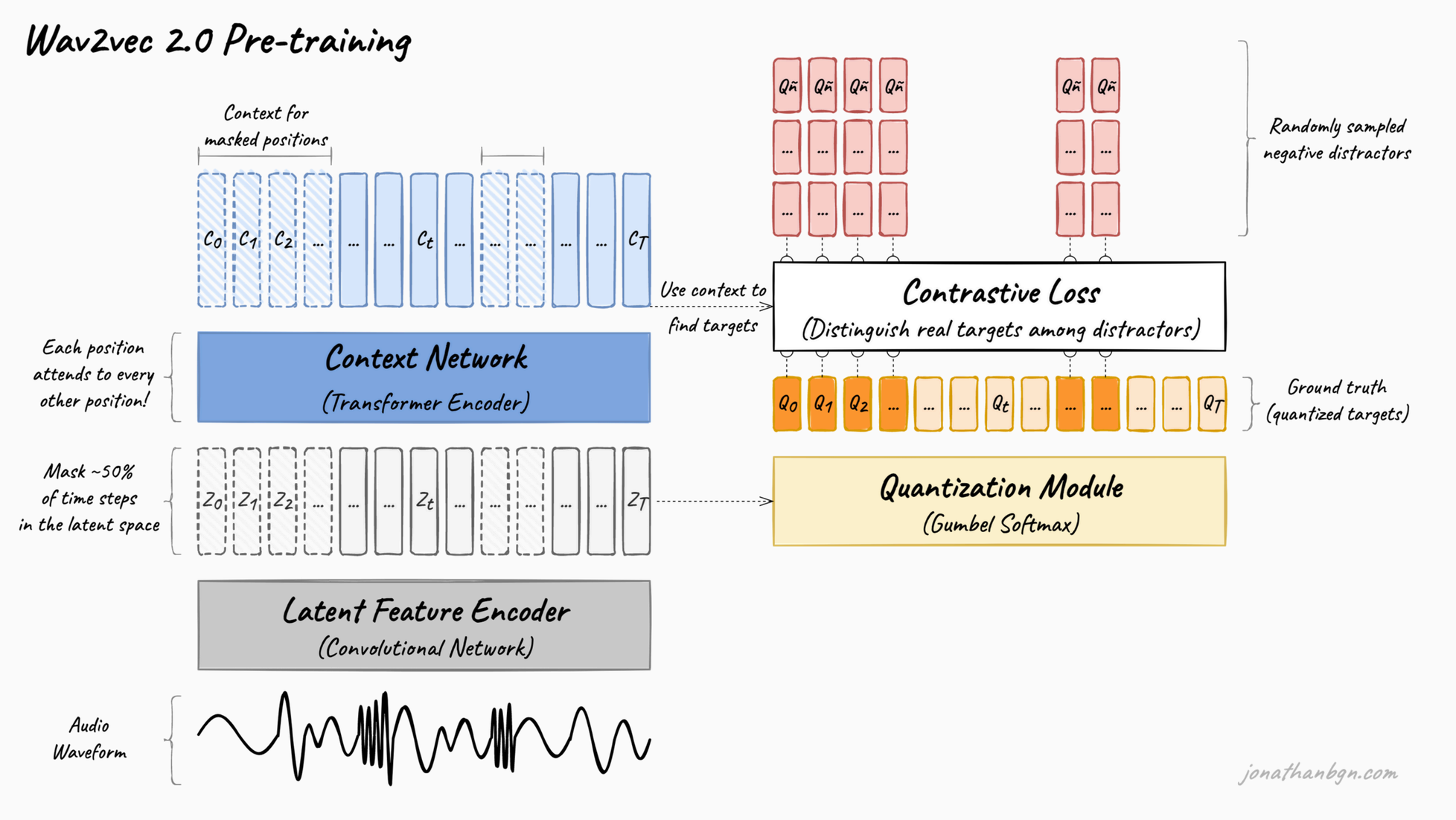
Speech Self-Supervision

- Audio data is a **continuous waveform** (amplitude across time)
  - What would self-supervision look like? **What should the model predict?**
- Solution in wav2vec 2.0: map waveform to **quantized** (discrete) representations, then **predict those**
  - Model has to learn **both** the quantized representations and how to predict them
  - Challenge: there is a **degenerate solution** available to the model. What is it, and how might you solve it? (wav2vec 2.0 implements a solution)



