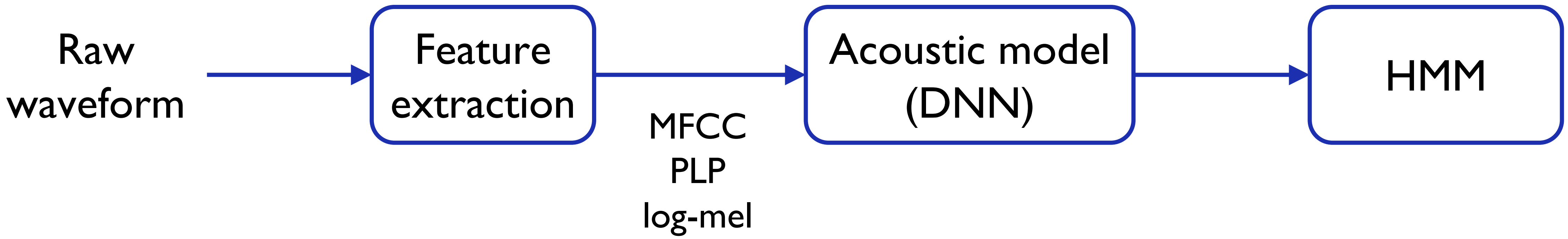
Unsupervised Raw Waveform Modelling: Self-supervised learning for Speech

Yumnah Mohamied

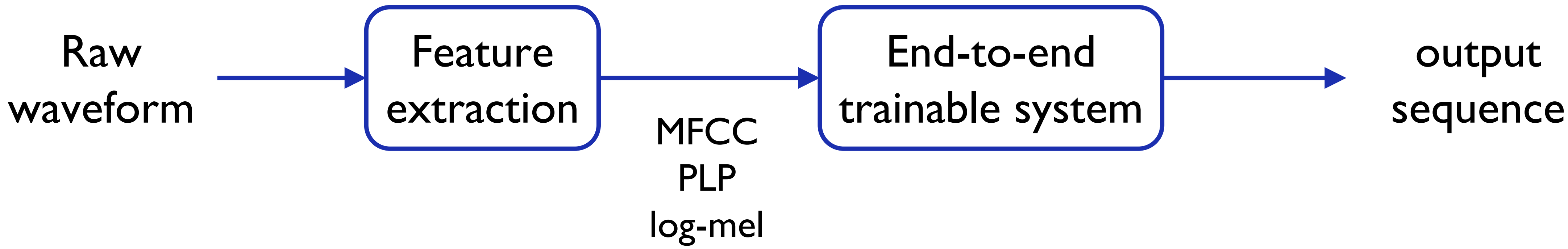
Automatic speech recognition – ASR lecture 18
24 March 2022

Divide and Conquer Strategy



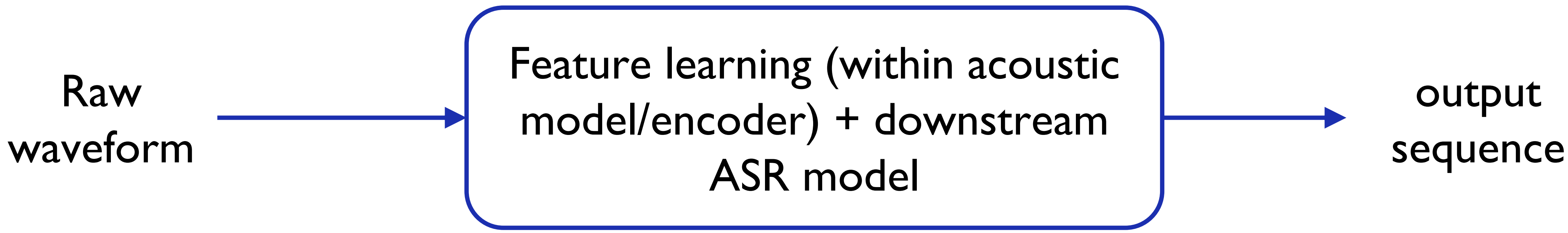
- Conventional ASR consists of composite subsystems trained and designed independently.
- Separates out feature extraction, acoustic modelling and decoding steps.
- Feature extraction is hand-crafted – based on prior knowledge of speech production and/or perception.

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



- 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

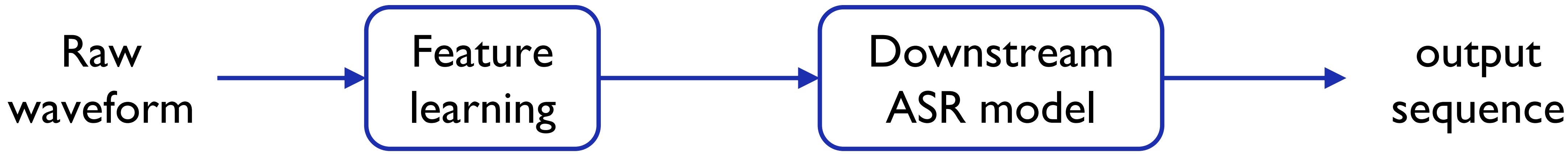
Raw waveform → Feature learning (within acoustic model/encoder) + downstream ASR model → output sequence

- 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.
- 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

Unsupervised Feature learning



- 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

Contrastive methods:
Wav2vec 2.0
(builds on CPC)

Clustering latent space:
HuBERT

Student-teacher:
Data2vec
(builds on BYOL)

Contrastive methods

CPC
wav2vec
VQ-wav2vec
Wav2vec 2.0
Wav2vec-C

Contrastive Predictive Coding

- Goal: learn to predict observations in the future from an encoded context window in the present (autoregressive modelling). Future observations are the "labels" created from the data (self-supervised learning)
- Intuition: learn representations that encode the underlying shared information between different parts of the high-dimensional speech signal
  - We have to predict further into the future so the model learns to infer more global structure rather than exploiting local smoothness of the signal.
- 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.
- Given an encoded context window in the present, c, and the future frame, x, modelling $p(x|c)$ (a generative model) to predict x, may not be optimal for extracting shared information between x and c.

