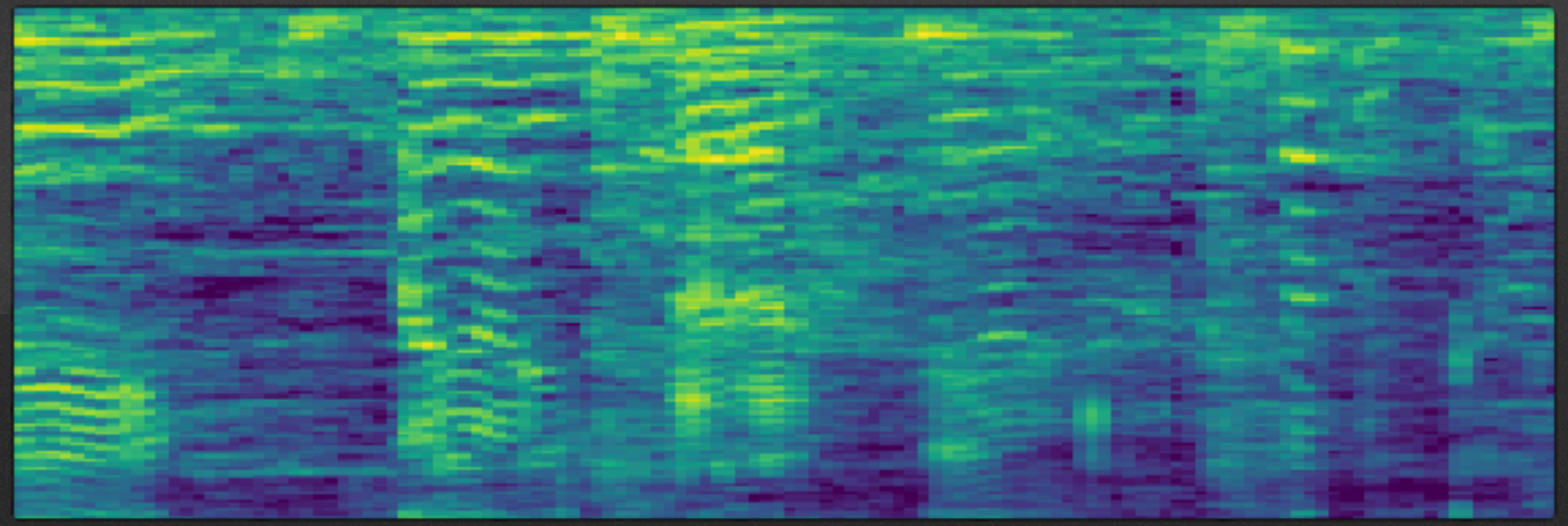
Speech analysis

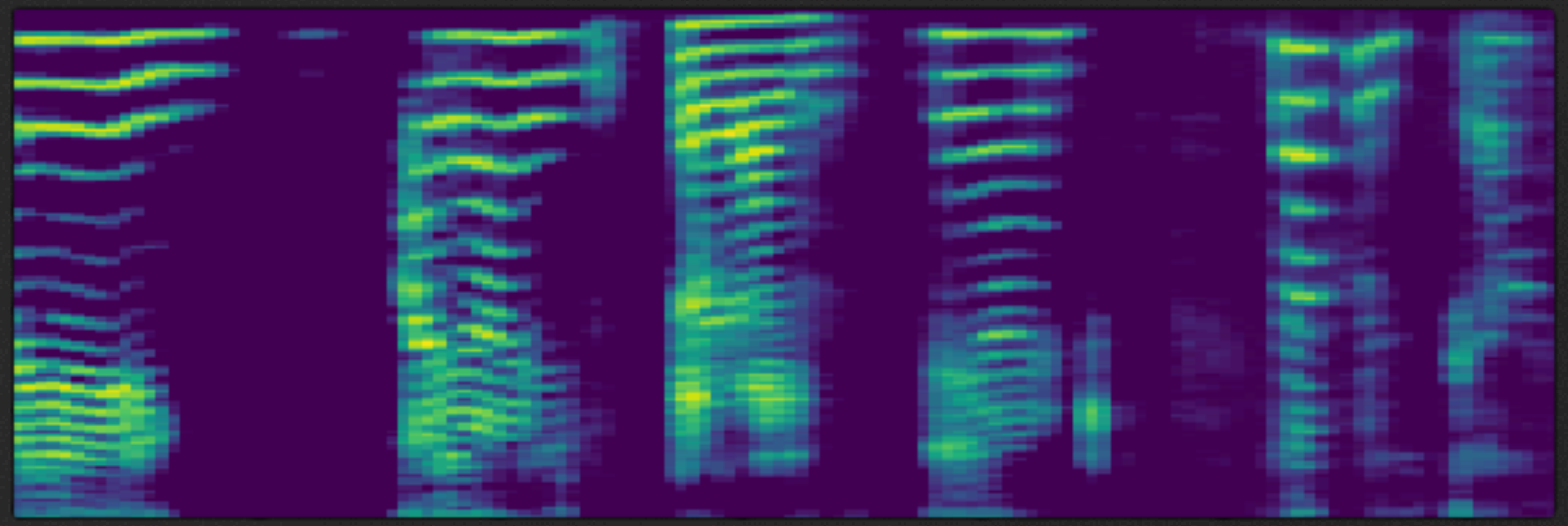
Dr. Evalds Urtans

Audio generation

Speech enhancement

- HuggingFace, TorchAudio Pre-trained models
- Wavenet, tacotron 2, TTS
- Noise removal, accent removal
  - asya.ai PESQ: 2.595
  - krisp.ai PESQ: 2.266





Audio classification

- HuggingFace, TorchAudio Pre-trained models
- **Whisper** STT / ASR
  - Our Latvian STT: CER: 12%
- Song classification
- Skaņu klasifikācija
- Neizmantot Kaldi