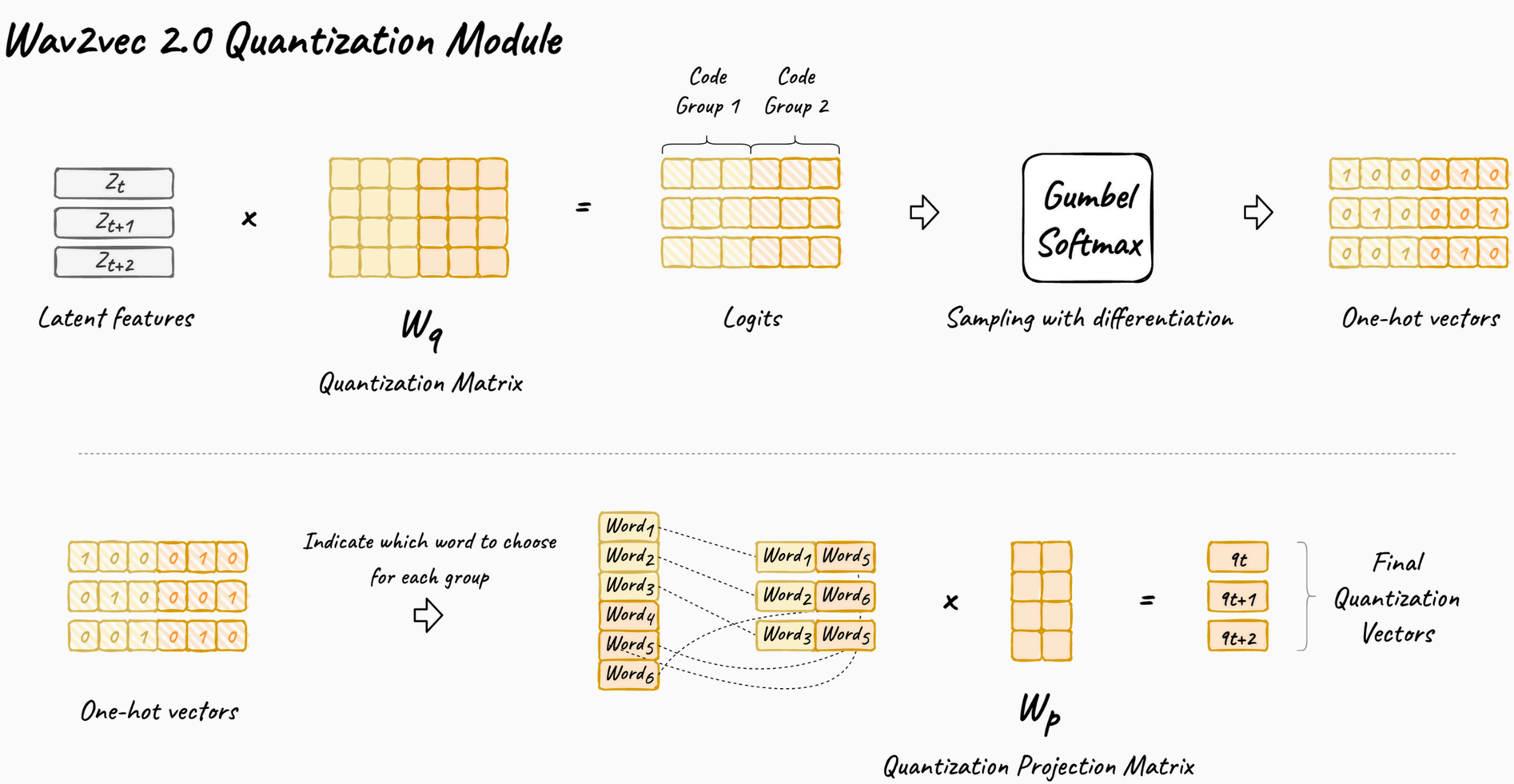


wav2vec (2.0) Overview

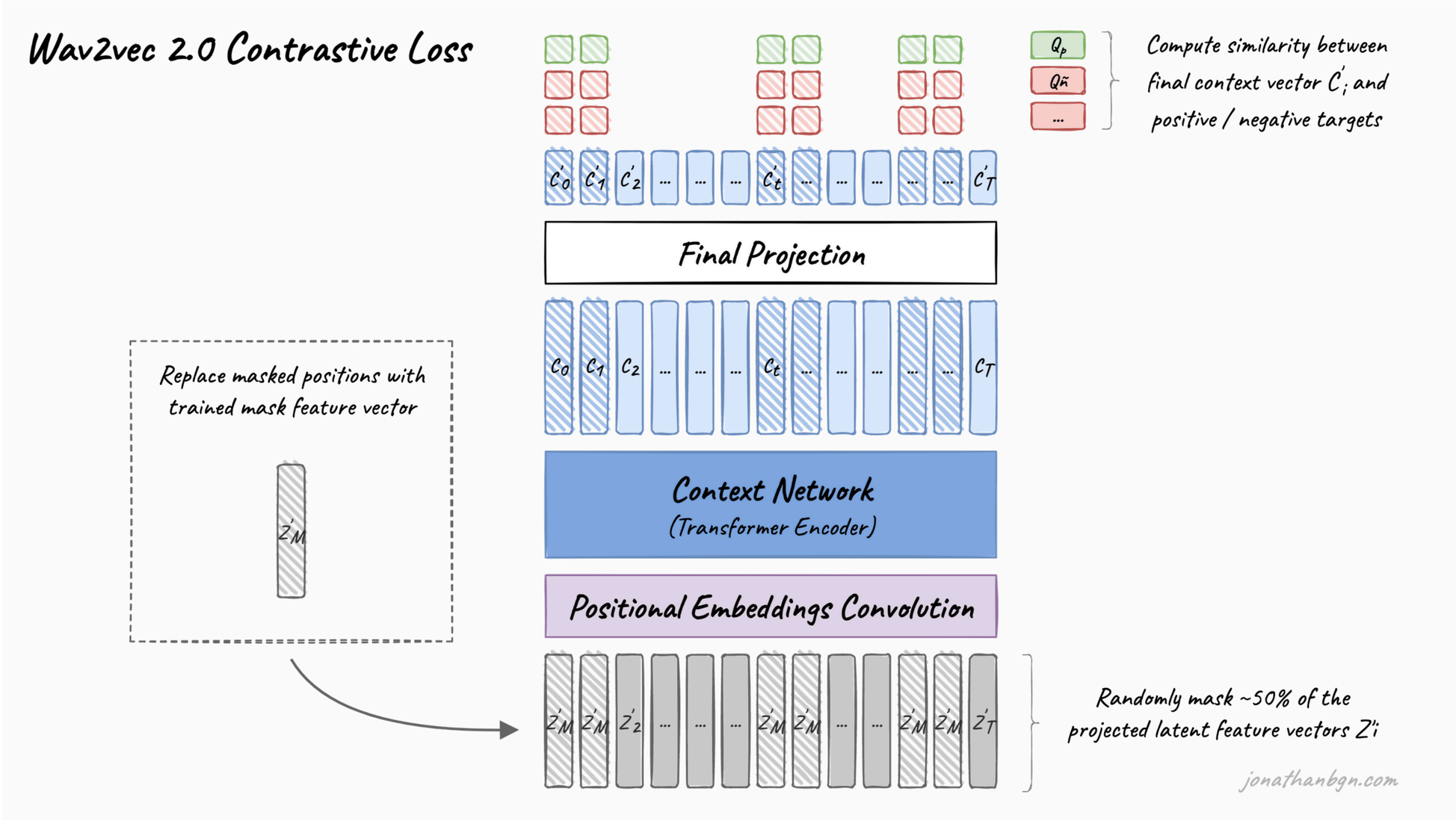


jonathanbgn.com

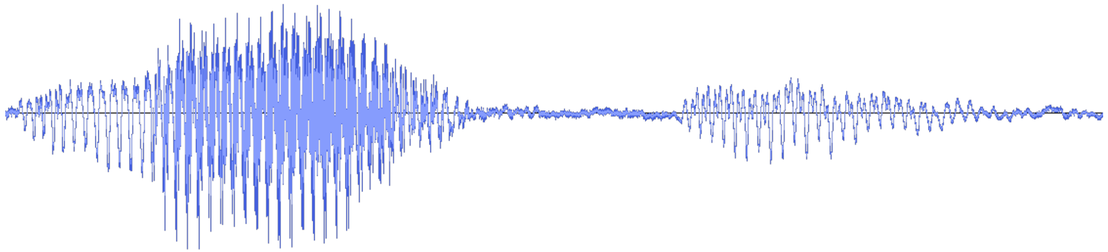
