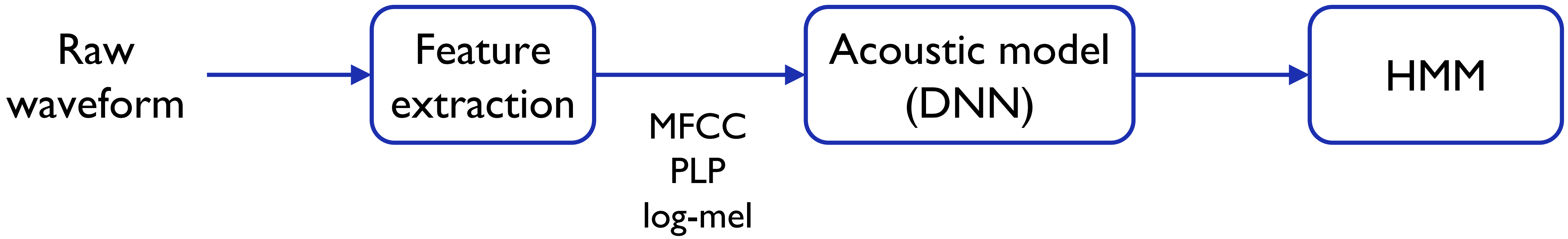
Unsupervised Raw Waveform Modelling: Self-supervised learning for Speech

Yumnah Mohamied

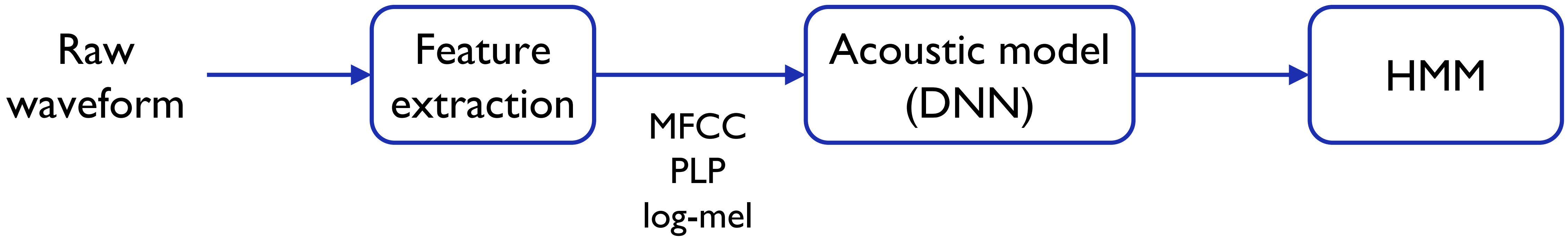
Automatic speech recognition – ASR lecture 18
23 March 2023

Divide and Conquer Strategy



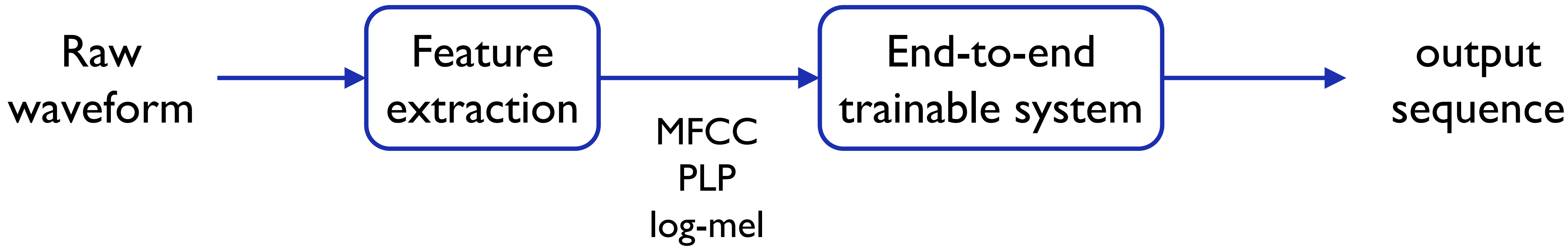
- Conventional ASR consists of composite subsystems trained and designed independently.
- Separates out feature extraction, acoustic modelling and decoding steps.

Divide and Conquer Strategy



- Conventional ASR consists of composite subsystems trained and designed independently.
- Separates out feature extraction, acoustic modelling and decoding steps.
- Feature extraction is hand-crafted – based on prior knowledge of speech production and/or perception.

End-to-end systems



- End-to-end systems directly map the extracted features to an output sequence (words).
- But we can extend end-to-end concept in the other direction: learnable feature extractor

Feature learning from the raw waveform

Raw waveform → Feature learning (within acoustic model/encoder) + downstream ASR model → output sequence

- Divide and conquer strategy was overwhelmingly outperformed by feature learning in image processing.
- The deep learning revolution: ability to train with raw signal with improved performance - no longer need to handcraft features.

Feature learning from the raw waveform

Raw waveform → Feature learning (within acoustic model/encoder) + downstream ASR model → output sequence

- HMM/GMM: sensitive to input features
  - Needs to be decorrelated to use a diagonal covariance matrix
  - Dimension needs to be low
- Expert knowledge of speech production/perception led to range of feature extraction pipelines: MFCC, log-mel, PLP, gammatone …
- Hybrid HMM/DNN don't have these constraints.
- Features designed from perceptual evidence is not guaranteed to be best features in a statistical modelling framework.
- Information loss from raw signal: models trained with a combination of hand-crafted features outperform those trained with a single feature type.

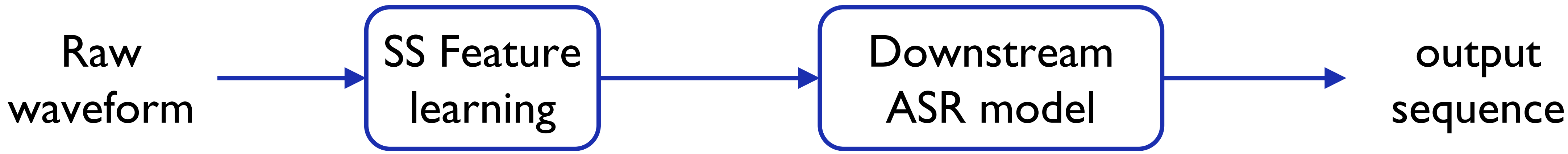
Supervised feature learning

- Feature learning part of the acoustic model: input is raw waveform.
- Can use DNN
  - But high-resolution and temporal aspect of raw waveform makes CNNs a better choice (reduces learnable parameters).
  - Then add a fully connected layer + softmax for classification and output probabilities.