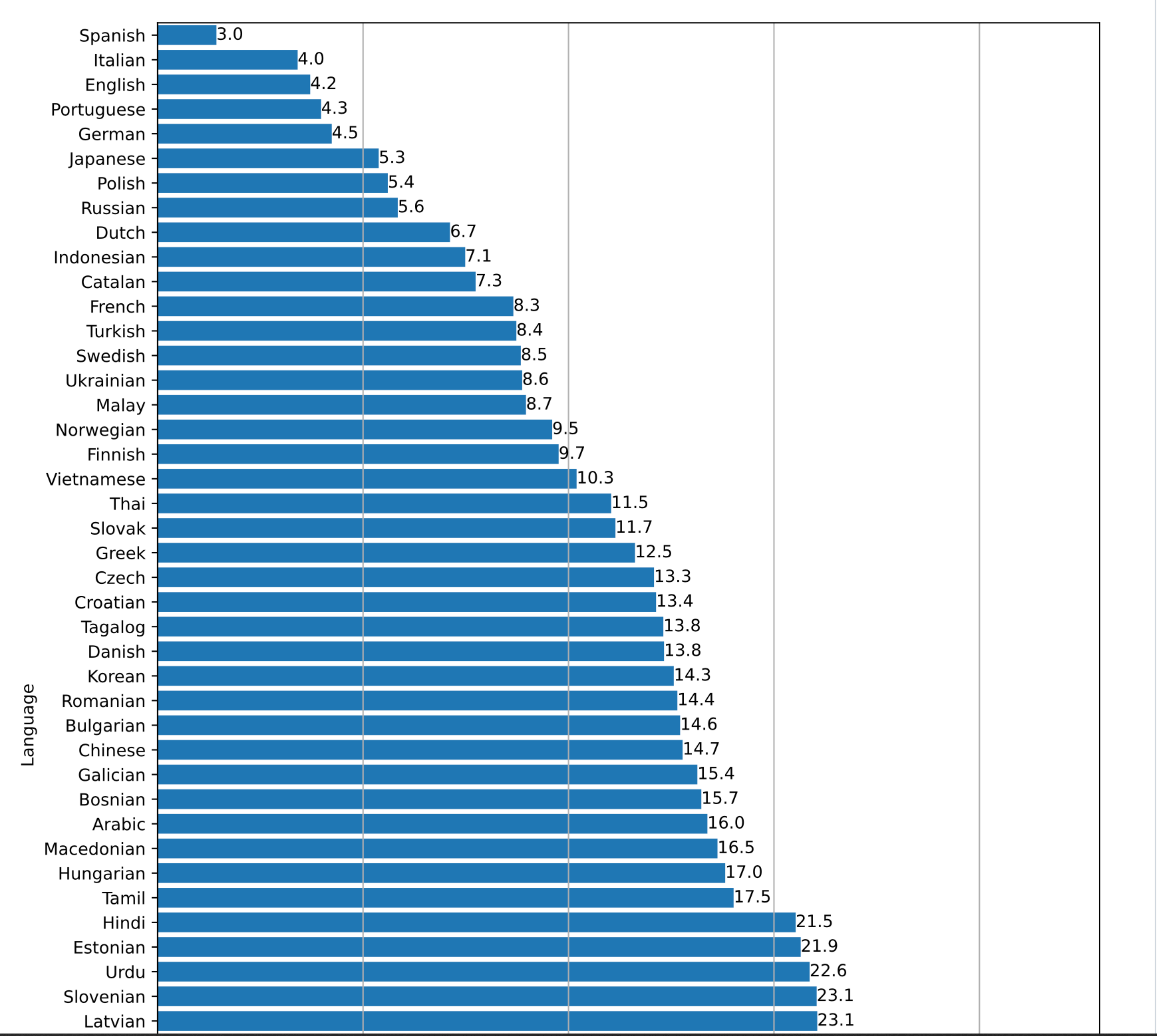
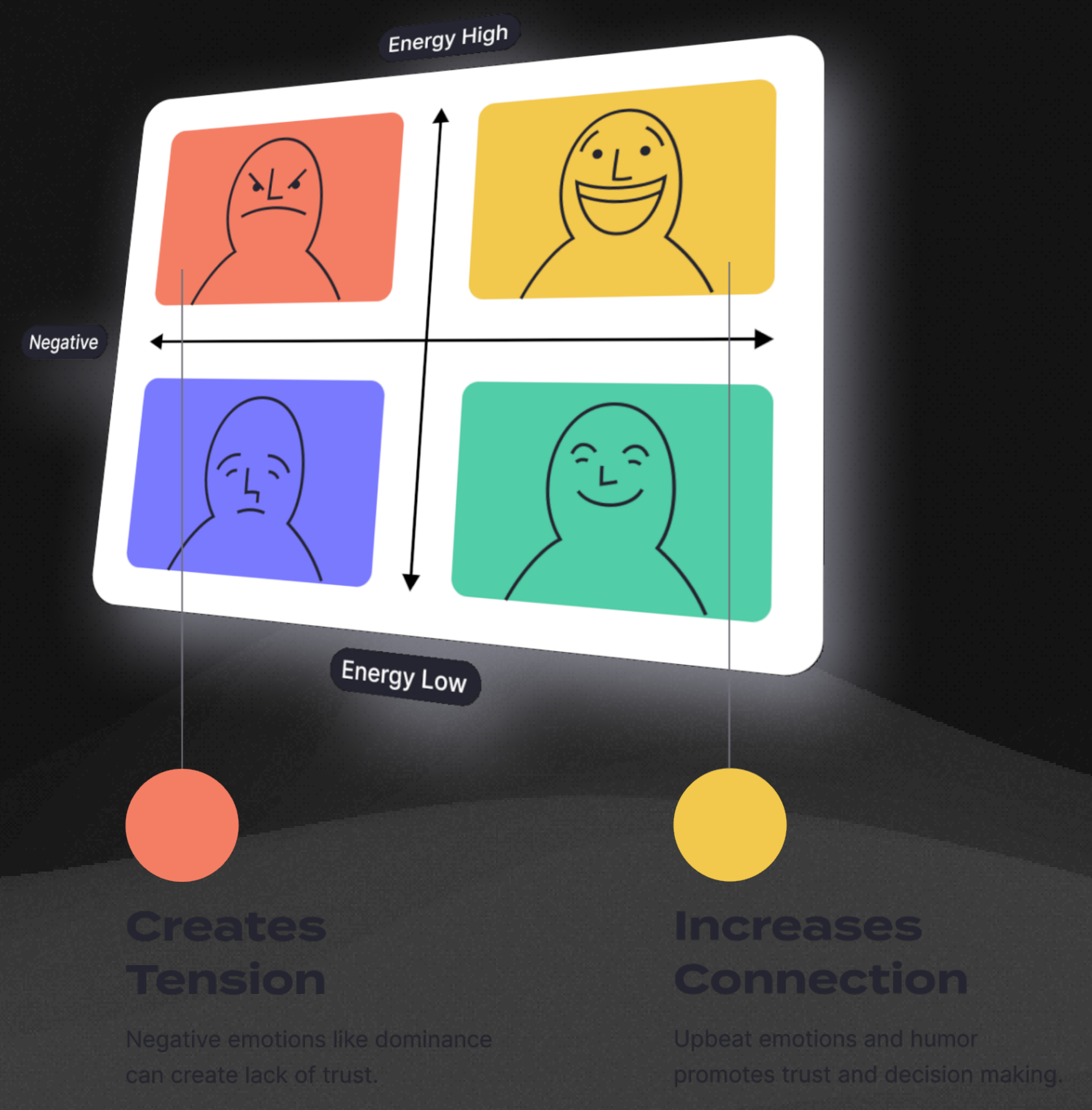
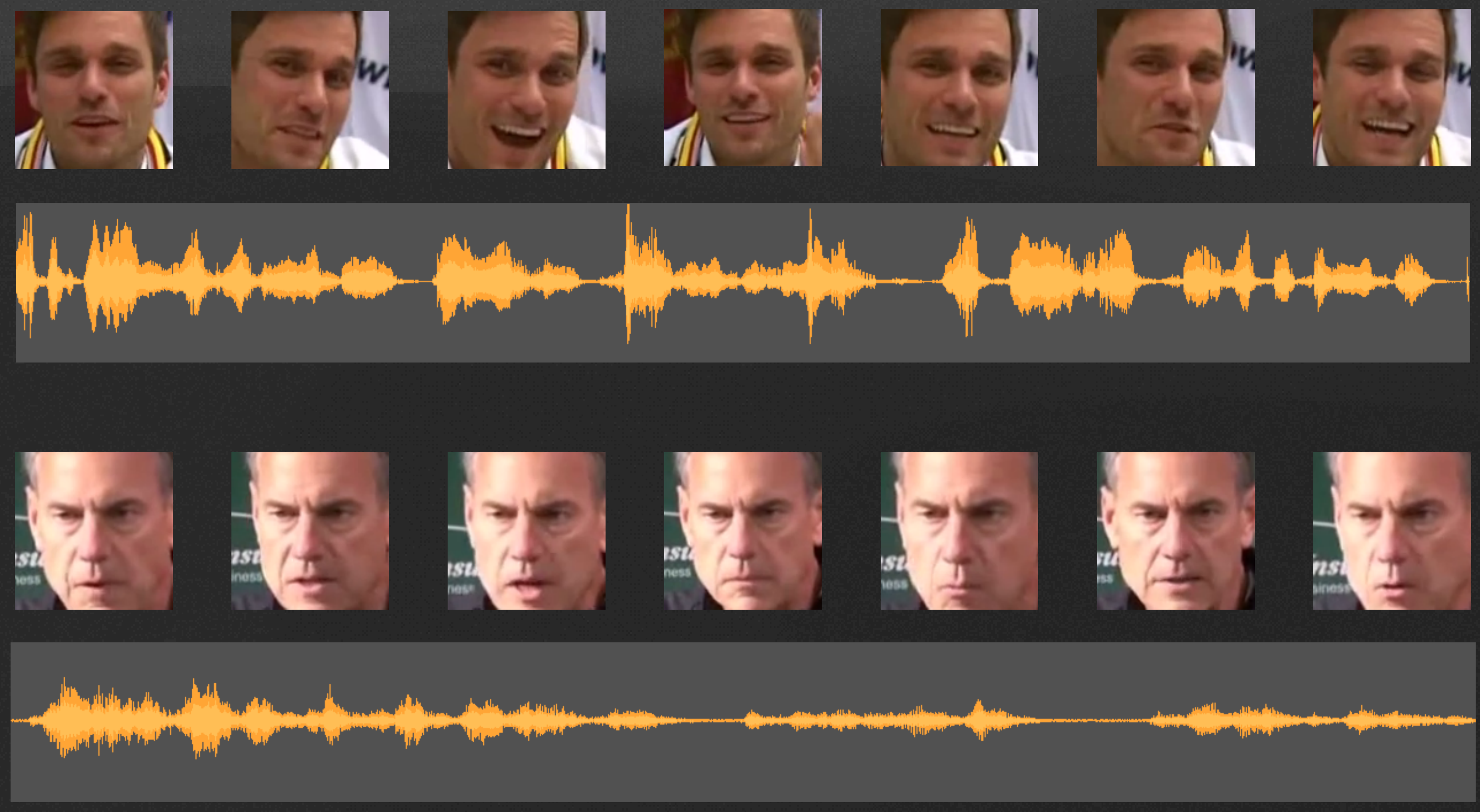


Image classification

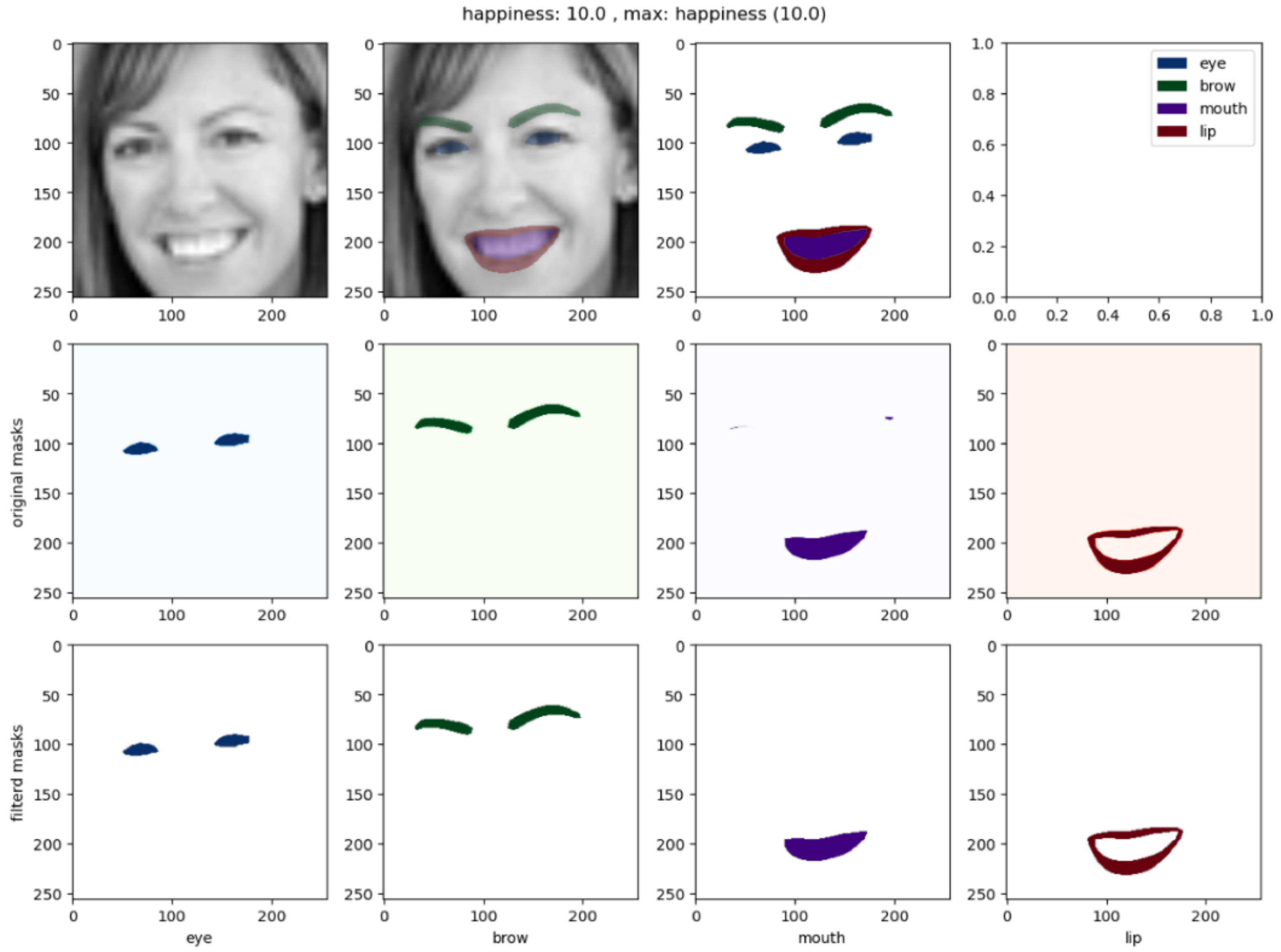
Emotion classification



- HuggingFace, TorchVision Pre-trained models (ImageNet)
- ConvNet, ResNet, DenseNet
- ViT, VisionTransformer
- Reset last layer, re-train with new classes
- **Can get away without training model CLIP**
- **Important data augmentations, cannot infer scale, rotation, color changes**



Emotion classification using Facial features



Cycle-GAN

Star-GAN

Emotion classification using tone of voice

Happiness

Interview of winners after a game

Anger

Interview of losers after a game

Text classification

- HuggingFacem TorchText Pre-trained models
- Word2Vec, GloVE, Sentence2Vec
- Sentiment classification, Named entity classification
- Much more expensive because of data, but way more precise (70% vs 99%)