CPC: Maximising Mutual Information

- We encode the future frame (the target, x) and the present context into compact representations which maximally preserve MI of the original signals x and c - we extract underlying latent variables that x and c have in common.
- 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)

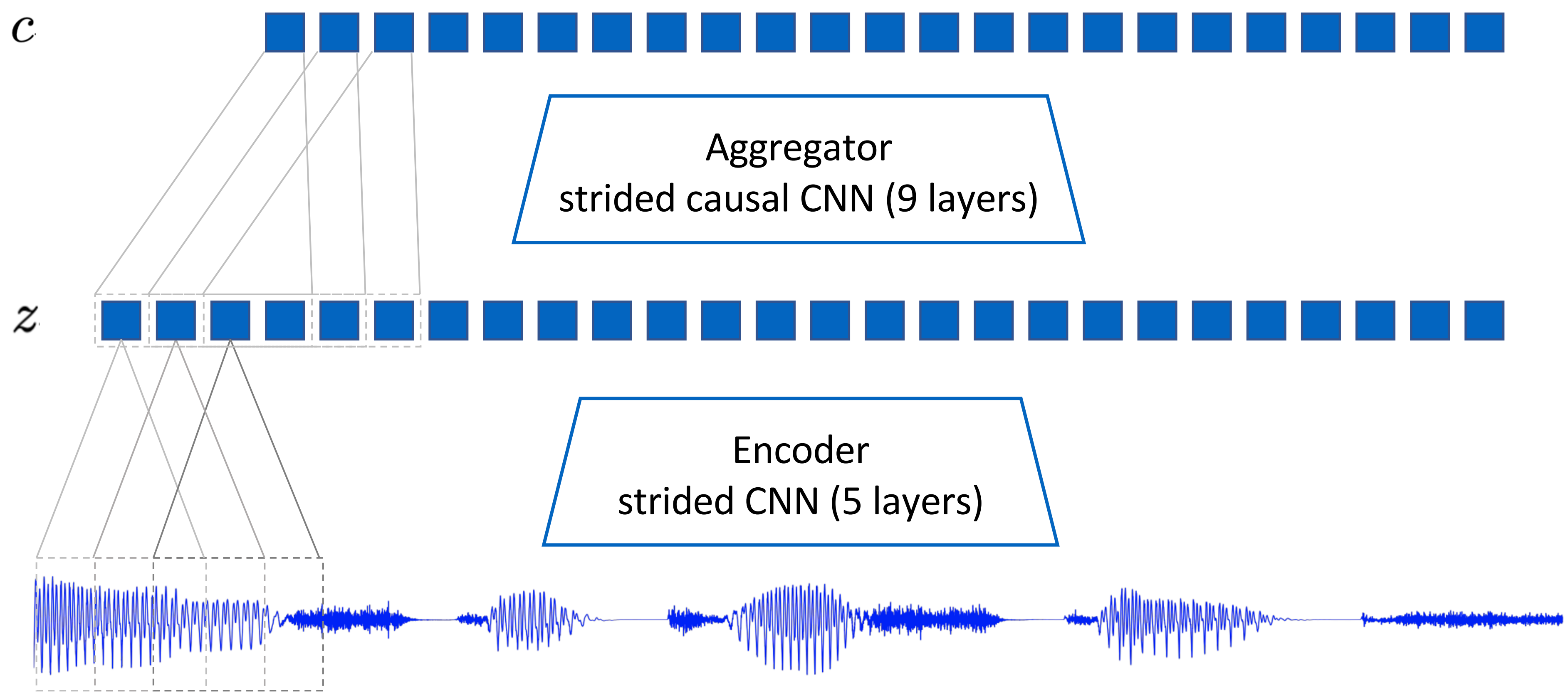
- We cannot evaluate p(x) or p(x|c) directly, but we can sample from these distributions
- One positive sample from p(x|c) (predicted future encoding), and N negative samples from the proposal distribution p(x) (random frame encodings within and across utterances)

