Deep Learning School

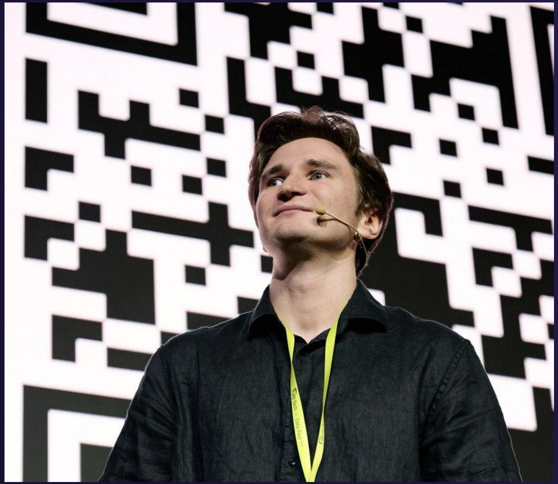
бесплатно.

онлайн.

фундаментально.

2024

DLS



Self-Supervised Learning in Audio

3rd week

Razvorotnev Ivan

MLE in Audio Team, Zvuk
HSE and Skoltech graduate

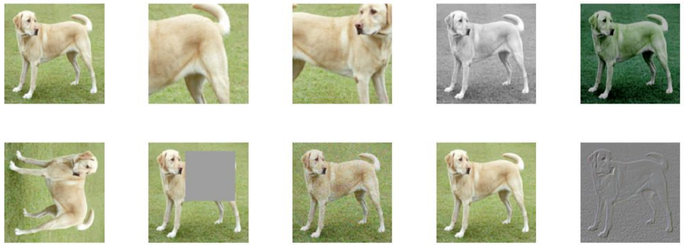
What is Self-Supervised Learning (SSL)?

DLS

No labels? — use cheap domain knowledge to generate pretext tasks with pseudo labels

Computer Vision

Augment image x to get different views x_i^+:

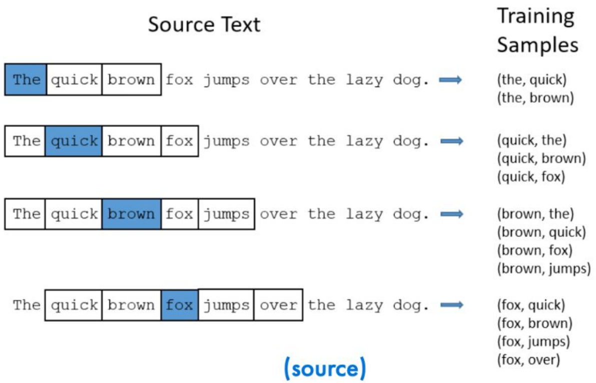


(Chen et al., 2020a)

Sample negatives x_k^-

Attract positives, repel negatives

Natural Language Processing



(source)

Sample context as words around position

Learn to predict context

Why Self-supervised Learning?

- Huge amount of unlabeled data
- Labeling is expensive and error-prone
- Solve many tasks at once
- Improved downstream task performance through feature generalization

Why Self-supervised Learning?