Process

Low quality 8-16kHz audio 32bit float → Denoiser & speech enhancement model → Speech Activity, VoiceID models → End-to-end STT → Punctuation & style model

asya.ai PESQ: 2.595
krisp.ai PESQ: 2.266

asya.ai WER: 15%, CER: 10%

Extra

VoceID models

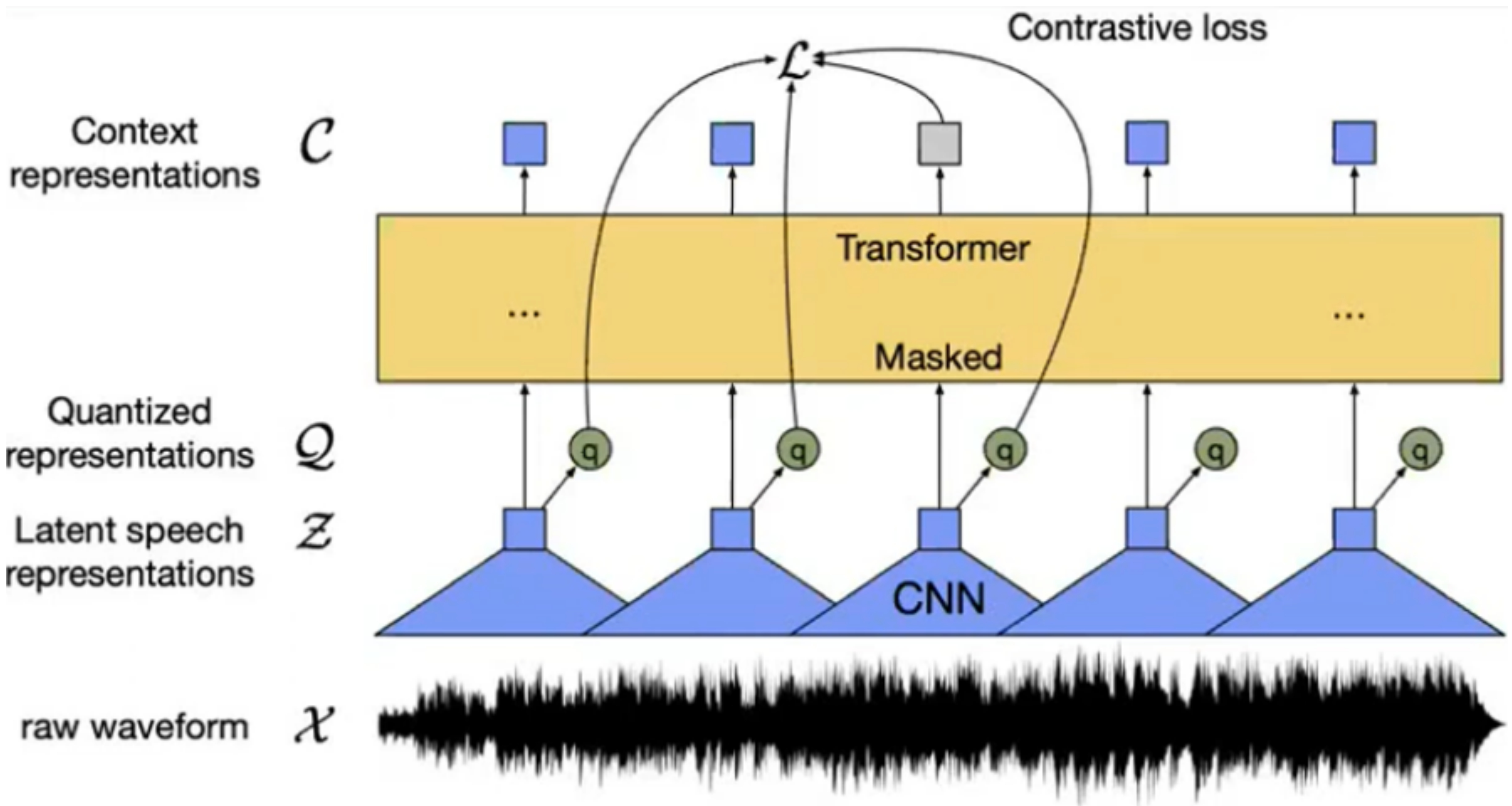
Emotions form
tone of voice, face,
sentiment, laughter

Topic classification

Flagging models

Wav2Vec, Data2Vec (2018-2023)

Wav2Vec, Data2Vec (2018-2023)



$f : \mathcal{X} \mapsto \mathcal{Z}$

multi-layer convolutional feature encoder

$\mathcal{X} \to$ input raw audio

$\mathcal{Z} = (z_1, z_2, \dots, z_T) \to$ latent speech representations

$g : \mathcal{Z} \mapsto \mathcal{C}$

transformer

$\mathcal{C} = (c_1, c_2, \dots, c_T) \to$ representations capturing information from the entire sequence

Instead of fixed positional embeddings which encode absolute positional information, use a convolutional layer which acts as relative positional embedding.

$\mathcal{Z} \mapsto \mathcal{Q}$

quantization module

$\mathcal{Q} = (q_1, q_2, \dots, q_T)$

diversity loss: encourage the model to use the codebook entries equally often ←

Pre-training

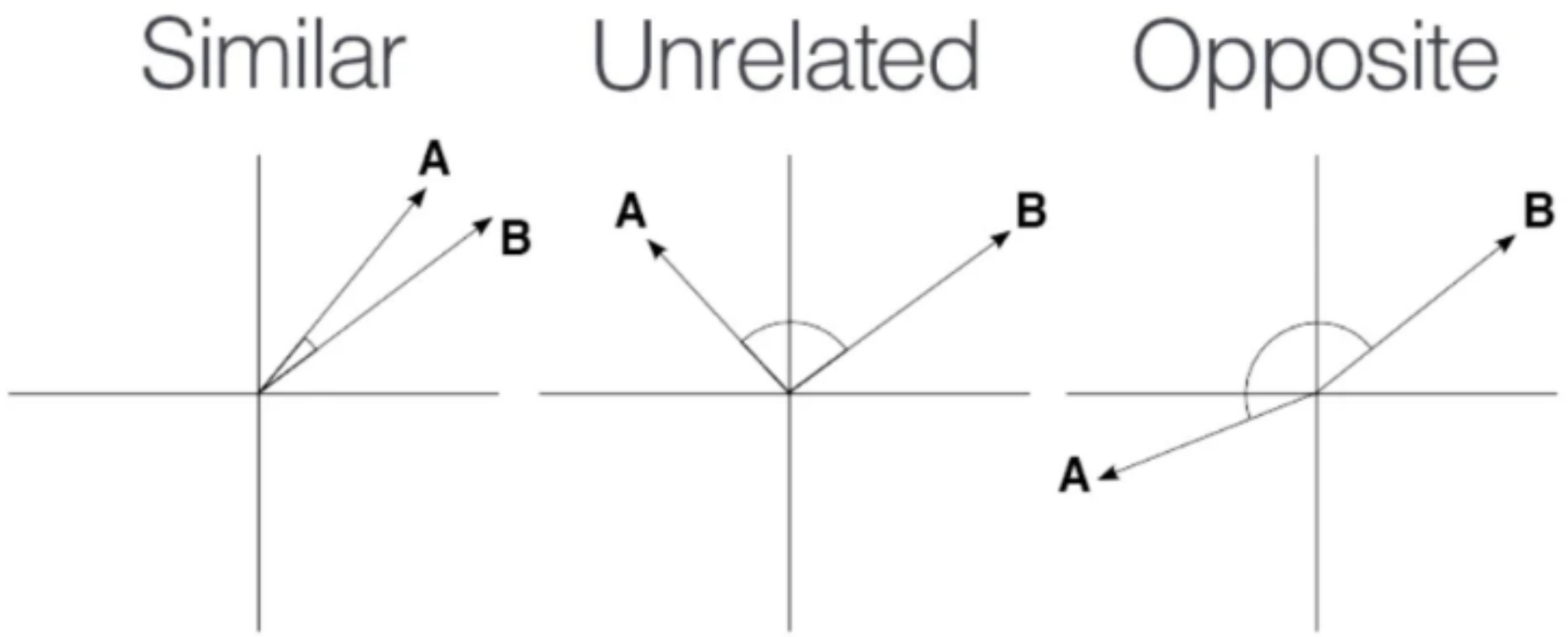
$$\mathcal{L}_m = -\log \frac{\exp(sim(\mathbf{c}_t, \mathbf{q}_t)/\kappa)}{\sum_{\tilde{\mathbf{q}} \sim \mathbf{Q}_t} \exp(sim(\mathbf{c}_t, \tilde{\mathbf{q}})/\kappa)} \qquad sim(\mathbf{a}, \mathbf{b}) = \mathbf{a}^T \mathbf{b} / \|\mathbf{a}\| \|\mathbf{b}\|$$

Wav2Vec, Data2Vec (2018-2023)

$$cos(\theta) = \frac{a \cdot b}{||a|| \cdot ||b||}$$

Since the $cos(\theta)$ value is in the range $[-1, 1]$:

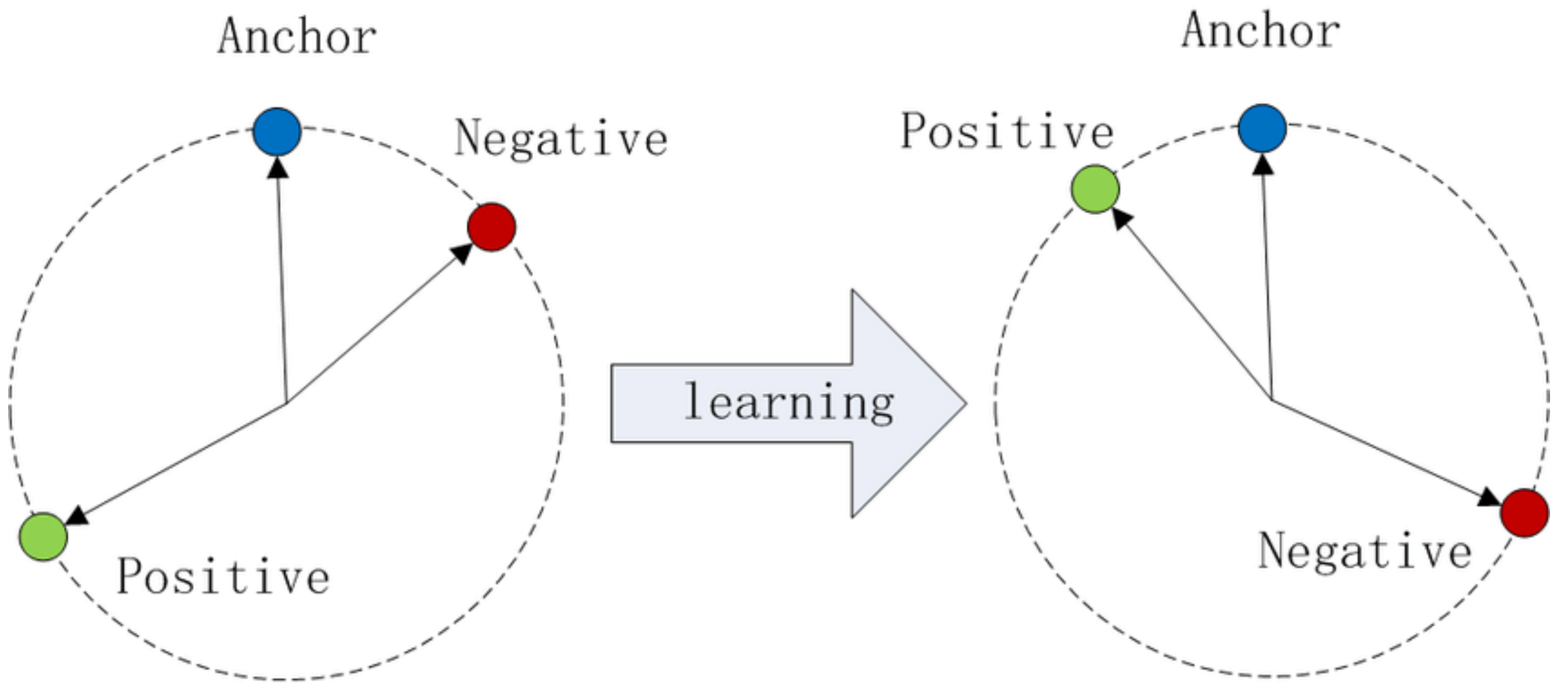
- -1 value will indicate strongly opposite vectors
- 0 independent (orthogonal) vectors
- 1 similar (positive co-linear) vectors. Intermediate values are used to assess the degree of similarity.