wav2vec 2.0 Quantization

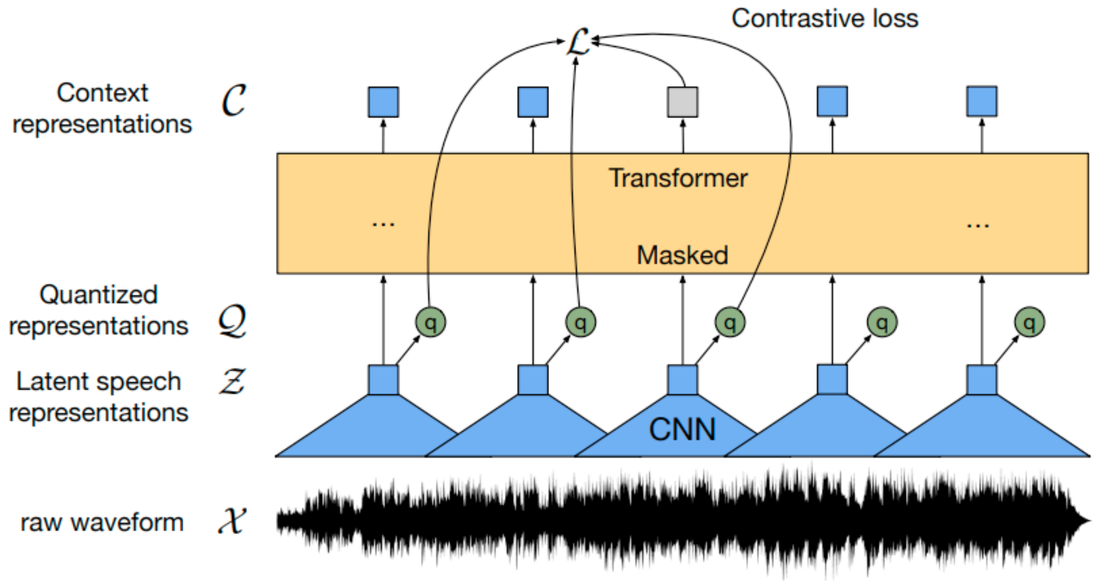


wav2vec 2.0 Contrastive Loss



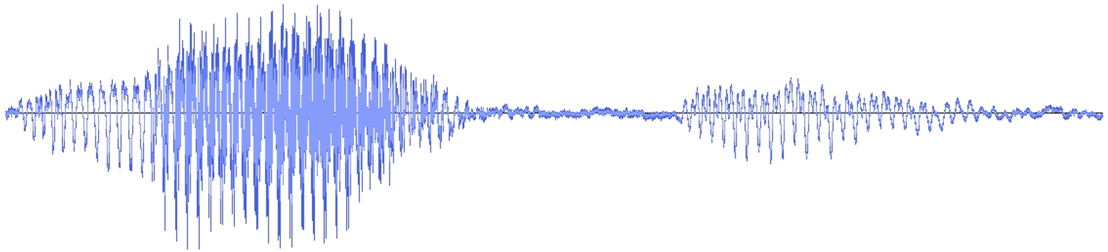
wav2vec Results