Supervised feature learning

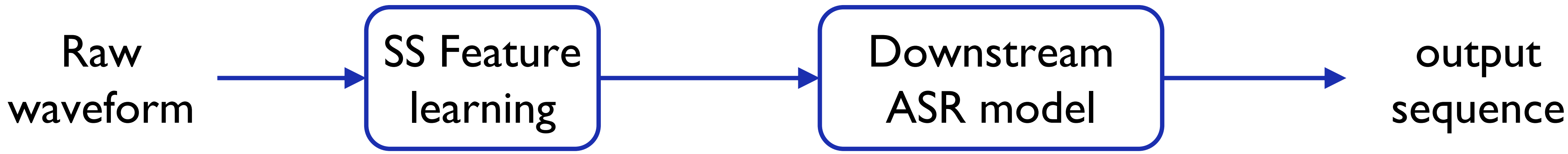
- Feature learning part of the acoustic model: input is raw waveform.
- Can use DNN
  - But high-resolution and temporal aspect of raw waveform makes CNNs a better choice (reduces learnable parameters).
  - Then add a fully connected layer + softmax for classification and output probabilities.
- Can use LSTM directly with raw waveform for temporal modelling
  - But higher-level modelling of the input features helps to disentangle underlying factors of variation within the input.
  - Requires unrolling LSTM for an infeasibly large number of steps
  - Precede with CNN layers.
- Combine CNN layers, LSTM and DNN layers and train altogether: CLDNN
- Performance comes close to hand-crafted features

Self-supervised learning (SSL)



- Feature learning step is separate to the acoustic model or end-to-end system – therefore no labels
- Goal: learn a representation from the raw waveform that is then frozen after training, and input into an ASR system as a replacement to handcrafted features.
- Leverage large amounts of unlabelled data to learn a general representation – features are not task specific.

Self-supervised learning (SSL)



- Feature learning step is separate to the acoustic model or end-to-end system – therefore no labels
- Goal: learn a representation from the raw waveform that is then frozen after training, and input into an ASR system as a replacement to handcrafted features.
- Leverage large amounts of unlabelled data to learn a general representation – features are not task specific.

Approaches we will discuss

SSL learning algorithm:

Pretext task:	Contrastive methods (CPC)	Deep clustering	Student-teacher methods (BYOL)
Masked acoustic modelling	wav2vec 2.0	HuBERT	Data2vec
Auto-regressive	wav2vec VQ-wav2vec		BPC

Contrastive methods

CPC
wav2vec
VQ-wav2vec
Wav2vec 2.0

Contrastive Predictive Coding

- Intuition: learn representations that encode the underlying shared information between different parts of the high-dimensional speech signal
    - Maximise the Mutual Information

- CPC loss objective operates in latent space: it is challenging to predict (i.e. generate) high-dimensional data.
    - Unimodal losses (MSE) are not adept (introduces too much blurring)
    - Powerful generative models that reconstruct every detail would be required: computational intense and waste capacity at modelling complex relationships in the data.

CPC in context of autoregressive modelling

- Autoregressive pretext task: learn to predict observations in the future, *x*, from an encoded context window in the present, *c*.
  - Future observations, *x*, are the "labels" created from the data
- Modelling $p(x|c)$ (a generative model) to predict *x*, may not be optimal for extracting shared information between *x* and *c*.
- We encode *x* and *c*, into compact representations which maximally preserve MI of the original signals - we extract underlying latent variables that *x* and *c* have in common
- Loss operates on these latent variables of *x* and *c*

CPC: Maximising Mutual Information

- MI given by:

$$I(x;c) = \sum_{x,c} p(x,c) \log \frac{p(x|c)}{p(x)}.$$

- Model a density ratio, *f*, that preserves MI (use a simple log-bilinear model):

$$f_k(x_{t+k}, c_t) \propto \frac{p(x_{t+k}|c_t)}{p(x_{t+k})} \qquad f_k(x_{t+k}, c_t) = \exp\left(z_{t+k}^T W_k c_t\right),$$

- Using a density ratio, and inferring z with an encoder, means the model does not need to model the high-dimensional x.

CPC: InfoNCE (noise contrastive loss)

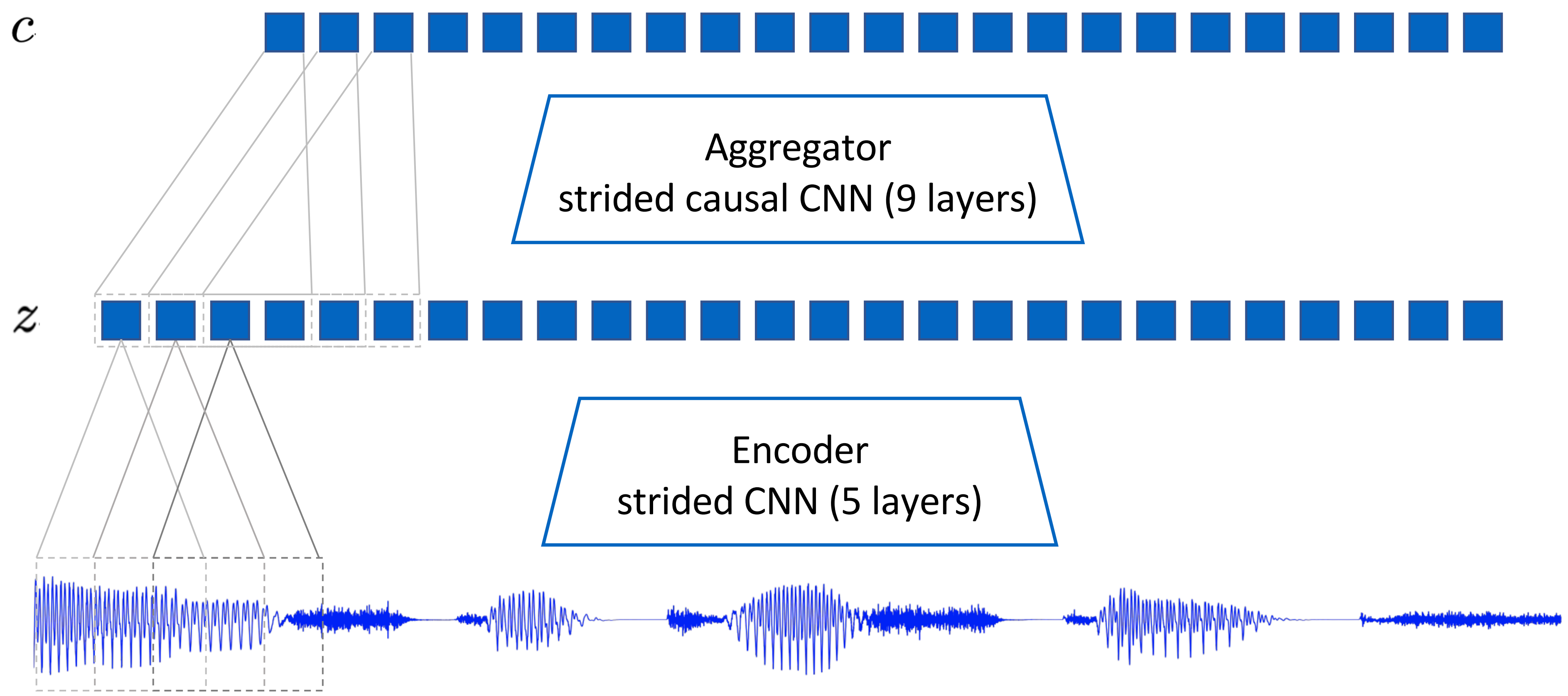
- We cannot evaluate *p(x)* or *p(x|c)* directly, but we can sample from these distributions
- One positive sample from *p(x|c)*, and N negative samples from the proposal distribution *p(x)* (random frame encodings within and across utterances)