Pre-training

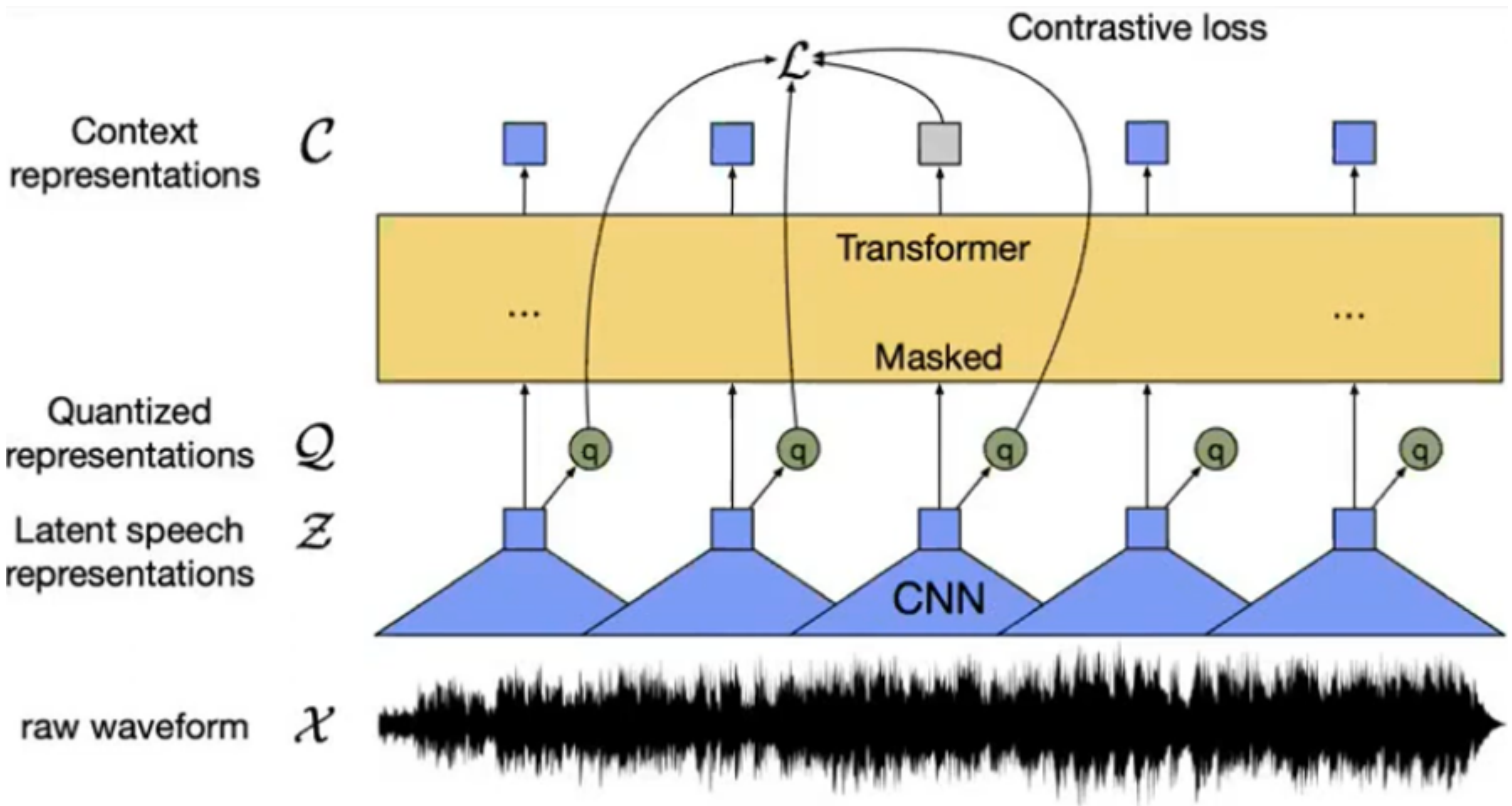
$$\mathcal{L}_m = -\log \frac{\exp(sim(\mathbf{c}_t, \mathbf{q}_t)/\kappa)}{\sum_{\tilde{\mathbf{q}}\sim\mathbf{Q}_t} \exp(sim(\mathbf{c}_t, \tilde{\mathbf{q}})/\kappa)} \qquad sim(\mathbf{a},\mathbf{b}) = \mathbf{a}^T\mathbf{b}/\|\mathbf{a}\|\|\mathbf{b}\|$$

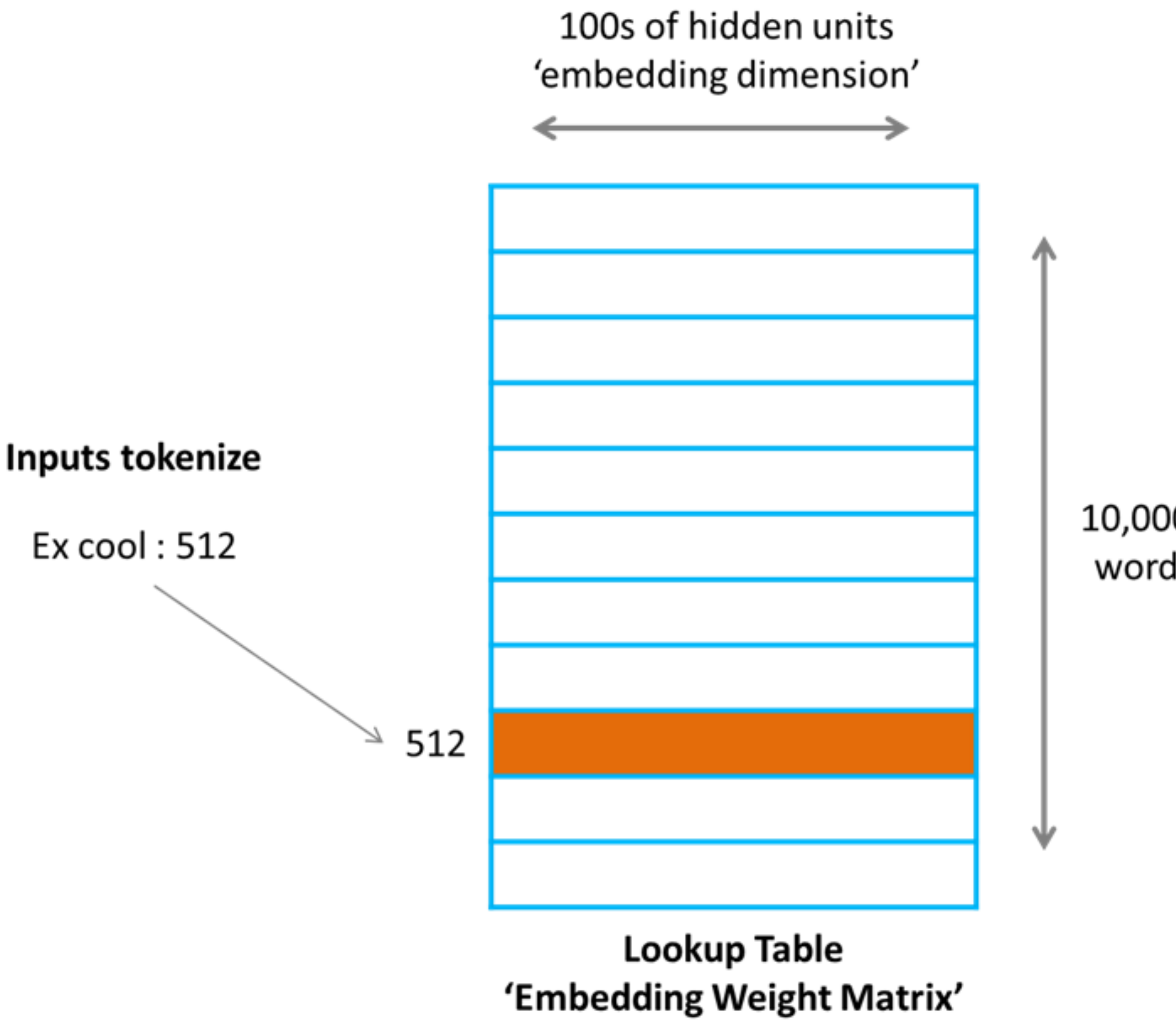
Wav2Vec, Data2Vec (2018-2023)



$$Loss = \sum_{i=1}^{N} \left[\| f_i^a - f_i^p \|_2^2 - \| f_i^a - f_i^n \|_2^2 + \alpha \right]_+$$