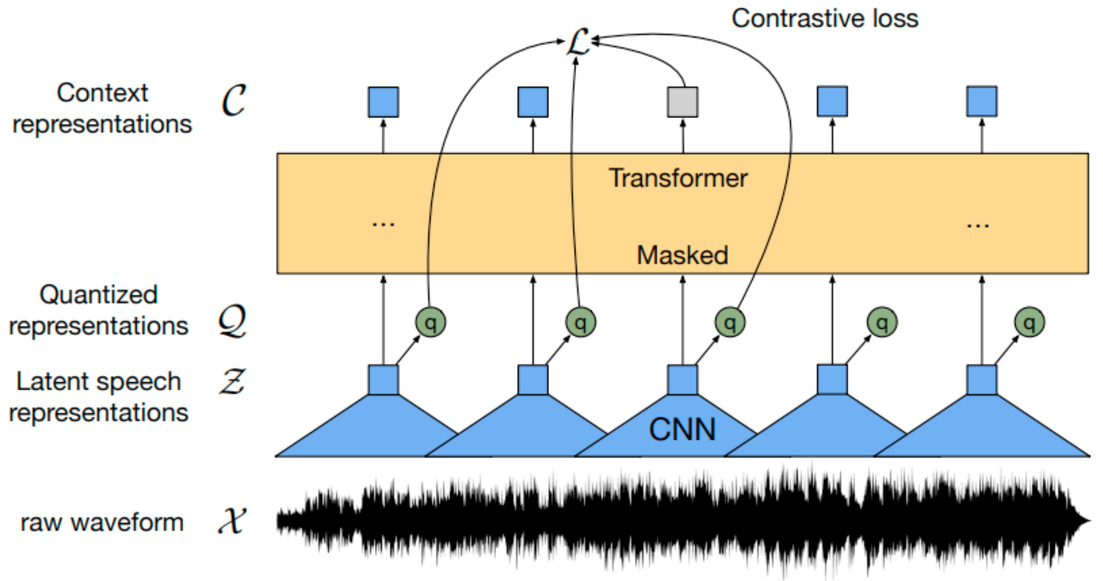




wav2vec Results

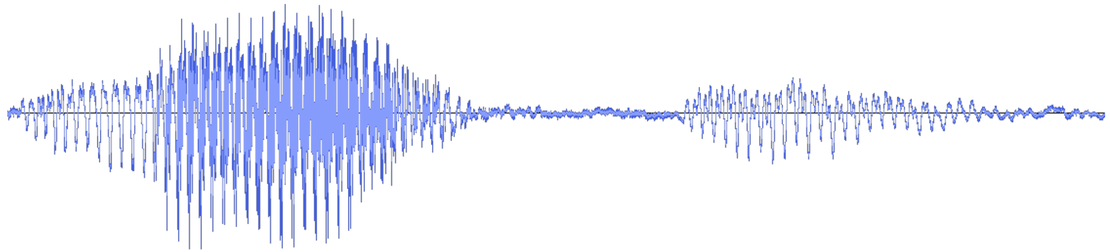
- Able to pre-train on **large amounts of raw audio**