$$\mathcal{L} = -\mathop{\mathbb{E}}_{X}\left[\log \frac{f_k(x_{t+k}, c_t)}{\sum_{x_j \in X} f_k(x_j, c_t)}\right] \qquad f_k(x_{t+k}, c_t) = \exp\left(z_{t+k}^T W_k c_t\right),$$

$$\mathcal{L}_k = -\sum_{i=1}^{T-k}\left(\log \sigma(\mathbf{z}_{i+k}^\top h_k(\mathbf{c}_i)) + \lambda \mathop{\mathbb{E}}_{\tilde{\mathbf{z}} \sim p_n}\left[\log \sigma(-\tilde{\mathbf{z}}^\top h_k(\mathbf{c}_i))\right]\right)$$

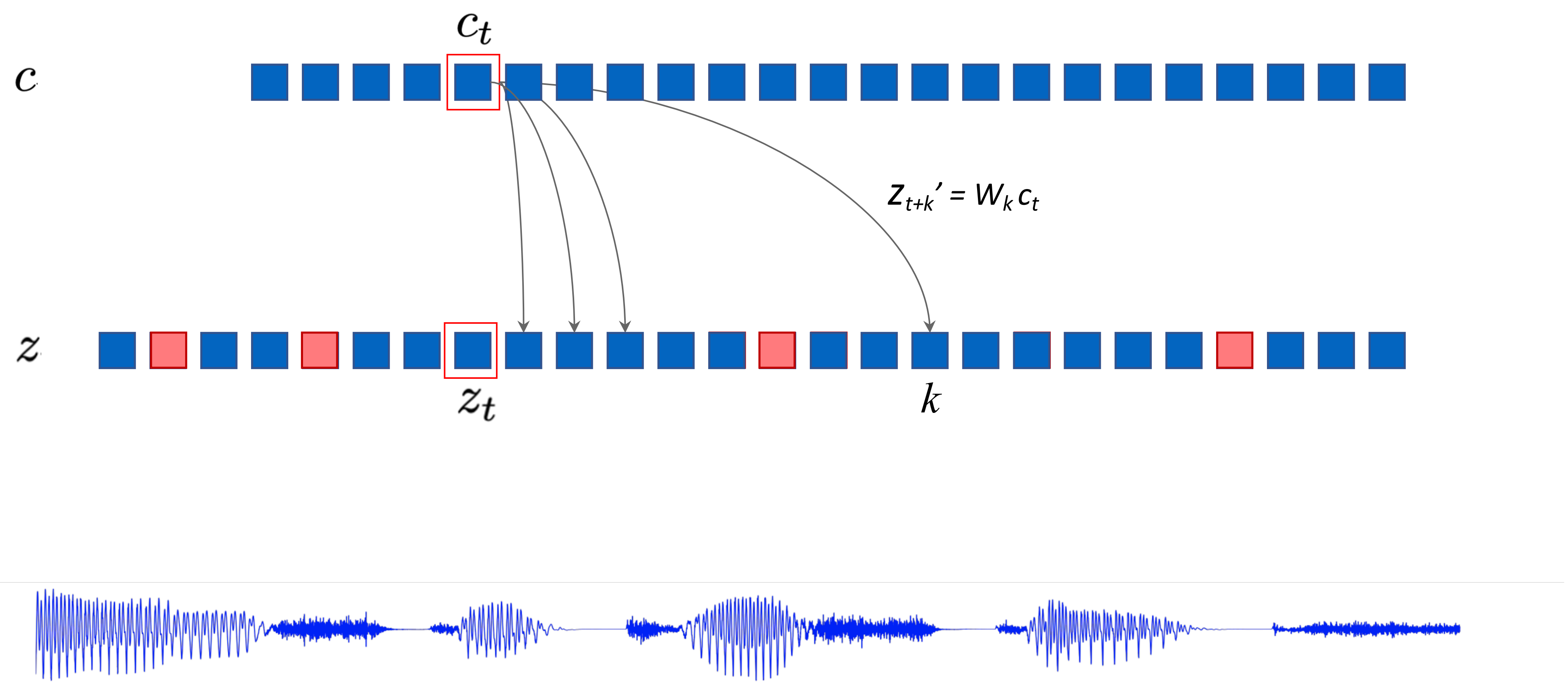
- Categorical cross-entropy loss of classifying the positive sample correctly

wav2vec


c
Aggregator
strided causal CNN (9 layers)
z
Encoder
strided CNN (5 layers)

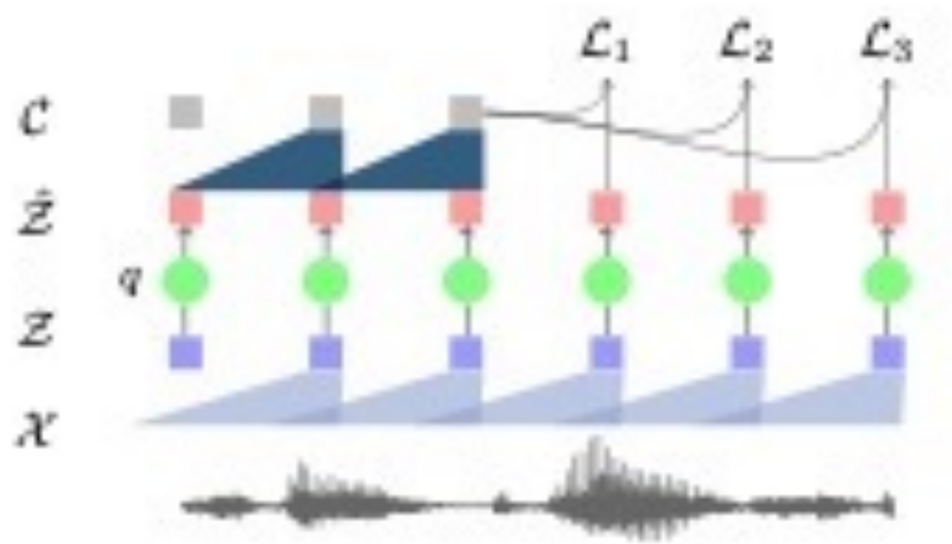
wav2vec

- Predict K steps into future using convTranspose
- Sample N negative z
- Model trained to distinguish predicted z from negative distractor samples