Wav2Vec Data2Vec

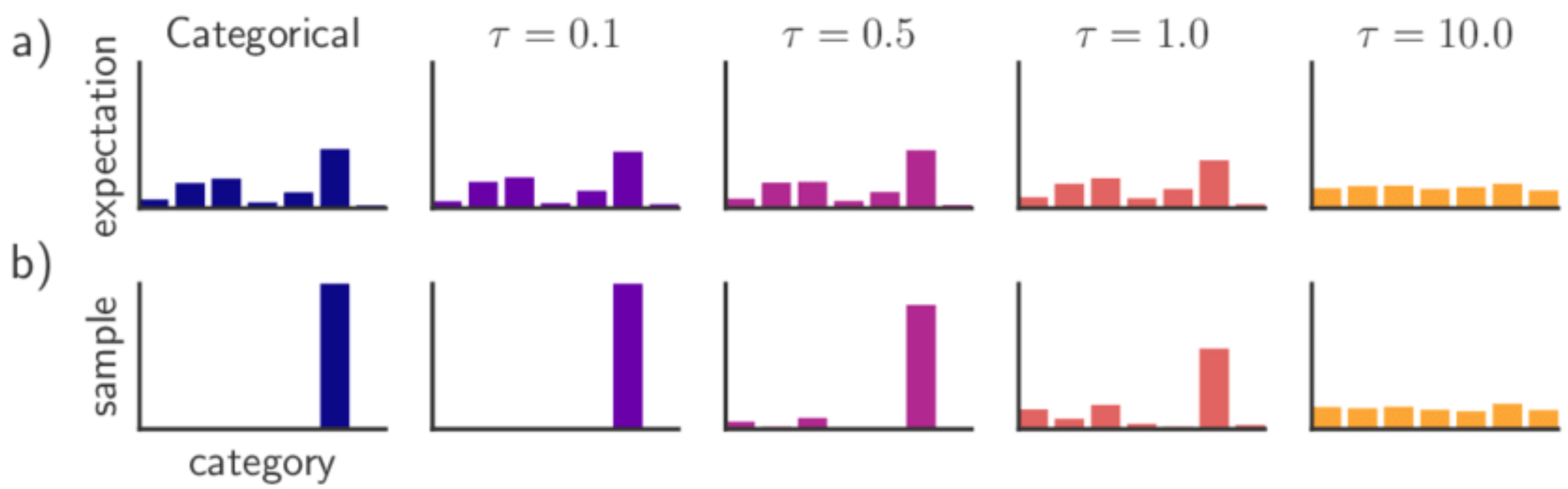


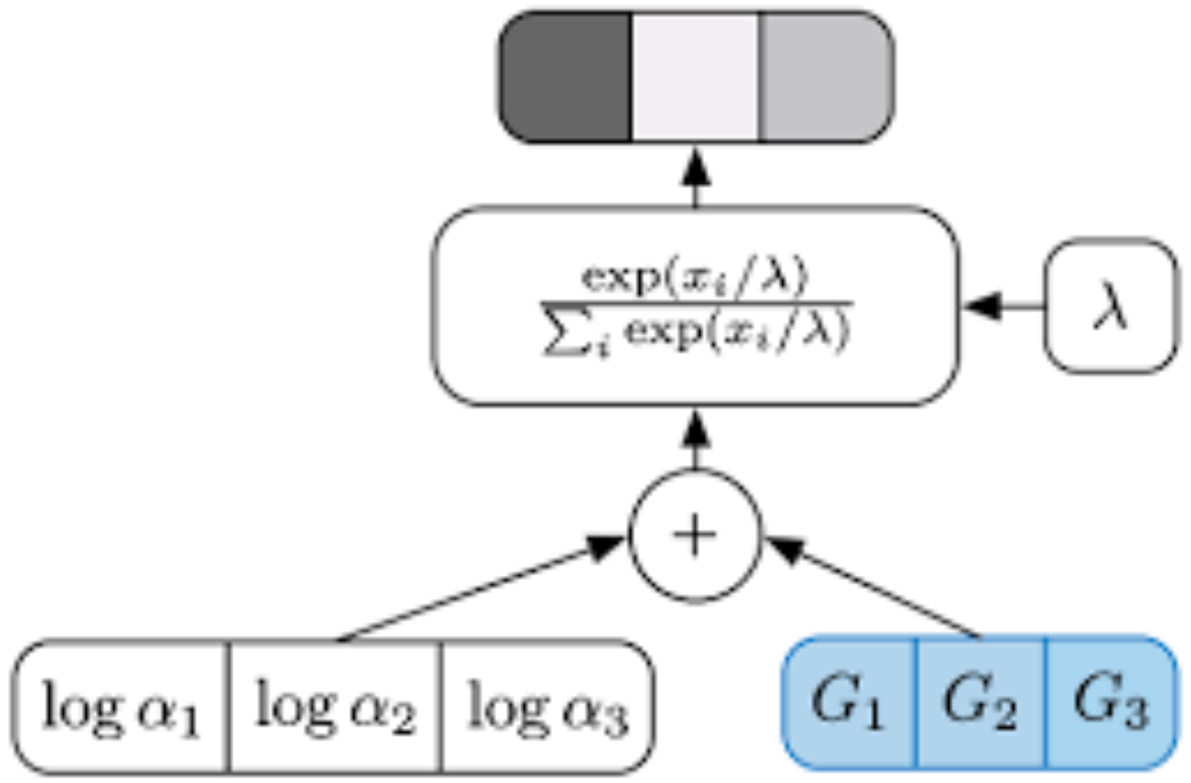


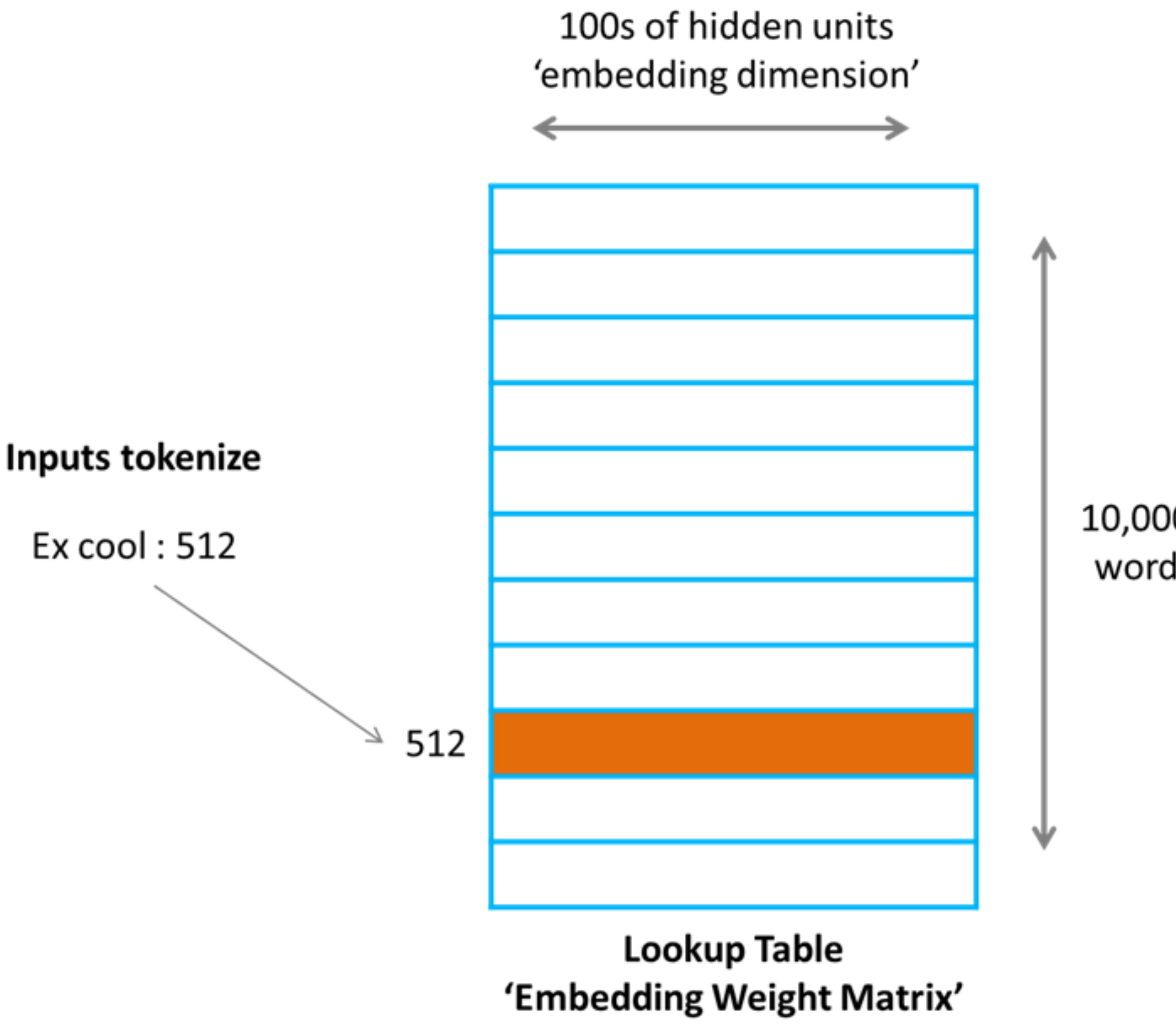
$$p_{g,v} = \frac{\exp(l_{g,v} + n_v)/\tau}{\sum_{k=1}^{V} \exp(l_{g,k} + n_k)/\tau}$$

probability of choosing the v-tł

Wav2Vec Data2Vec



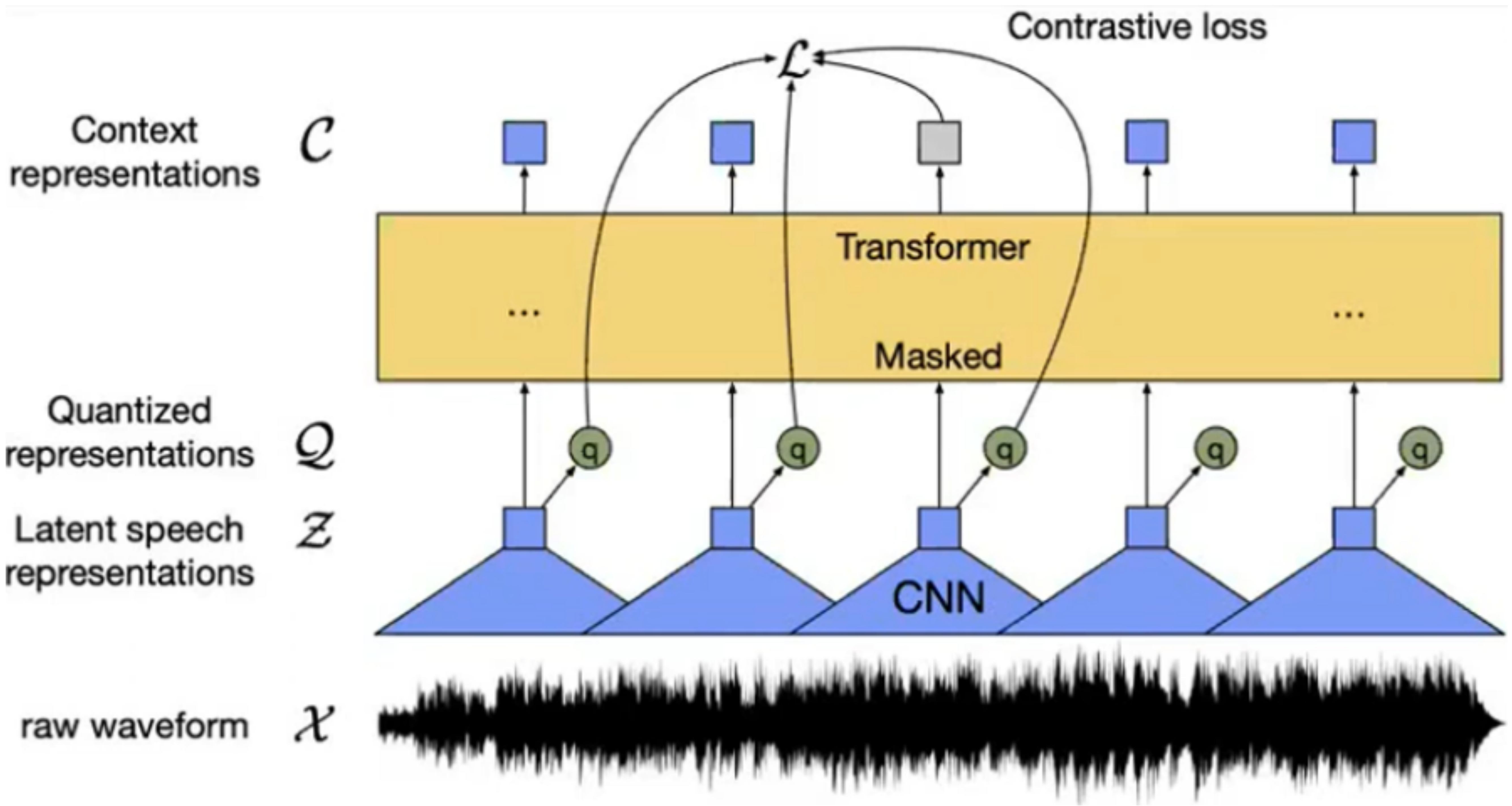




$$p_{g,v} = \frac{\exp(l_{g,v} + n_v)/\tau}{\sum_{k=1}^{V} \exp(l_{g,k} + n_k)/\tau}$$

probability of choosing the v-th

Wav2Vec 2.0, Data2Vec (2018-2023)