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

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

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

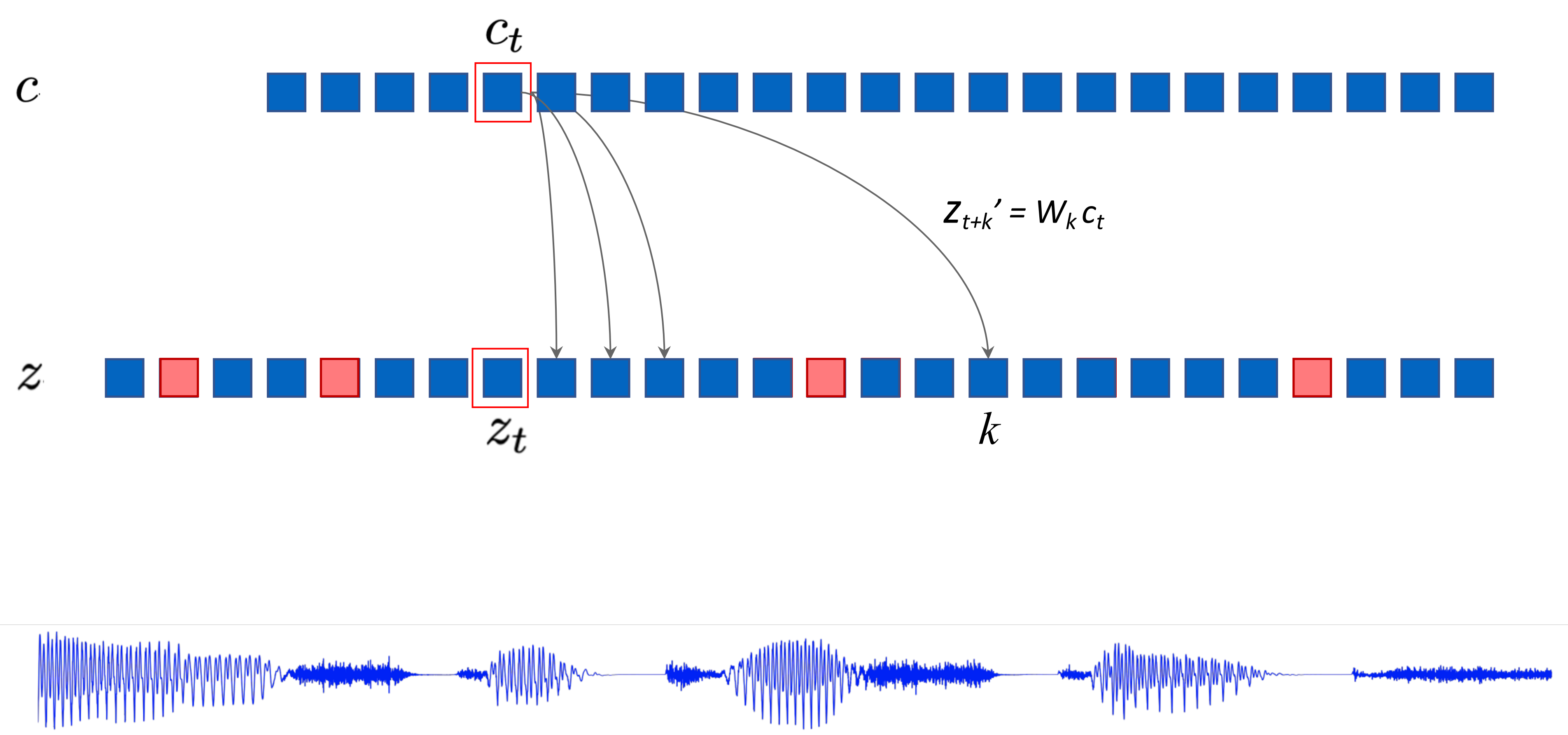
wav2vec



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

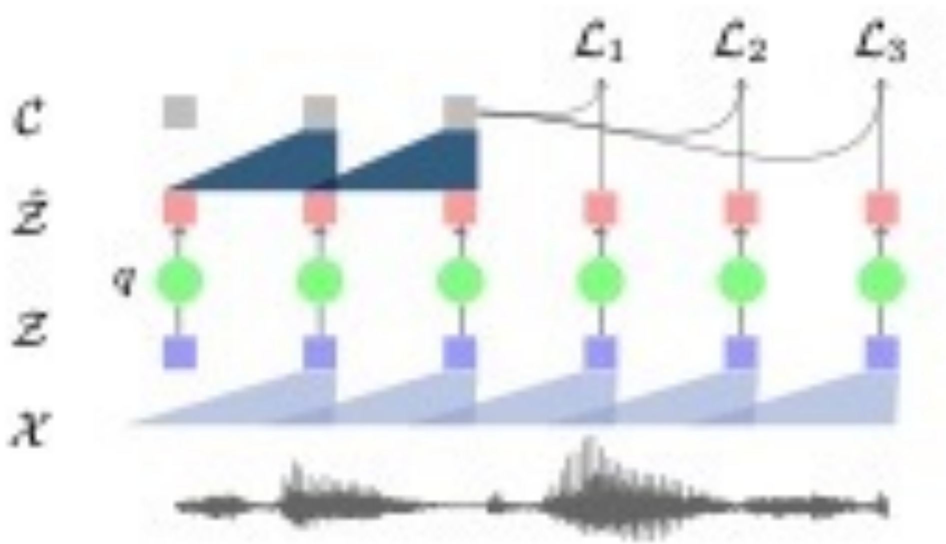
wav2vec