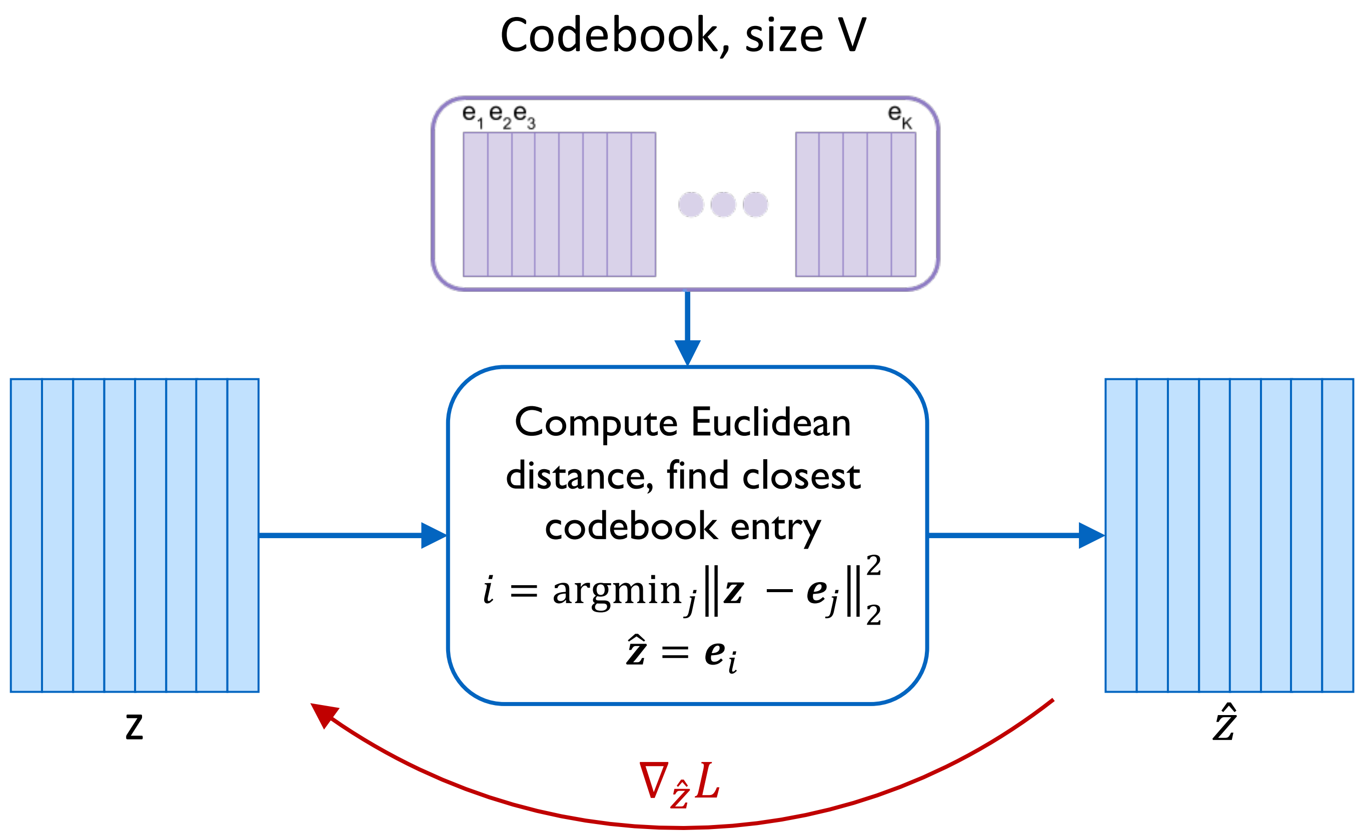


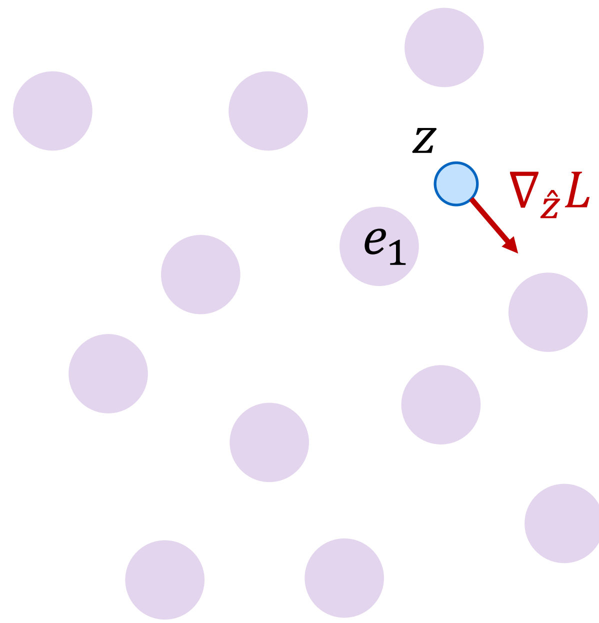
VQ-wav2vec

- Discretize the latent encoding of the raw audio, z, and pass this into aggregator to generate context c.
- Model still trained with categorical cross-entropy loss – want to predict future encoding z, from context vector c, and use negative samples to form the contrastive loss.
- Loss function has additional terms for the quantization module.