Research Opportunity:
Sample mining
Contrastive loss

Pre-training

$$\mathcal{L}_m = -\log \frac{\exp(sim(\mathbf{c}_t, \mathbf{q}_t)/\kappa)}{\sum_{\tilde{\mathbf{q}} \sim \mathbf{Q}_t} \exp(sim(\mathbf{c}_t, \tilde{\mathbf{q}})/\kappa)} \qquad sim(\mathbf{a}, \mathbf{b}) = \mathbf{a}^T \mathbf{b} / \|\mathbf{a}\| \|\mathbf{b}\|$$

$$\mathcal{L}_d = \frac{1}{GV} \sum_{g=1}^{G} -H(\bar{p}_g) = \frac{1}{GV} \sum_{g=1}^{G} \sum_{v=1}^{V} \bar{p}_{g,v} \log \bar{p}_{g,v}$$

Wav2Vec Data2Vec

	dev	test
CNN + TD-filterbanks (Zeghidour et al., 2018a)	15.6	18.0
Li-GRU + MFCC (Ravanelli et al., 2018)	–	16.7 $\pm$ 0.26
Li-GRU + FBANK (Ravanelli et al., 2018)	–	15.8 $\pm$ 0.10
Li-GRU + fMLLR (Ravanelli et al., 2018)	–	14.9 $\pm$ 0.27
Baseline	16.9 $\pm$ 0.15	17.6 $\pm$ 0.11
wav2vec (Librispeech 80h)	15.5 $\pm$ 0.03	17.6 $\pm$ 0.12
wav2vec (Librispeech 960h)	13.6 $\pm$ 0.20	15.6 $\pm$ 0.23
wav2vec (Librispeech + WSJ)	**12.9 $\pm$ 0.18**	**14.7 $\pm$ 0.42**

Table 2: Results for phoneme recognition on TIMIT in terms of PER. All our models use the CNN-8L-PReLU-do0.7 architecture (Zeghidour et al., 2018a).

Similar model approches (Quantizing Z vectors)

VQ-GAN
VQ-VAE

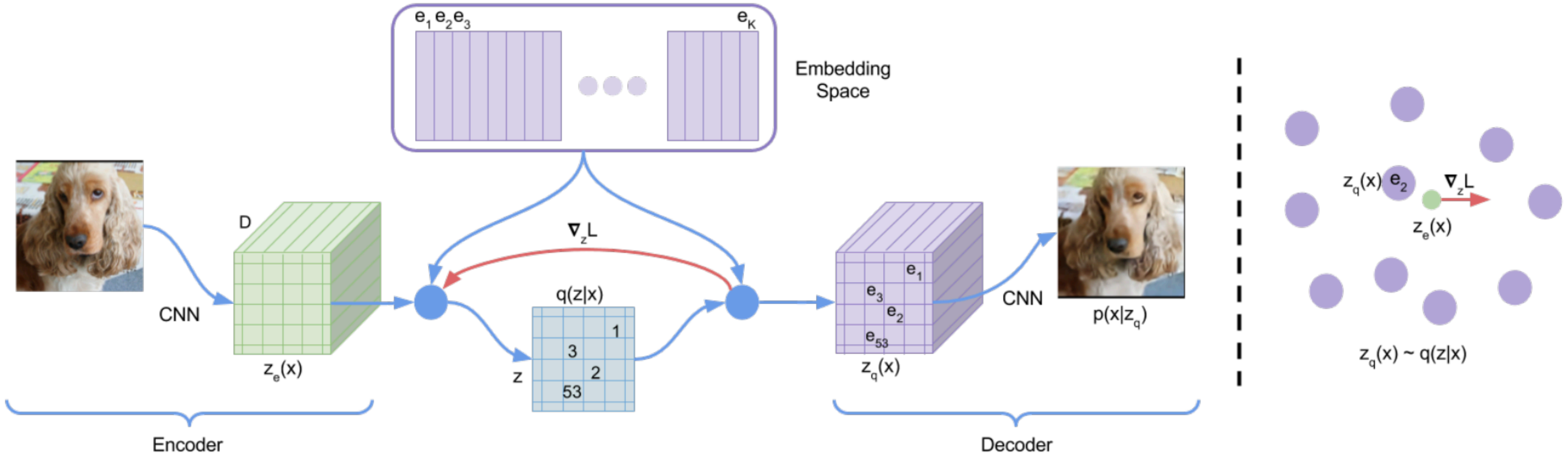
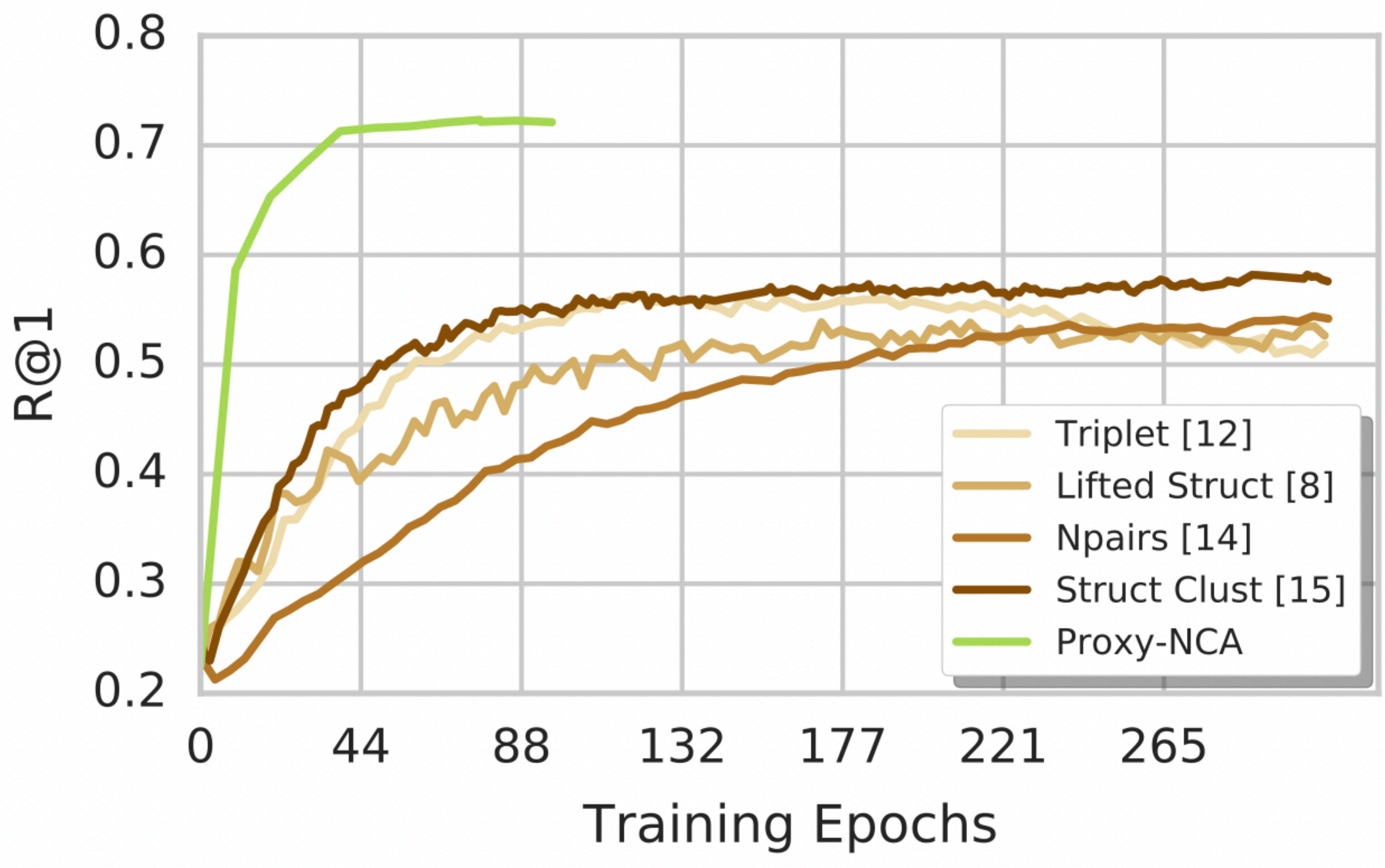


Figure 1: Left: A figure describing the VQ-VAE. Right: Visualisation of the embedding space. The output of the encoder $z(x)$ is mapped to the nearest point e_2. The gradient $\nabla_z L$ (in red) will push the encoder to change its output, which could alter the configuration in the next forward pass.

Similar model approches (Temporary Variable)

Proxy NCA, Proxy Ranking Loss



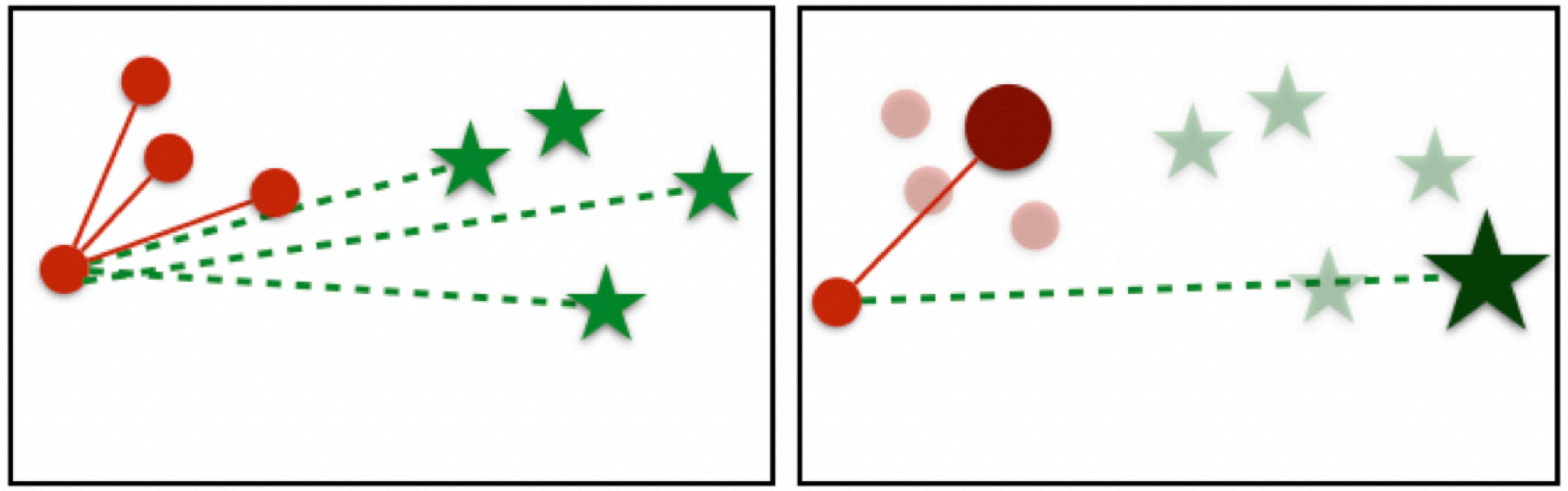


Figure 2: Illustrative example of the power of proxies. [Left panel] There are 48 triplets that can be formed from the instances (small circles/stars). [Right panel] Proxies (large circle/star) serve as a concise representation for each semantic concept, one that fits in memory. By forming triplets using proxies, only 8 comparisons are needed.