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

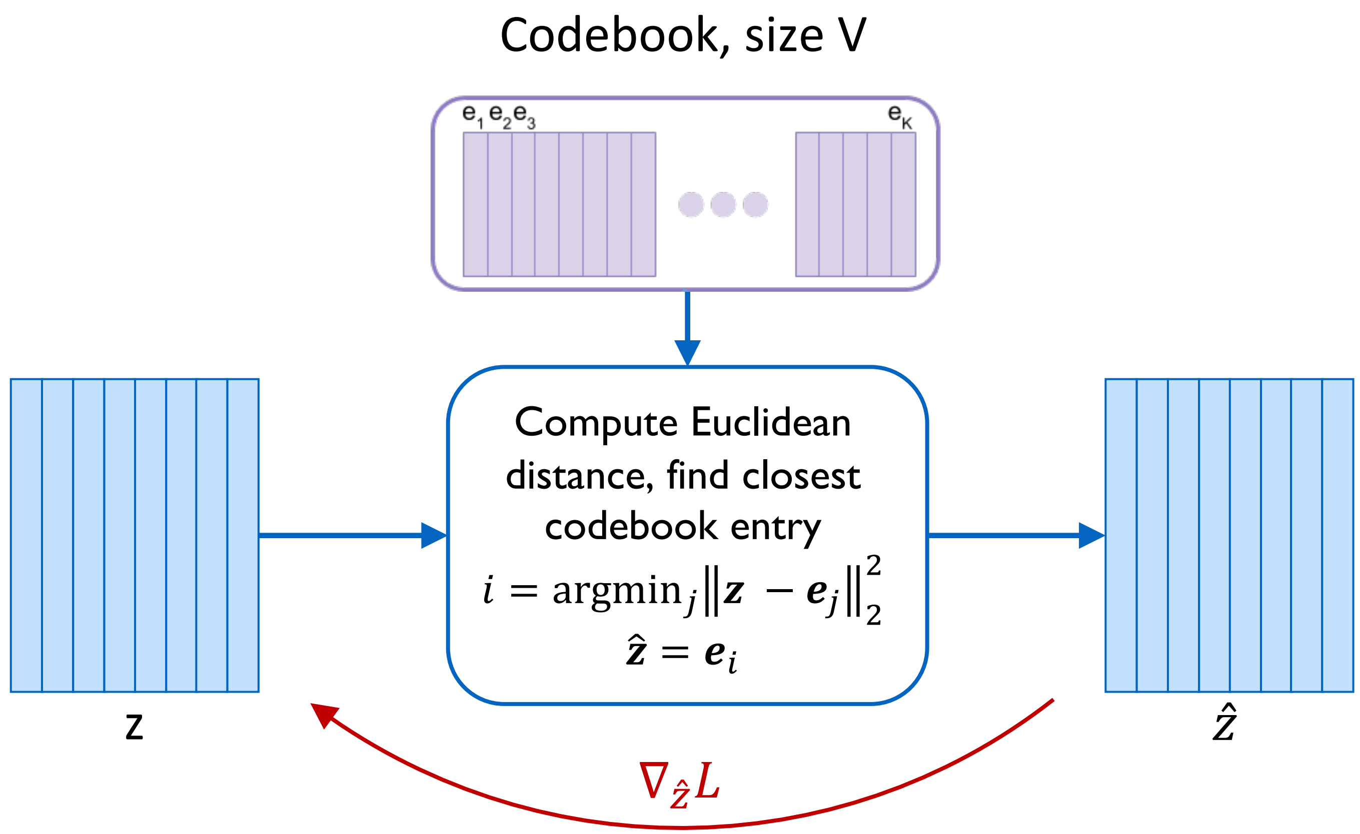


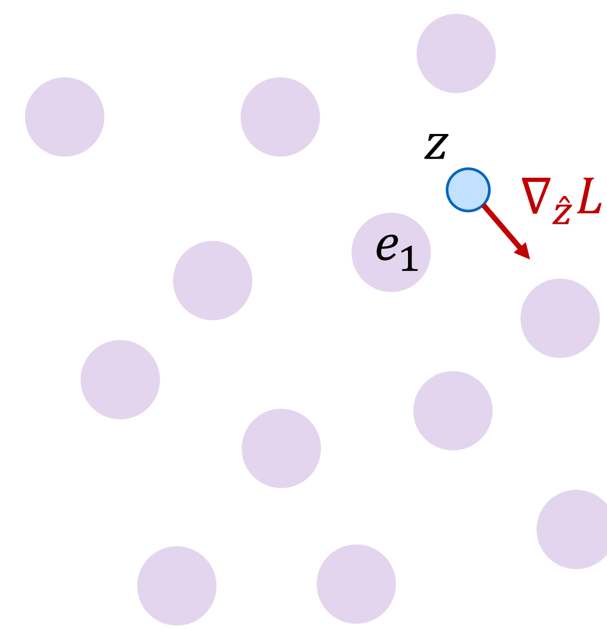
VQ-wav2vec

- 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

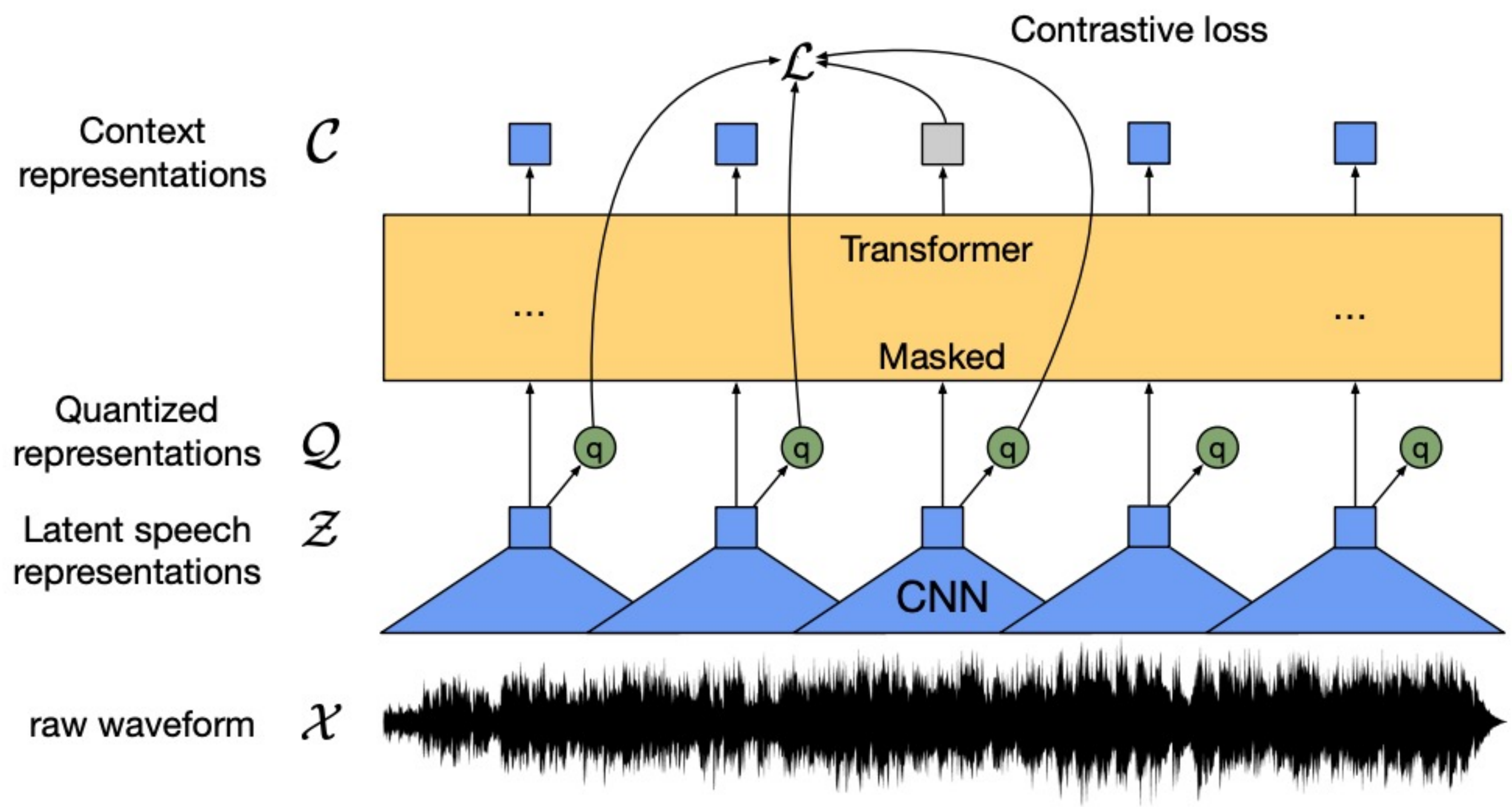




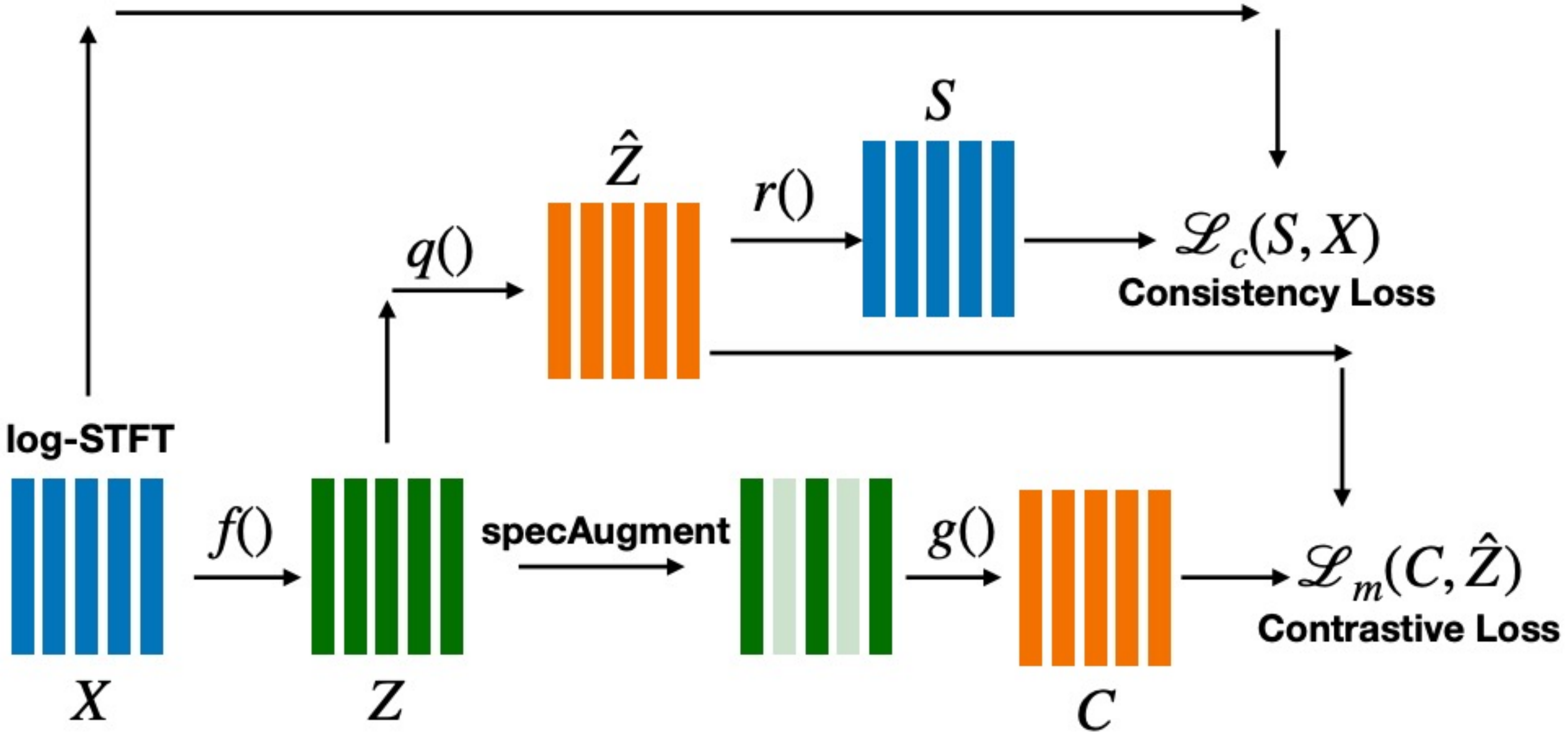
$$\mathcal{L} = \sum_{k=1}^{K} \mathcal{L}_k^{\text{wav2vec}} + \|\text{sg}(\mathbf{z}) - \hat{\mathbf{z}}\|^2 + \gamma\|\mathbf{z} - \text{sg}(\hat{\mathbf{z}})\|^2$$

Contrastive loss	Vector Quantisation loss	Commitment loss
Trains encoder and aggregator parameters	Trains embedding space: L_2 pushes codebook vectors towards encoder outputs	Ensures encoder commits to a codebook entry without limitless growth