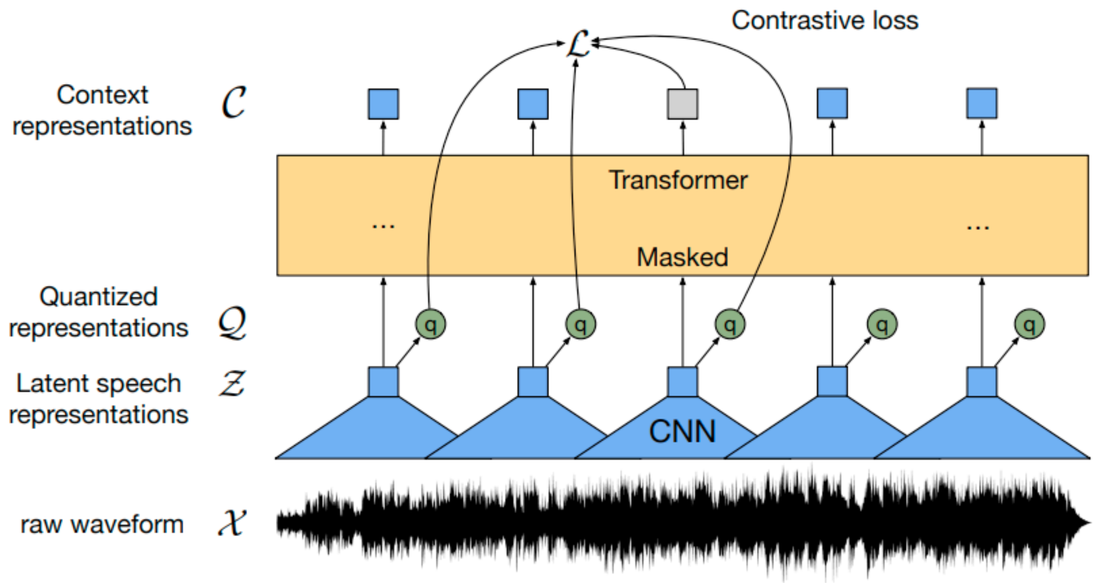




wav2vec Results

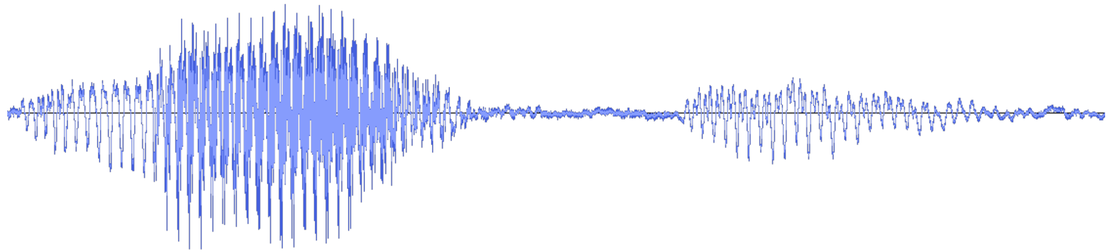
- Able to pre-train on **large amounts of raw audio**
- Then can **fine-tune** on the audio-to-text mapping (ASR) with **small amounts of transcribed speech** (e.g. 10mins-10hrs)

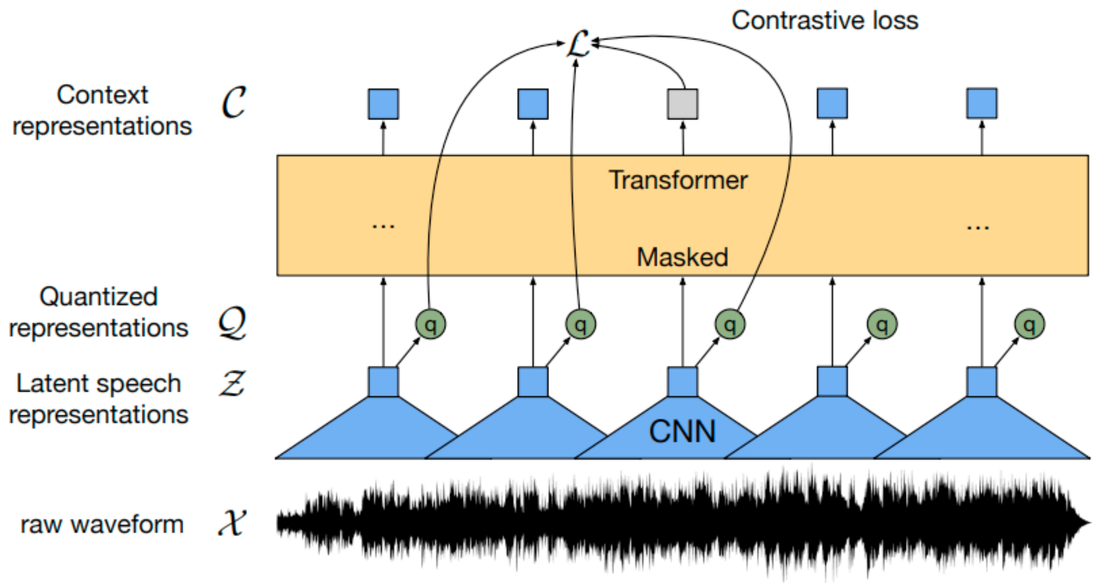




wav2vec Results

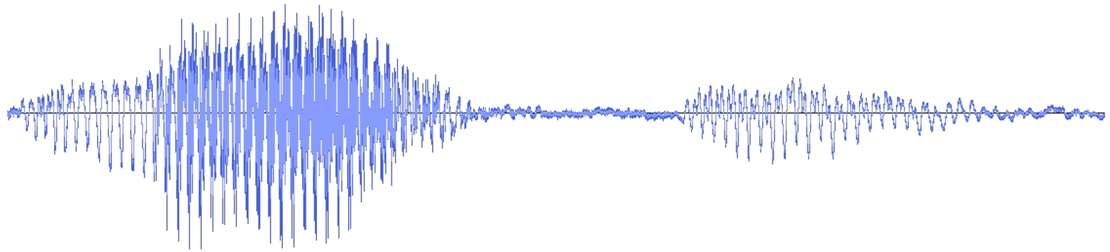
- Able to pre-train on **large amounts of raw audio**
- Then can **fine-tune** on the audio-to-text mapping (ASR) with **small amounts of transcribed speech** (e.g. 10mins-10hrs)
- Competitive with **fully-supervised** models for high-resource languages