VQ-wav2vec: loss function





$$\mathcal{L} = \sum_{k=1}^{K} \mathcal{L}_k^{\text{wav2vec}} + \|\text{sg}(\mathbf{z}) - \hat{\mathbf{z}}\|^2 + \gamma\|\mathbf{z} - \text{sg}(\hat{\mathbf{z}})\|^2$$

Contrastive loss

Trains encoder and aggregator parameters

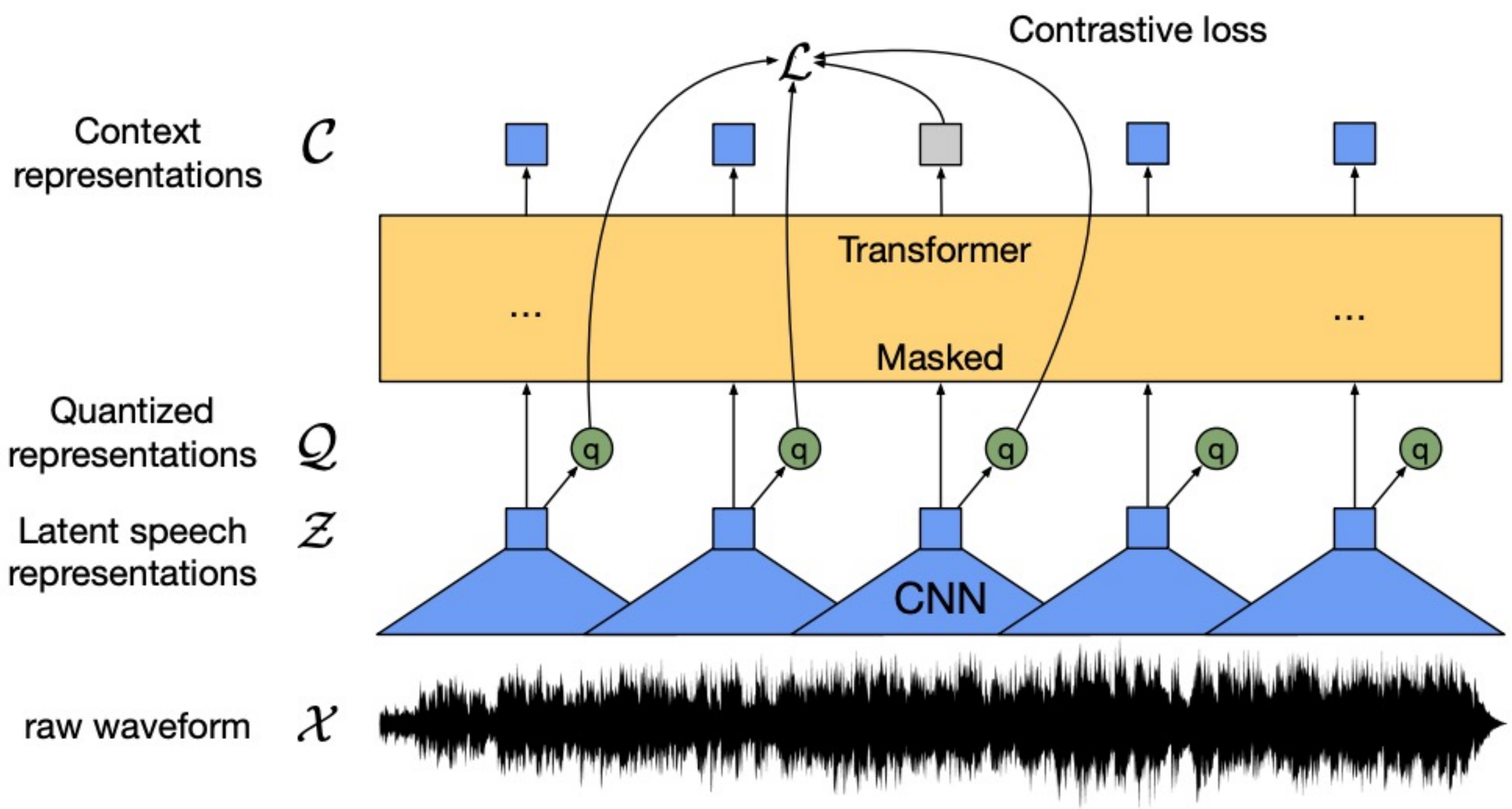
Vector Quantization loss

Trains embedding space: L_2 pushes codebook vectors towards encoder outputs

Commitment loss

Ensures encoder commits to a codebook entry without limitless growth