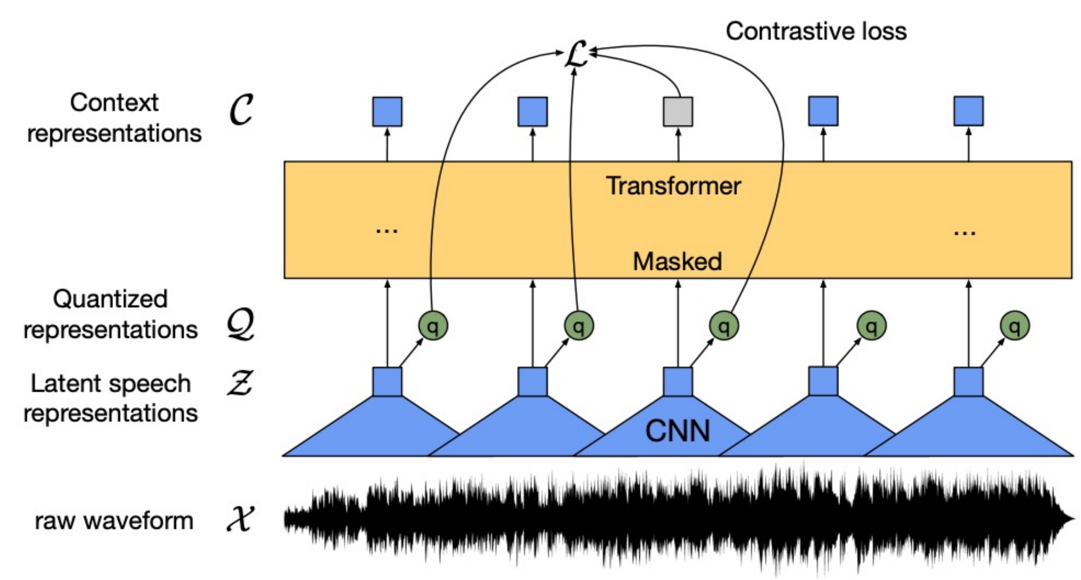
wav2vec 2.0 – masked prediction



Wav2vec-C



$\hat{Z}$
S
$q()$
$r()$
$\mathscr{L}_c(S, X)$
Consistency Loss
log-STFT
$f()$
specAugment
$g()$
$\mathscr{L}_m(C, \hat{Z})$
Contrastive Loss
X
Z
C

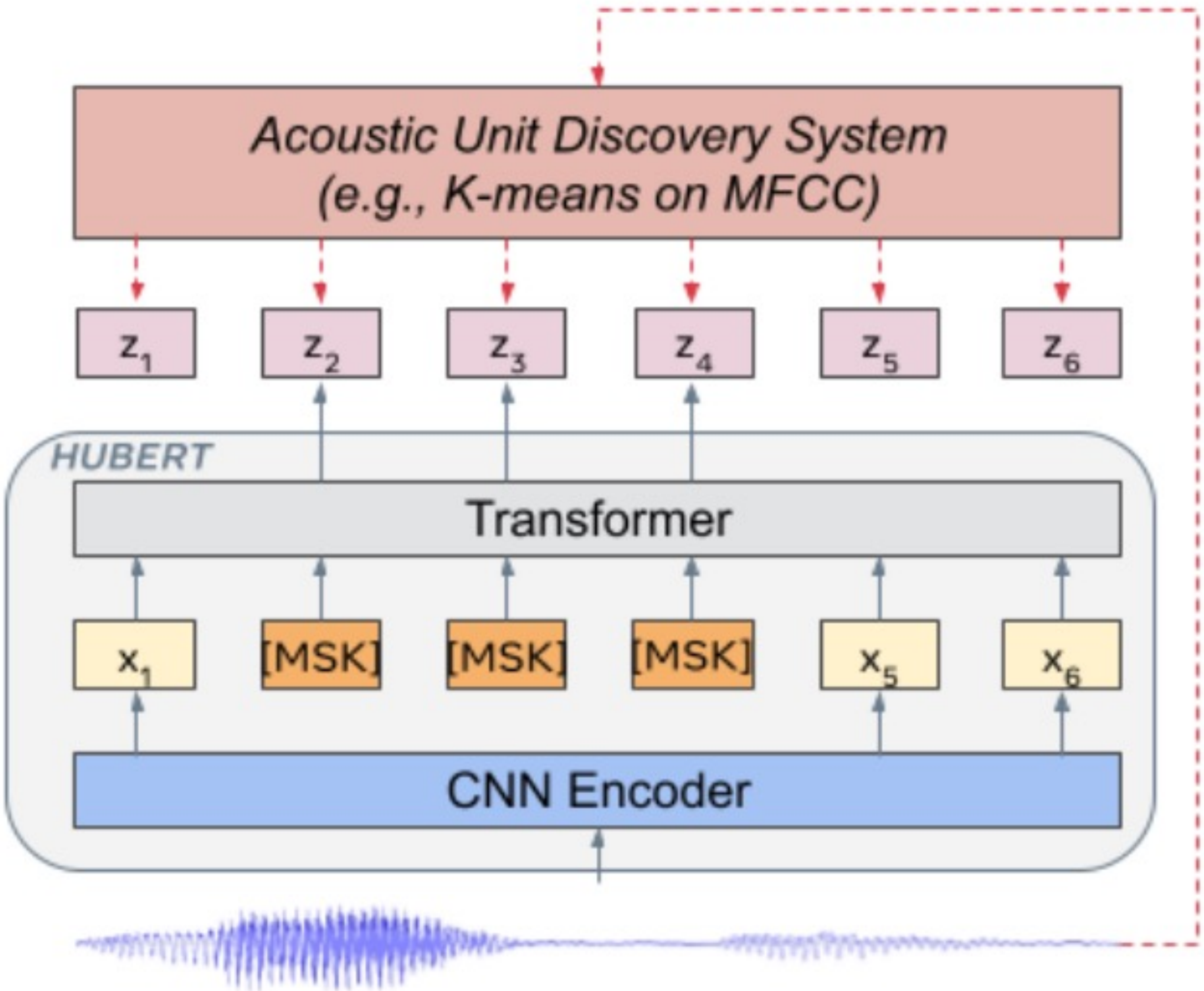


Contrastive loss
$\mathcal{L}$
Context representations $\mathcal{C}$
Transformer
...
Masked
Quantized representations $\mathcal{Q}$
q
Latent speech representations $\mathcal{Z}$
CNN
raw waveform $\mathcal{X}$