Similar model approches (Loss on Z-Vectors)

Denoising diffusion models

- **Forward / noising process**
  - Sample data $p(\mathbf{x}_0)$ → turn to noise

$p_0(\mathbf{x}_0)$

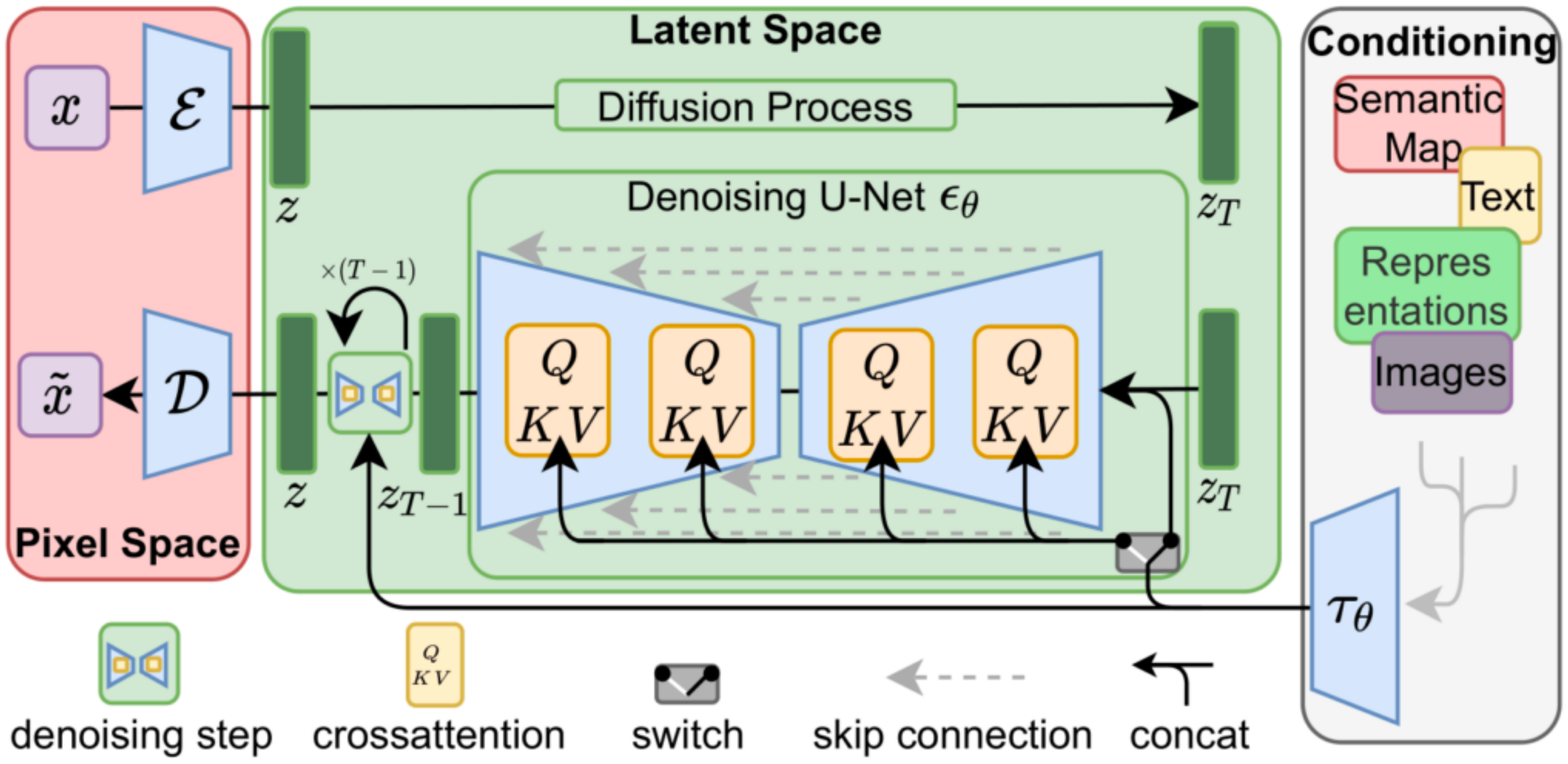
$p_T(\mathbf{x}_T) \sim \mathcal{N}(0, I$

Clean sample $\mathbf{x}_0$ $\mathbf{x}_1$ $\mathbf{x}_{T-1}$ $\mathbf{x}_T$ *Pure noise*

- **Reverse / denoising process**
  - Sample noise $p_T(\mathbf{x}_T)$ → turn into data

Similar model approches (Loss on Z-Vectors)

Stable Diffusion



Time-Series Training Tricks (Fine Tunning)

Scheduled Sampling for Sequence Prediction with Recurrent Neural Networks

Table 2: F1 score (the higher the better) on the validation set of the parsing task.

Approach	F1
Baseline LSTM	86.54
Baseline LSTM with Dropout	87.0
Always Sampling	-
Scheduled Sampling	**88.08**
Scheduled Sampling with Dropout	**88.68**

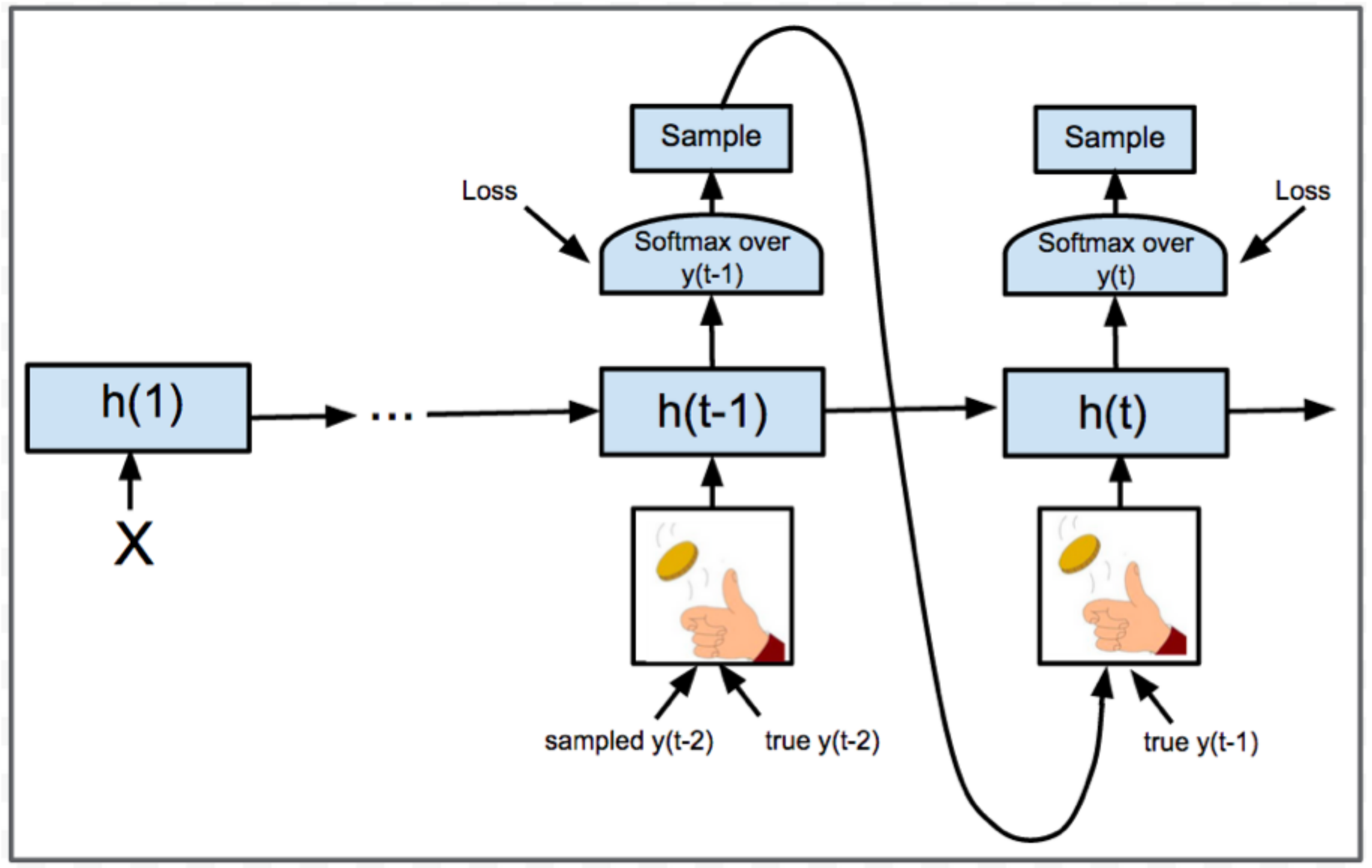
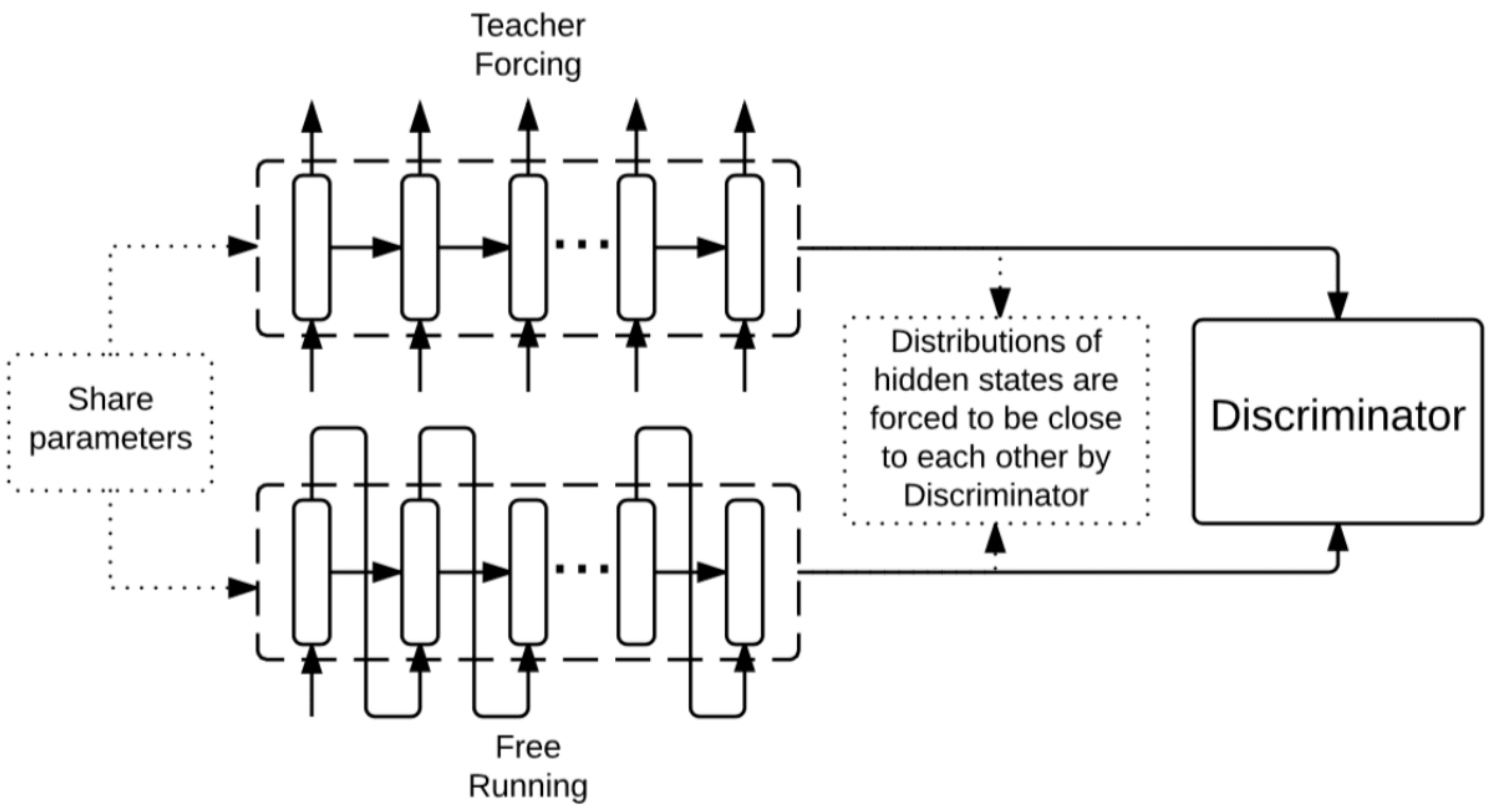
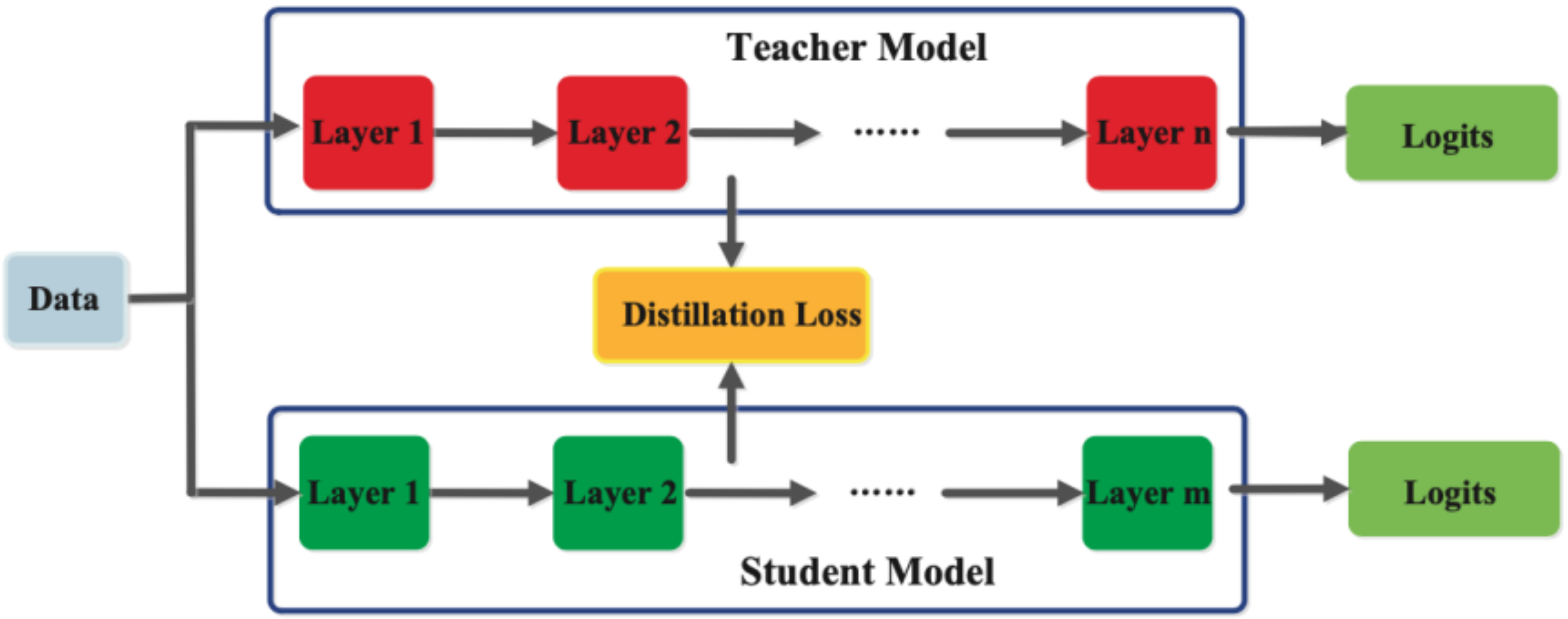