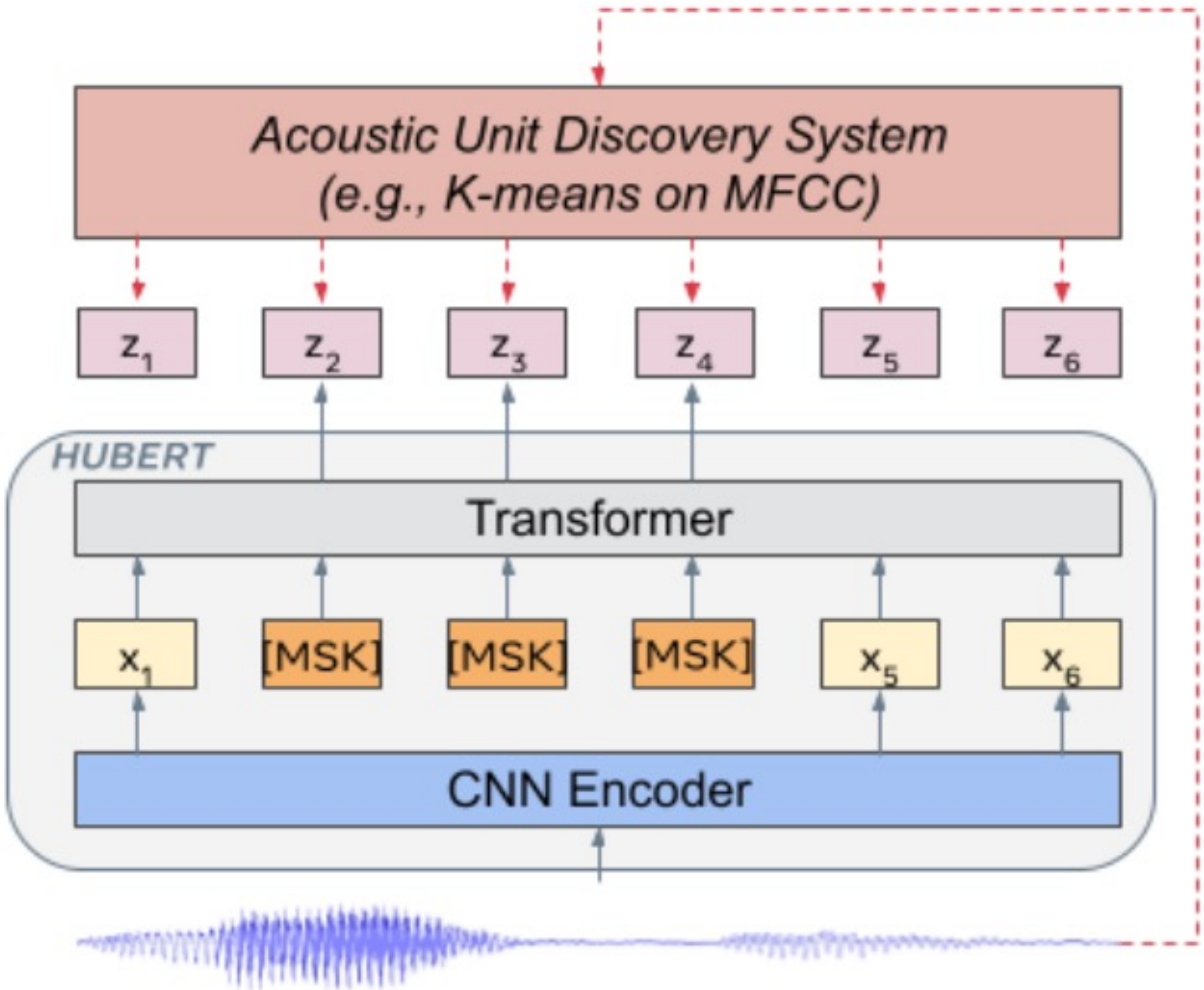
wav2vec 2.0 – masked acoustic modelling



Deep clustering and masked prediction

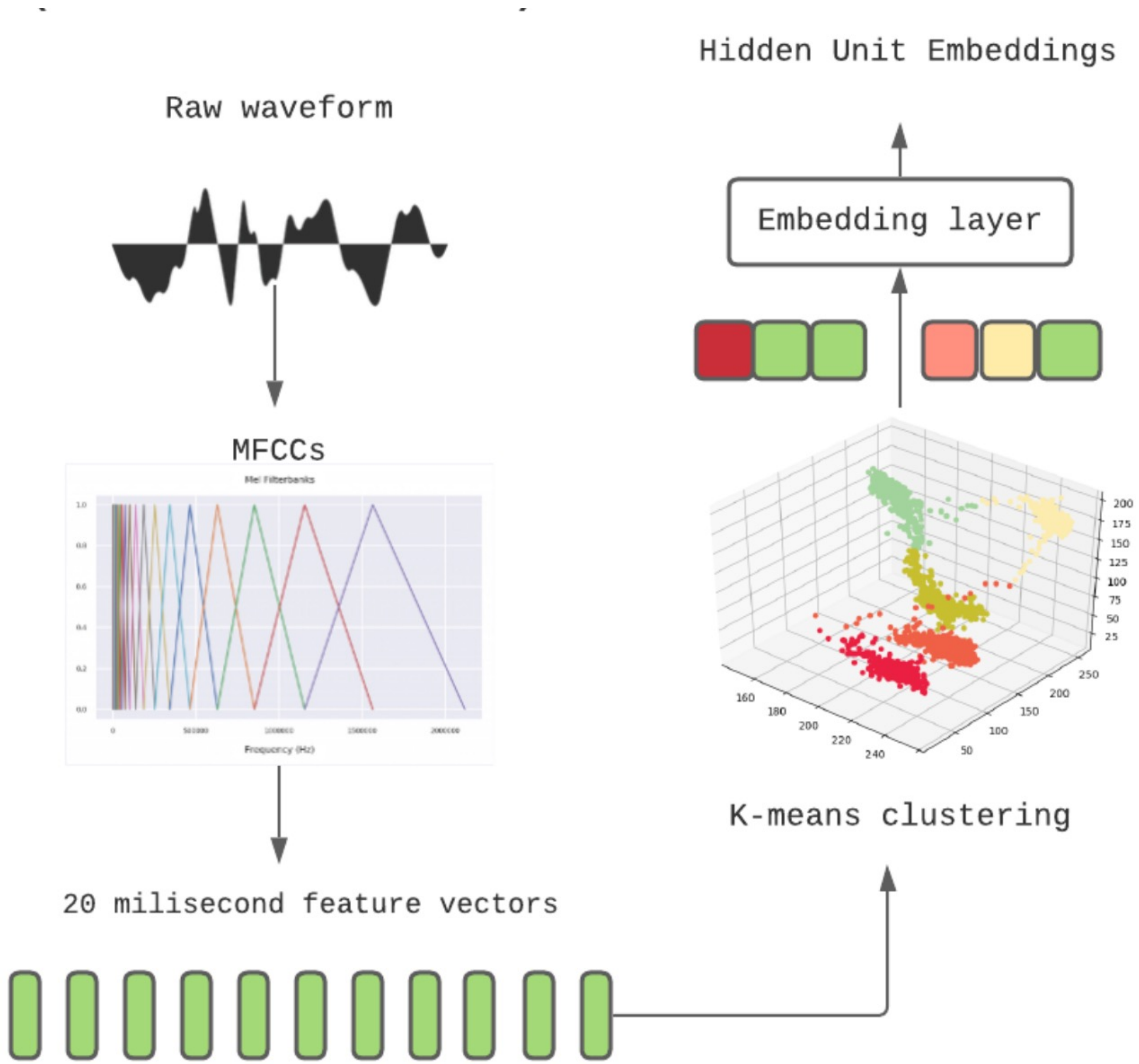
HuBERT: Hidden Unit BERT

HuBERT



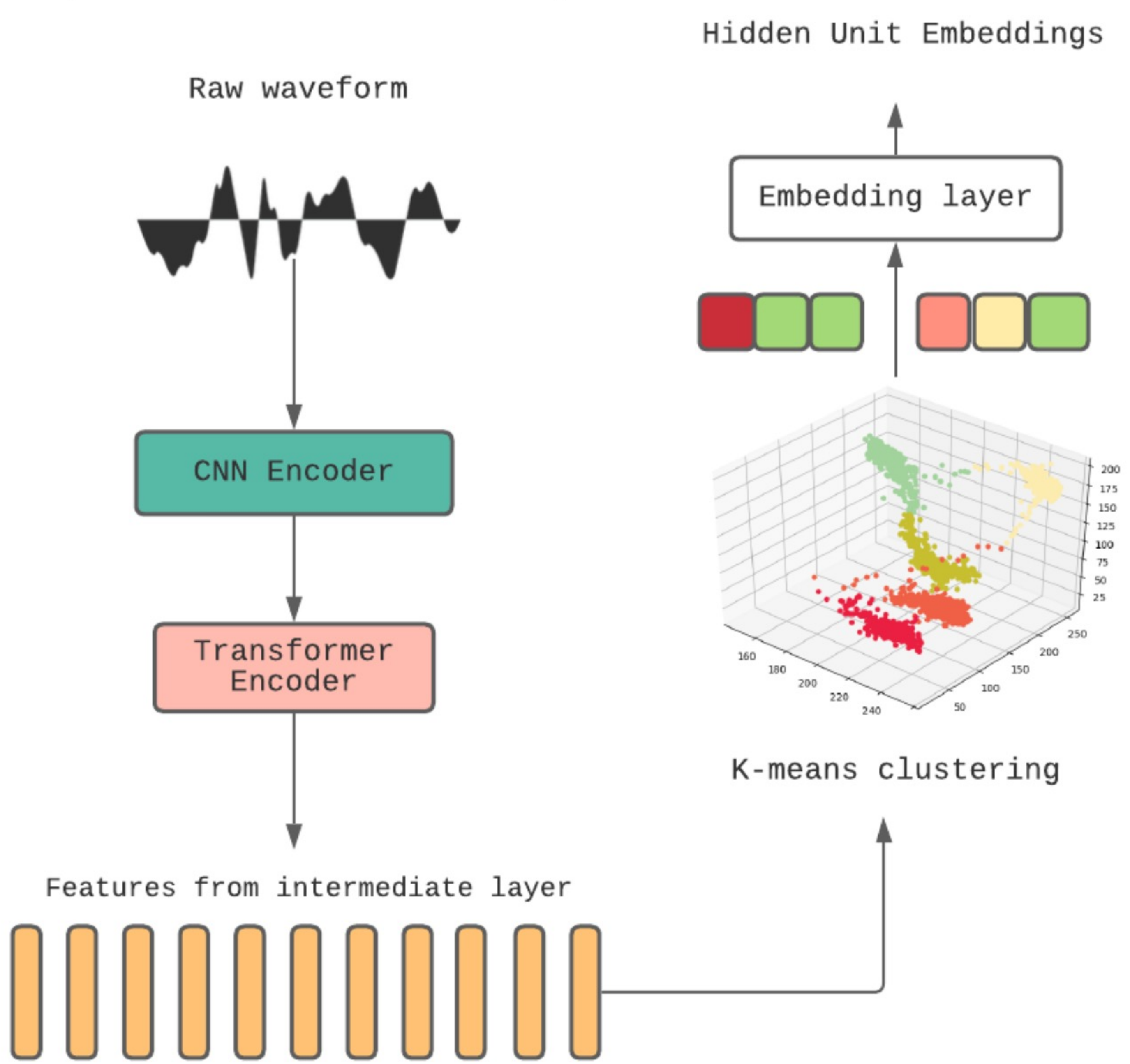
Acoustic Unit Discovery System
(e.g., K-means on MFCC)
z_1
z_2
z_3
z_4
z_5
z_6
HUBERT
Transformer
x_1
[MSK]
[MSK]
[MSK]
x_5
x_6
CNN Encoder

HuBERT: Clustering happens offline (MFCC)



https://blog.devgenius.io/hubert-explained-6ec7c2bf71fc

HuBERT: Clustering happens offline (latents)