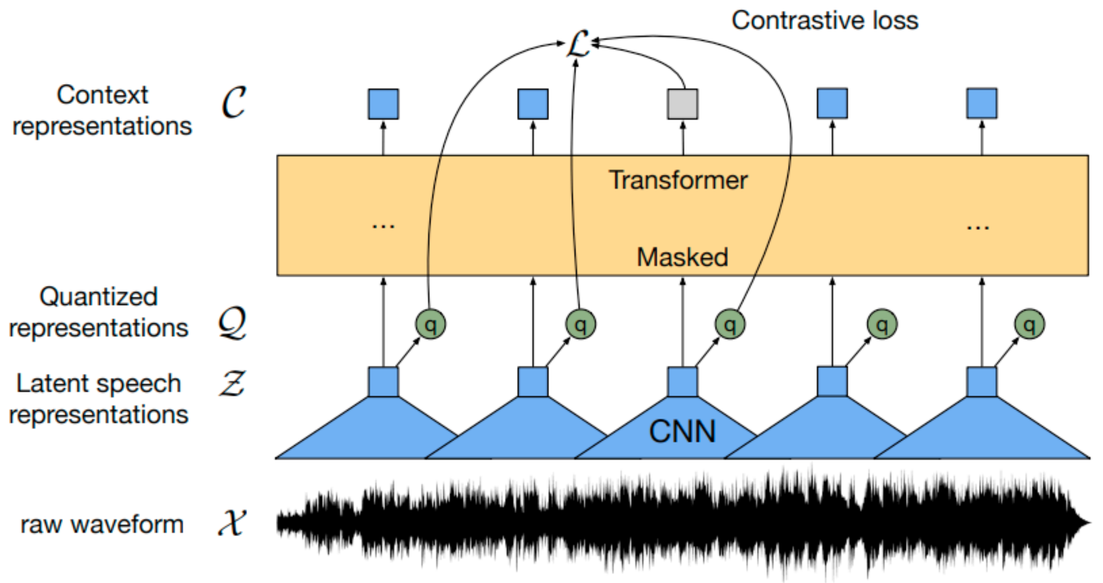




wav2vec Results

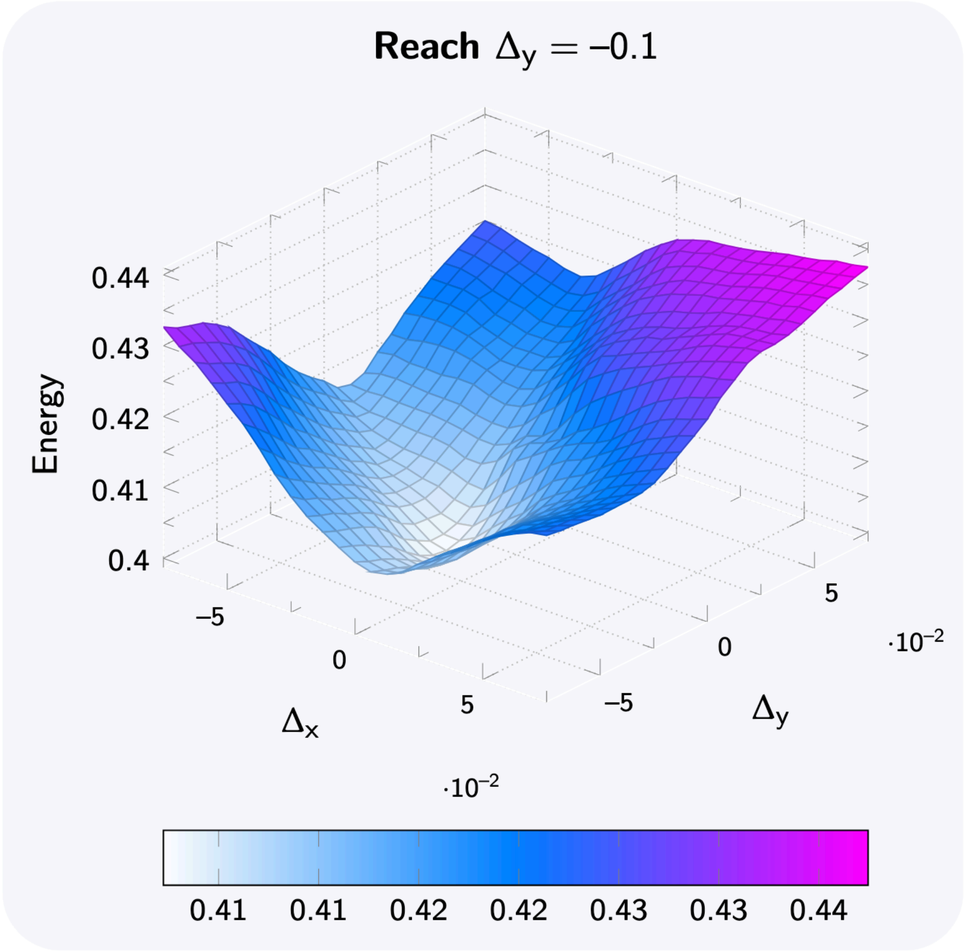
- Able to pre-train on **large amounts of raw audio**
- Then can **fine-tune** on the audio-to-text mapping (ASR) with **small amounts of transcribed speech** (e.g. 10mins-10hrs)
- Competitive with **fully-supervised** models for high-resource languages
- Pretty much the **only option** for low-resource languages
  - Not wav2vec specifically, but SSL models