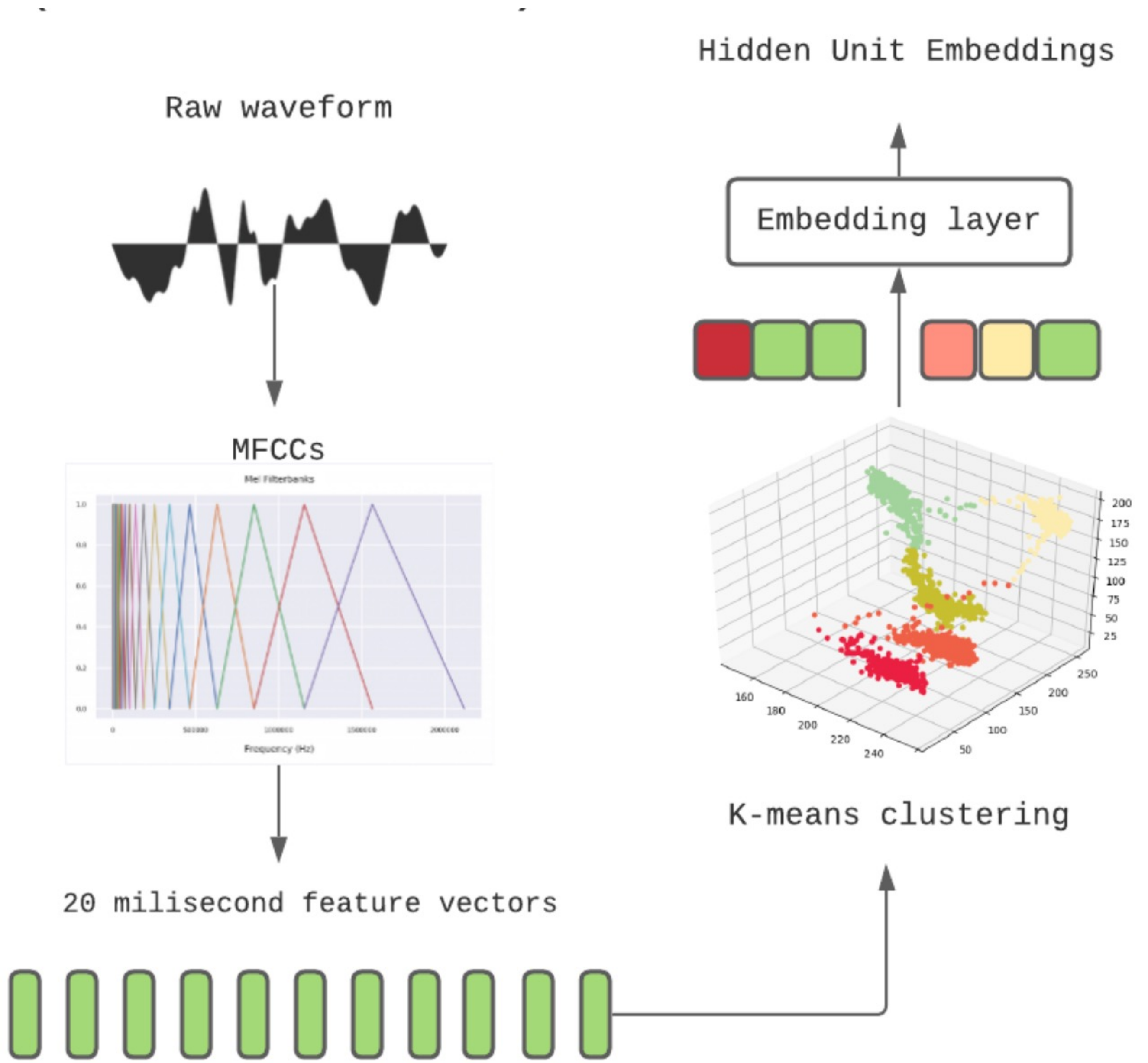
Deep clustering and masked prediction

HuBERT: Hidden Unit BERT

HuBERT

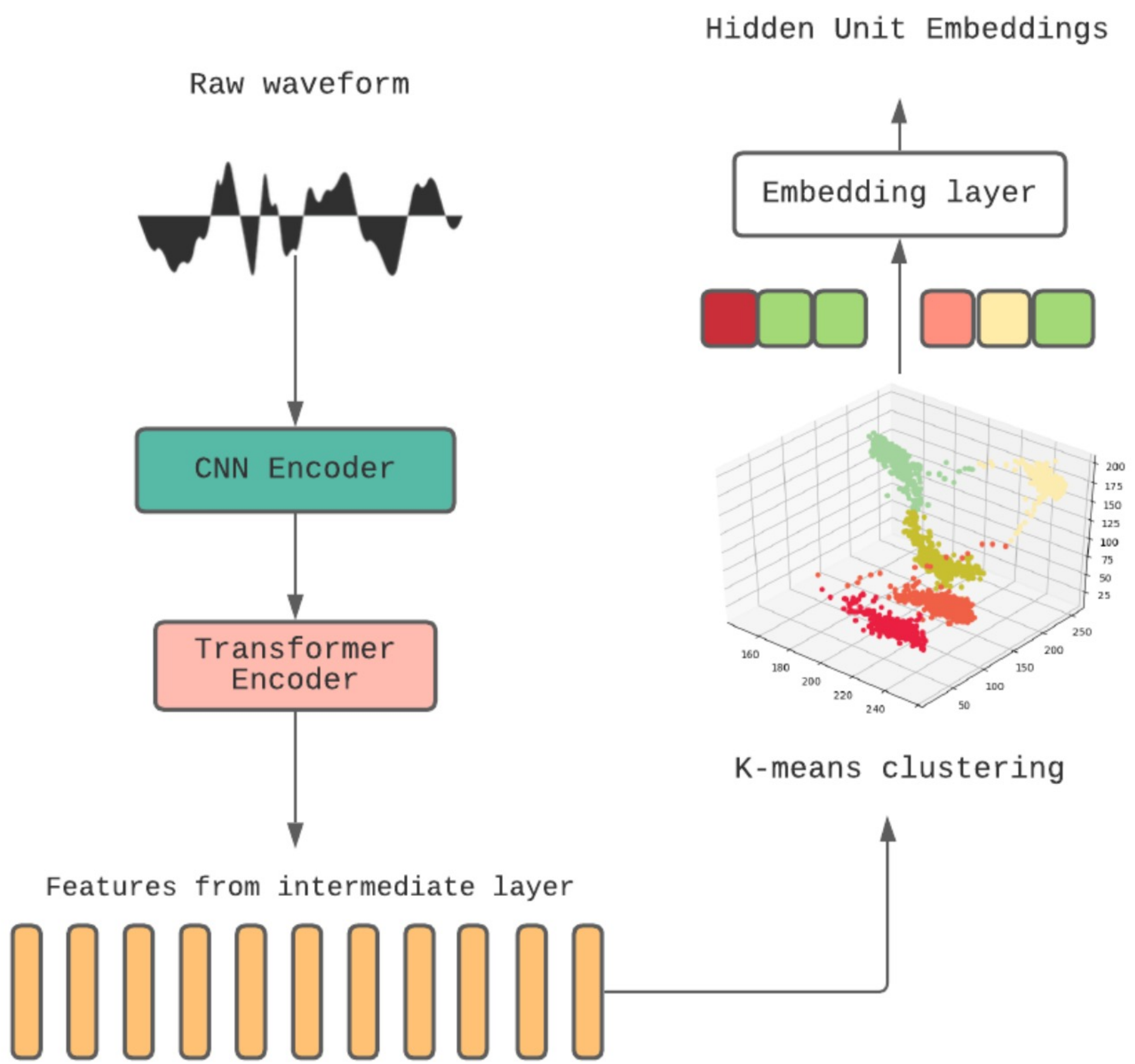

Acoustic Unit Discovery System
(e.g., K-means on MFCC)
z1
z2
z3
z4
z5
z6
HUBERT
Transformer
x1
[MSK]
[MSK]
[MSK]
x5
x6
CNN Encoder

HuBERT: Clustering happens offline (MFCC)



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

HuBERT: Clustering happens offline (latents)

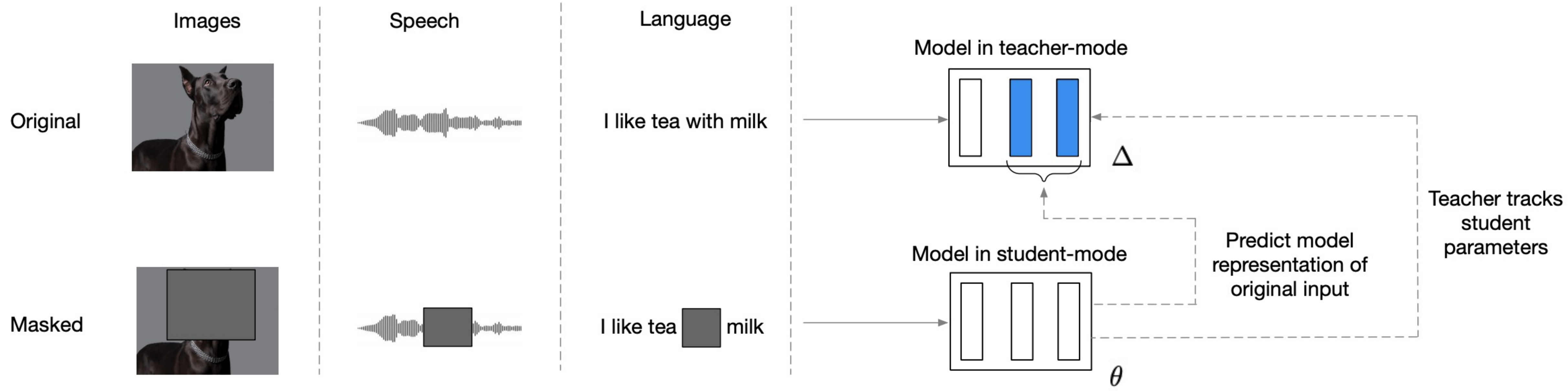


https://blog.devgenius.io/hubert-explained-6ec7c2bf71fc

Student - Teacher

Data2vec

Data2Vec



$$\mathcal{L}(y_t, f_t(x)) = \begin{cases} \frac{1}{2}(y_t - f_t(x))^2/\beta & |y_t - f_t(x)| \leq \beta \\ (|y_t - f_t(x)| - \frac{1}{2}\beta) & \text{otherwise} \end{cases}$$

$$\Delta \leftarrow \tau\Delta + (1-\tau)\,\theta$$

Summary

- Supervised feature learning embedded within the ASR system not competitive with state-of-the-art systems that use handcrafted features.
- Unsupervised feature learning to extract a latent representation is a powerful approach minimizing information loss from the raw signal and leveraging large amounts of unlabelled data.
- Wav2vec 2.0 uses contrastive loss, data2vec and HuBERT avoid this while still applying loss to the latent representations.
- Background reading:
  - A van den Oord et al (2018) "Representation learning with Contrastive Predictive Coding".
  - S Schneider et al (2019). "wav2vec: unsupervised pre-training for speech recognition". *Interspeech*.
  - 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*.
  - A Baevski et al (2022). "Data2vec: A general framework for self-supervised learning in speech, vision and language".