Figure 1: Illustration of the Scheduled Sampling approach, where one flips a coin at every time step to decide to use the true previous token or one sampled from the model itself.

Time-Series Training Tricks (Fine Tunning)

Professor forcing, Distillation

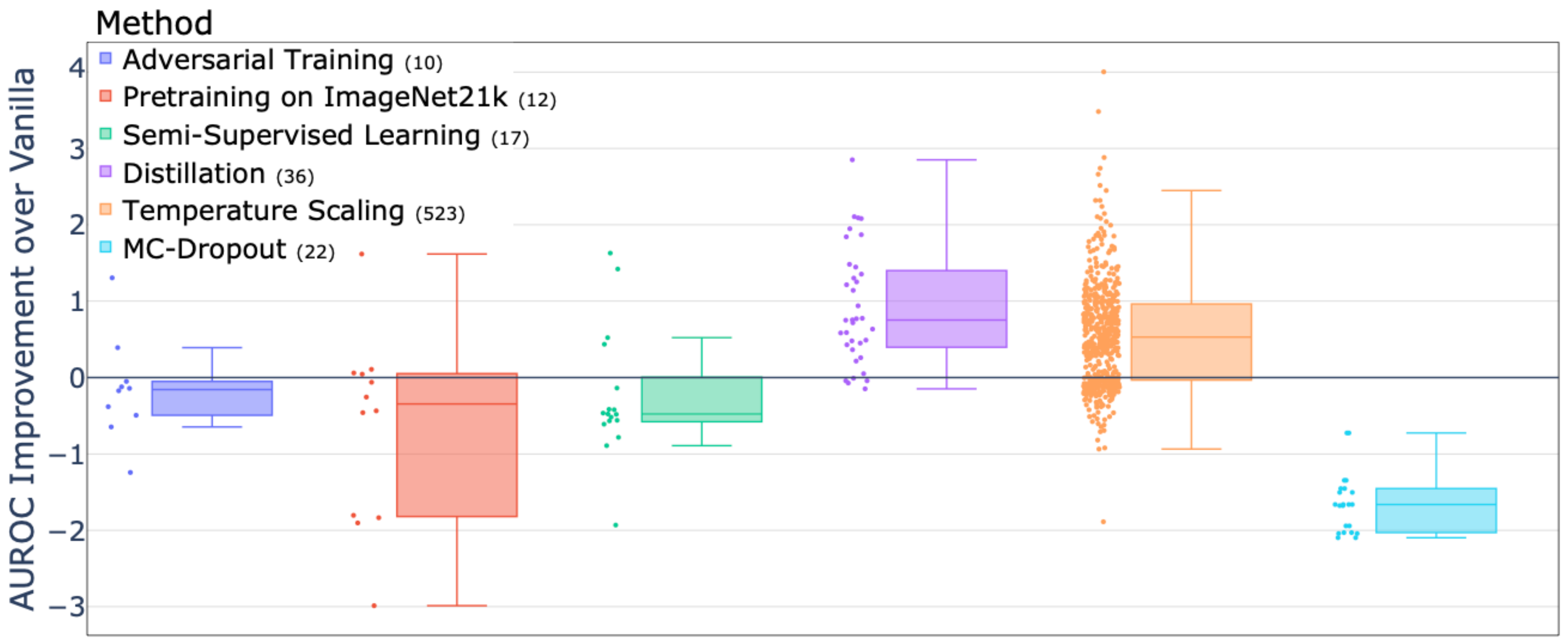




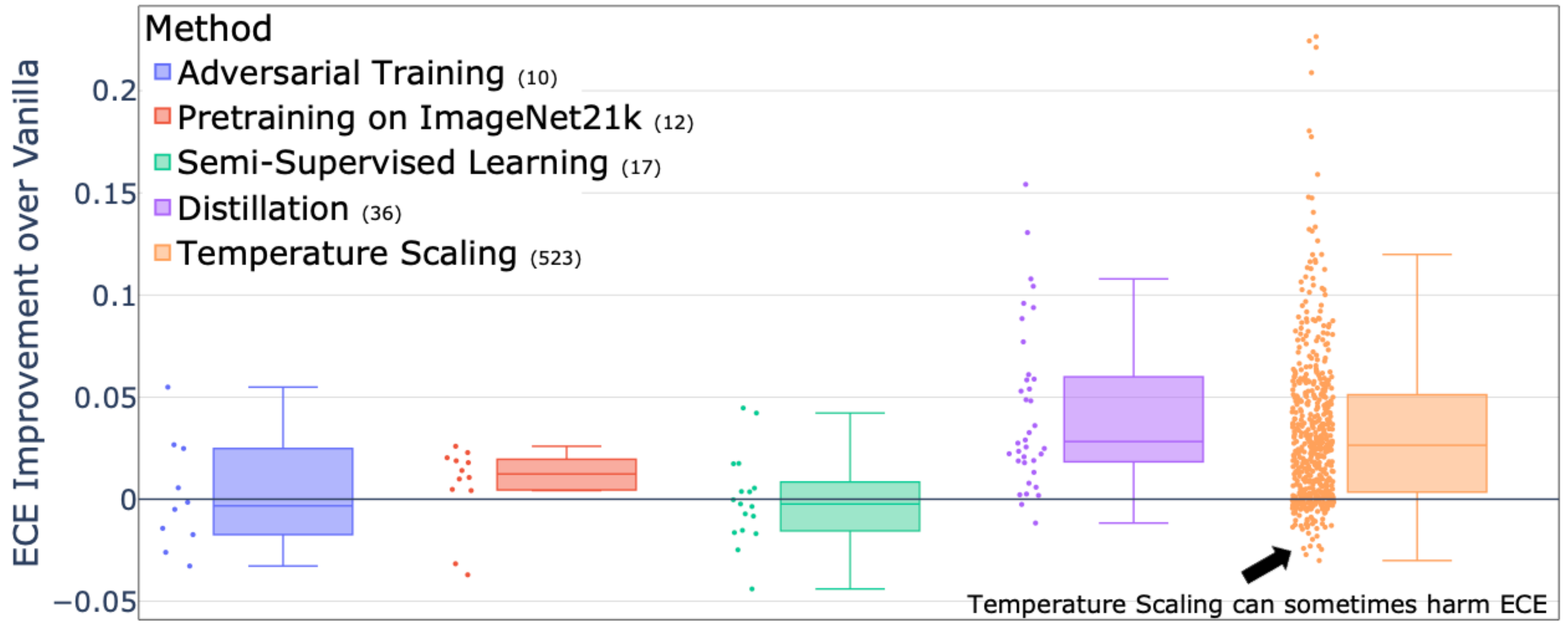
Time-Series Training Tricks (Fine Tunning)

Professor forcing, Distillation

WHAT CAN WE LEARN FROM THE SELECTIVE PREDICTION AND UNCERTAINTY ESTIMATION PERFORMANCE OF 523 IMAGENET CLASSIFIERS?



(a)



(b)

Figure 4: A comparison of different methods and their improvement in terms of (a) AUROC and (b) ECE, relative to the same model's performance without employing the method. Markers above the x-axis represent models that benefited from the evaluated method, and vice versa. The numbers in the legend to the right of each method indicate the number of pairs compared. Temperature scaling can sometimes harm ECE, even though its purpose is to improve it.

Dr. Evalds Urtans
evalds@asya.ai