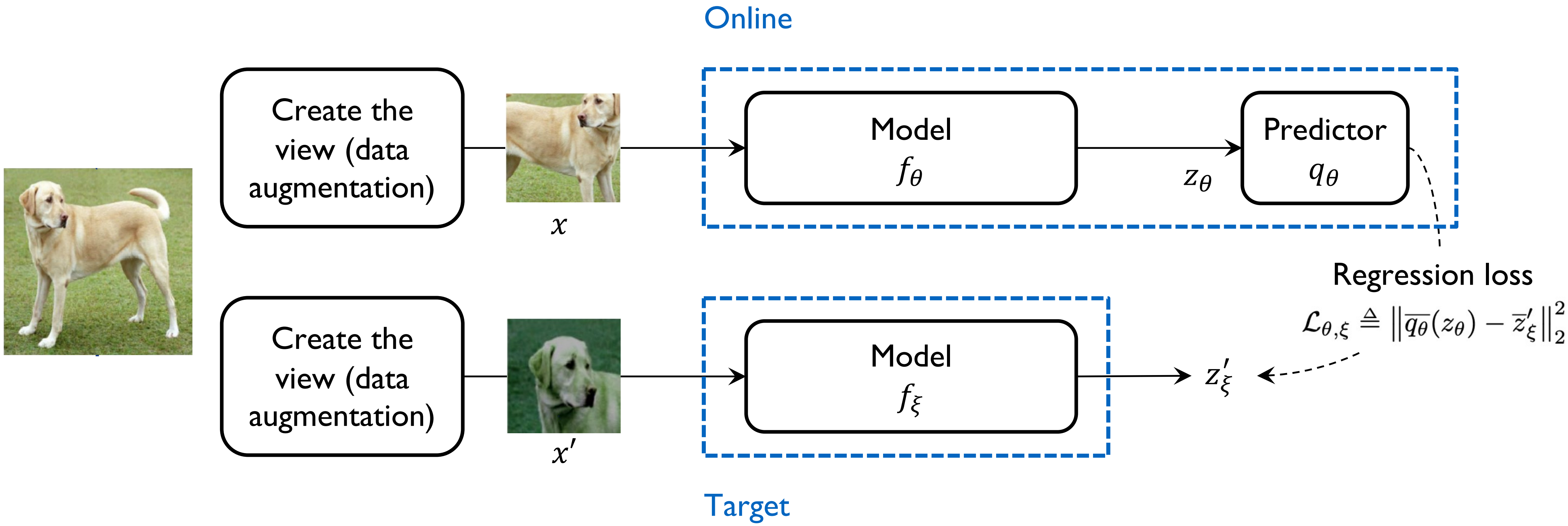


https://blog.devgenius.io/hubert-explained-6ec7c2bf71fc

Student – Teacher

BYOL
Data2vec
BPC

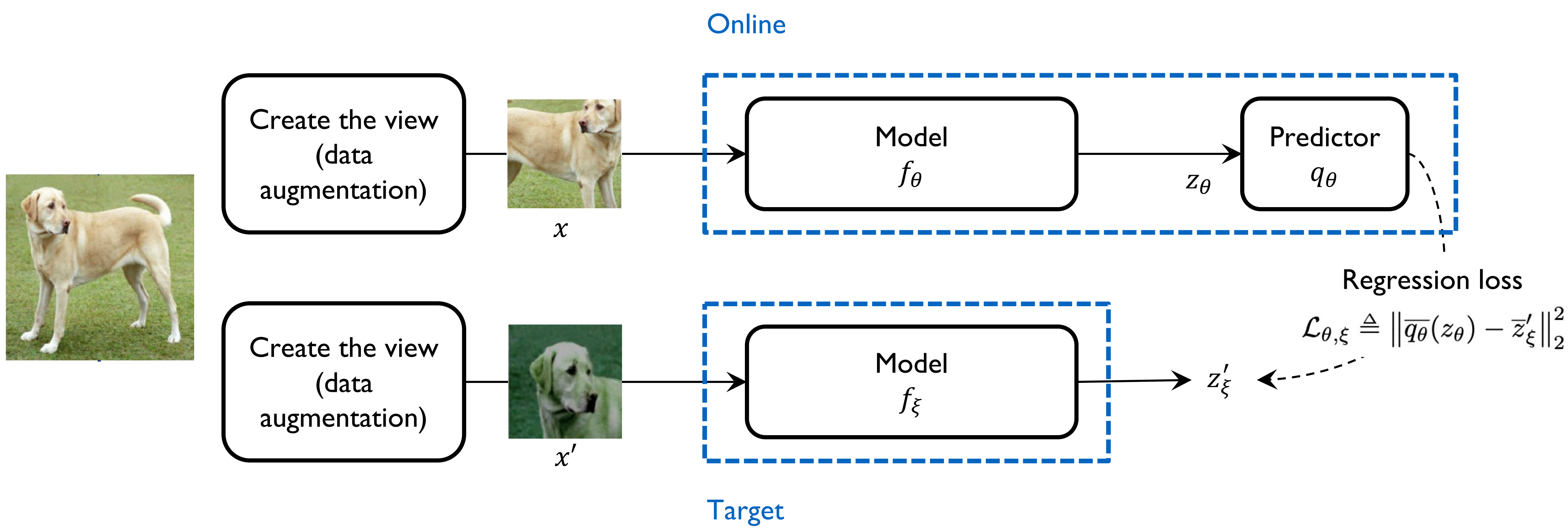
Bootstrap Your Own Latent (BYOL)



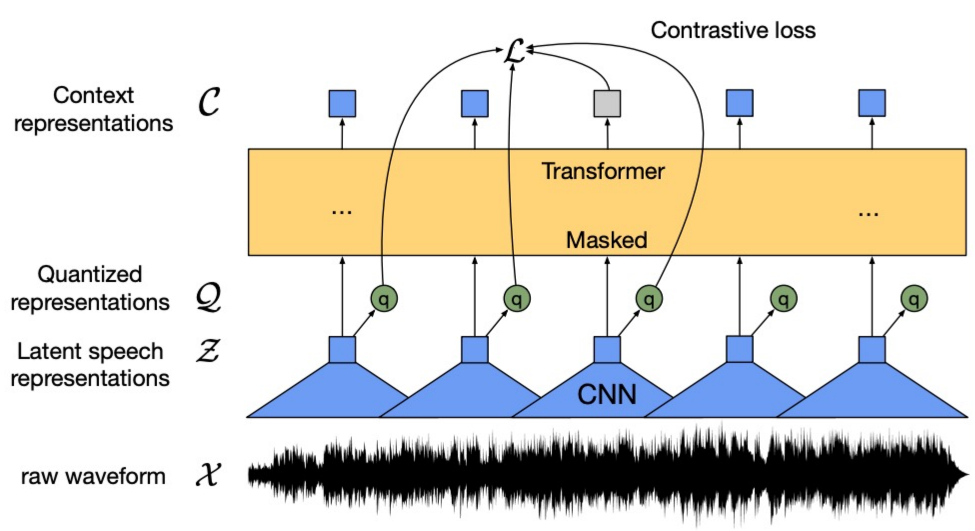
Iteratively train target network, parametrized as a moving average of online:

$$\xi \leftarrow \tau\xi + (1-\tau)\theta$$

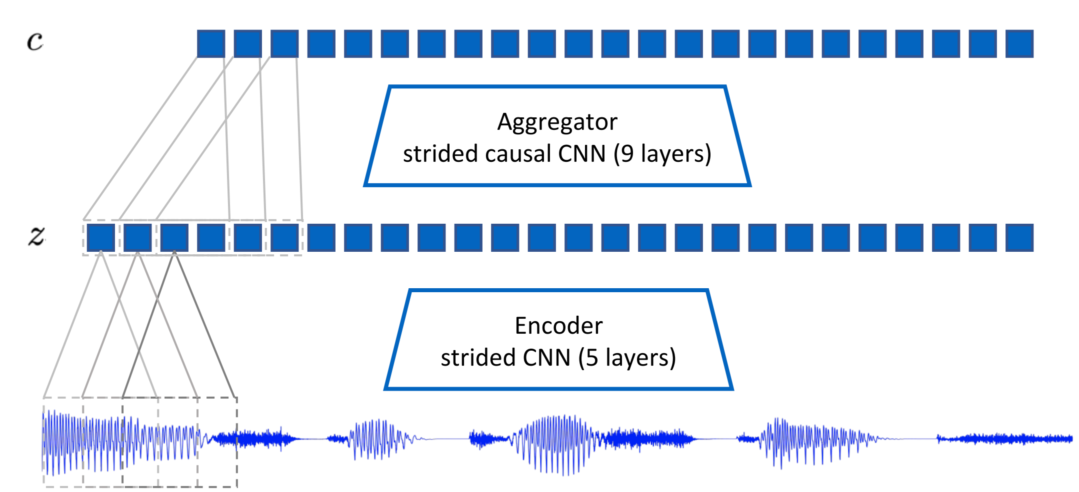
Bootstrap Predictive Coding (BPC)



Data2vec



BPC



Summary

- Supervised feature learning embedded within the ASR system is competitive with state-of-the-art systems that use handcrafted features.

- Self-supervised learning to extract a latent representation for features is a powerful approach minimizing information loss from the raw signal and leveraging large amounts of unlabelled data.

- Covered contrastive and non-contrastive SSL methods, and two pretext tasks: masked acoustic modelling and autoregressive modelling. All methods apply loss to the latent representations.

- Background reading:
  - A van den Oord et al (2018) "Representation learning with Contrastive Predictive Coding". *Arxiv.*
  - A Baevski et al (2020). "wav2vec 2.0: A framework for self-supervised learning of speech representations. *NeurIPS.*
  - W Hsu et al (2021). "HuBERT: Self-supervised speech representation learning by masked prediction of hidden units". *IEEE/ACM Transactions on Audio, Speech and Language processing.*
  - JB Grill et al (2020). "Bootstrap your own latent: A new approach to self-supervised learning". *NeurIPS.*
  - A Baevski et al (2022). "Data2vec: A general framework for self-supervised learning in speech, vision and language". *ICML.*