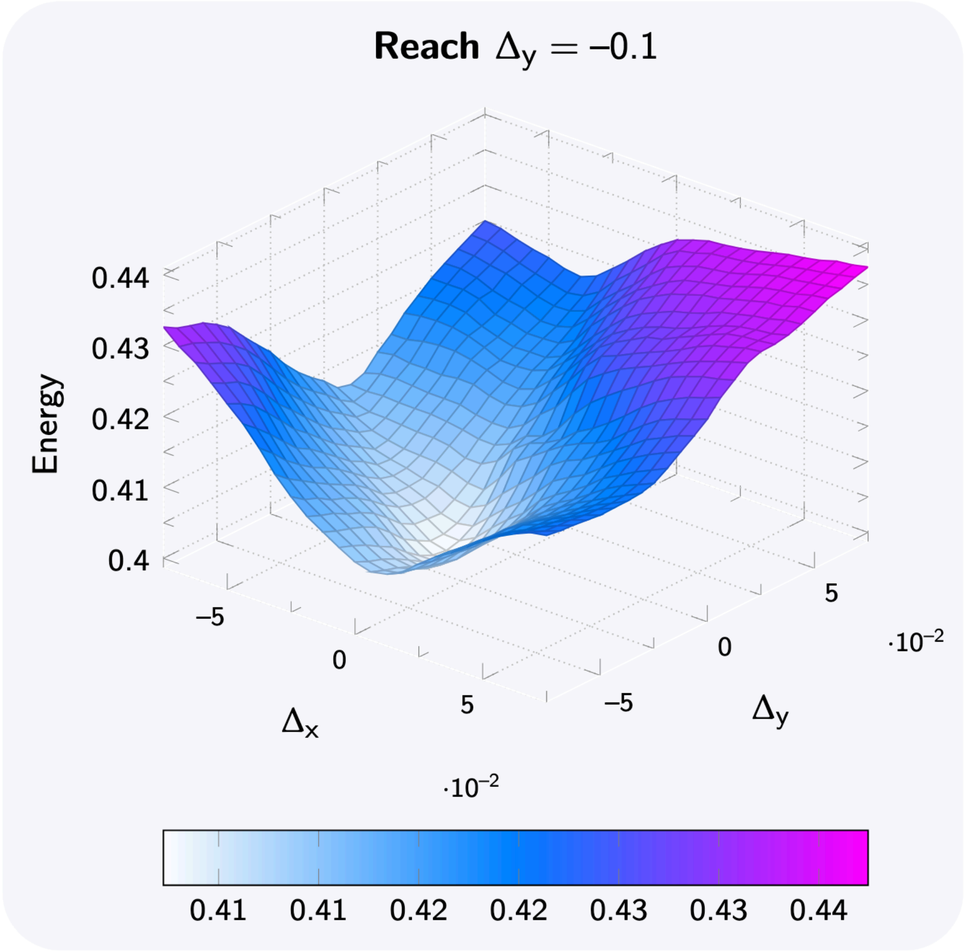
Bonus: Beyond Discrete Prediction

"Energy-Based Models"



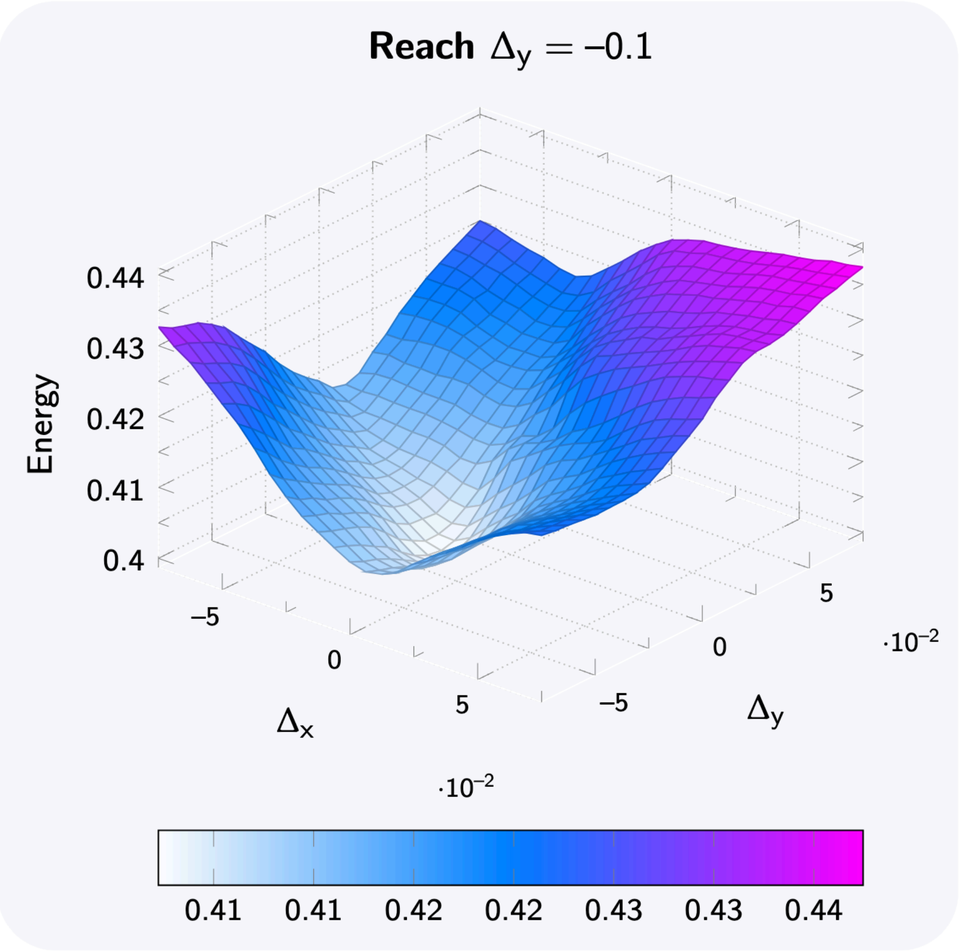
"Energy-Based Models"

- Many SSL methods predict in a **discrete output space**
  - e.g. next word, next video frame

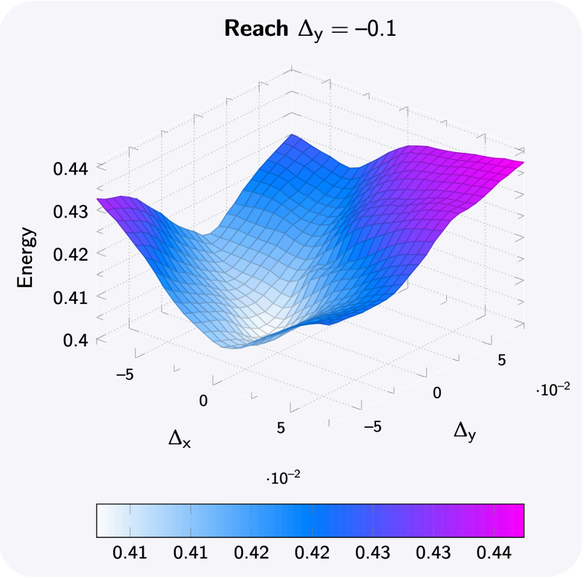


"Energy-Based Models"

- Many SSL methods predict in a **discrete output space**
  - e.g. next word, next video frame
- Yann LeCun (among others) argues we should only care about the **latent space** (vector space)
  - Intuition: predicting an entire next video frame is wasteful. Just make sure that the **embeddings of adjacent frames are "compatible"**

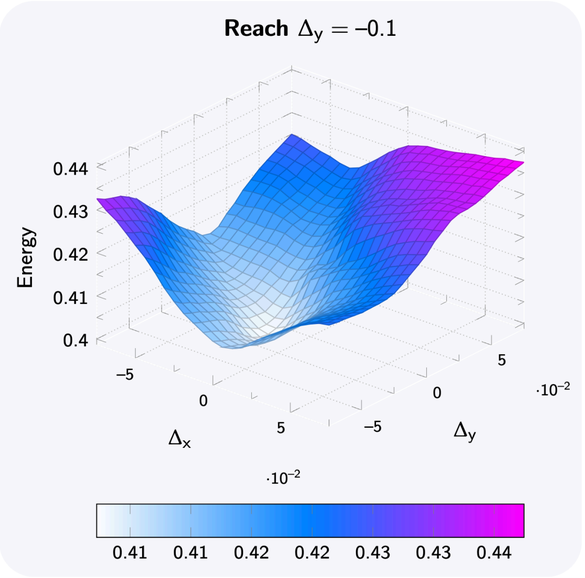


"Energy-Based Models"



"Energy-Based Models"

- In latent space, compatibility is measured by **low "energy"** between embeddings
  - Goal: minimize the energy, don't predict actual outputs



"Energy-Based Models"

- In latent space, compatibility is measured by **low "energy"** between embeddings
  - Goal: minimize the energy, don't predict actual outputs
- Challenge: **vector collapse**
  - Model might learn to **map all frames** to the **same embedding**
  - **Contrastive learning** is one way to prevent collapse
  - Active area of making EBMs practical by avoiding collapse (and without negatives)