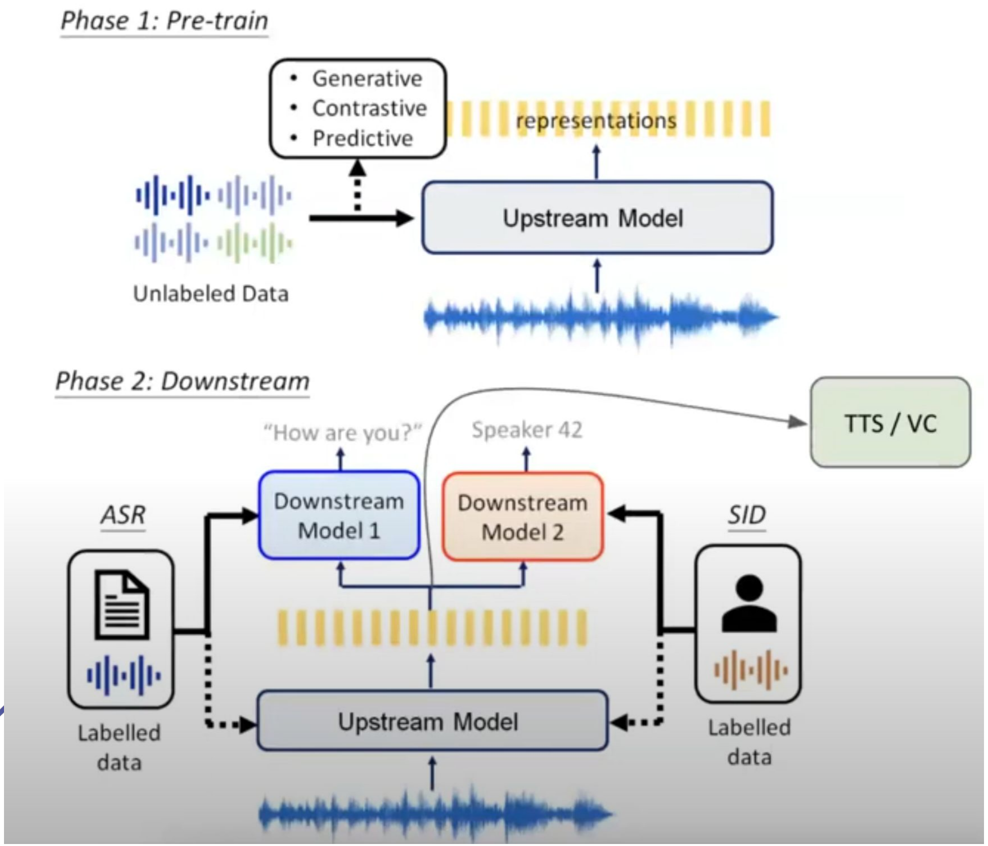
Storing knowledge from solving one problem and applying it to a different problem.

Pretraining on Imagenet-1k, evaluation on 12 datasets

	Food	CIFAR10	CIFAR100	Birdsnap	SUN397	Cars	Aircraft	VOC2007	DTD	Pets	Caltech-101	Flowers
Linear Eval												
SimCLR	**76.9**	**95.3**	80.2	48.4	**65.9**	60.0	61.2	**84.2**	**78.9**	89.2	**93.9**	**95.0**
Supervised	75.2	**95.7**	**81.2**	**56.4**	64.9	**68.8**	**63.8**	83.8	**78.7**	**92.3**	**94.1**	94.2
Fine-tuned												
SimCLR	**89.4**	**98.6**	**89.0**	**78.2**	**68.1**	**92.1**	**87.0**	**86.6**	**77.8**	92.1	**94.1**	97.6
Supervised	88.7	98.3	**88.7**	**77.8**	67.0	91.4	**88.0**	86.5	**78.8**	**93.2**	**94.2**	**98.0**
Random init	88.3	96.0	81.9	**77.0**	53.7	91.3	84.8	69.4	64.1	82.7	72.5	92.5

(Chen et al., 2020a)

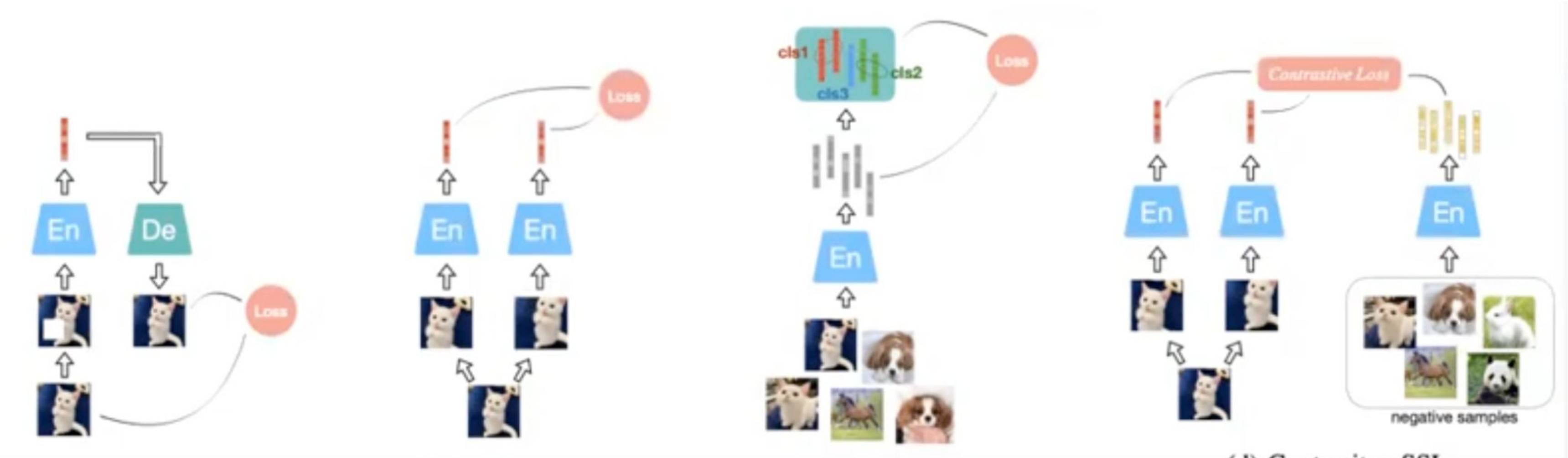
SSL Pipeline



Types of SSL methods:

1. **Predictive** – predict missing or corrupted parts (e.g., BERT, MAE, wav2vec)
2. **Contrastive** – pull positives together, push negatives apart (e.g., SimCLR, MoCo)
3. **Non-contrastive** – learn invariances without negatives (e.g., BYOL, Barlow Twins)
4. **Clustering / Prototype** – group features into clusters, predict assignments (e.g., SwAV, DINO)
5. **Generative** – generate realistic data (e.g., GANs, diffusion, autoregressive models)
6. **Cross-modal** – align different modalities (e.g., CLIP, audio-text, video-text)

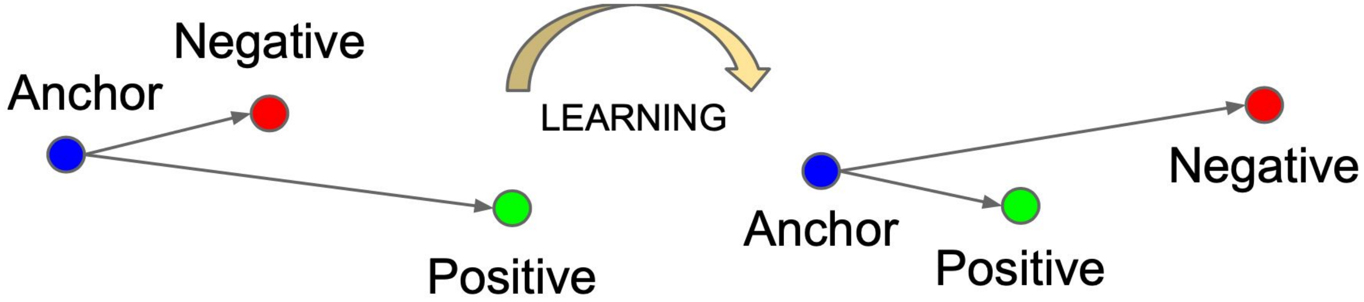
SSL Framework



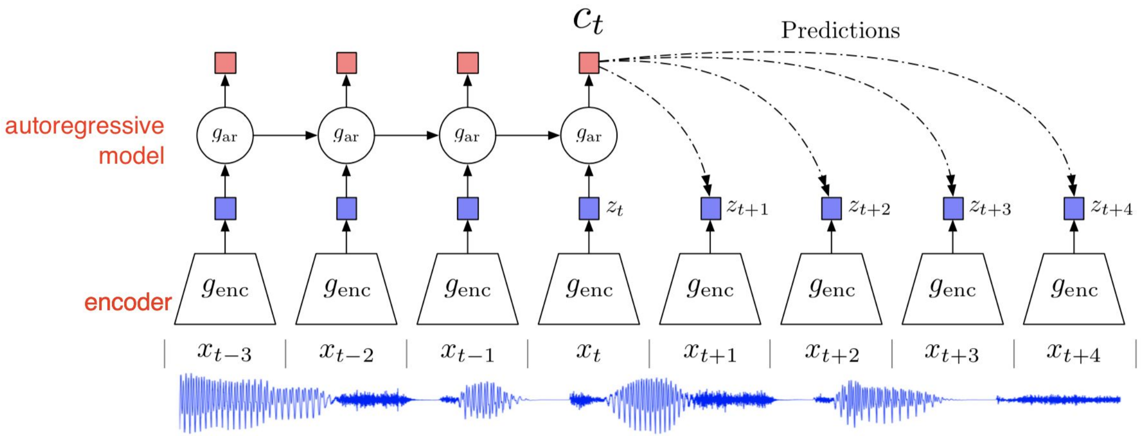
(a) Auto-encoding (b) Siamese Network (c) Clustering (d) Contrasitve SSL

Predictive SSL framework (a), contrastive (d),
non-contrastive (d), clustering (c)

Contrastive Learning



Contrastive Predictive Coding(CPC)



Given context c, predict observation x **without** directly modelling conditional $p(x|c)$

Maximally preserve MI between x and c: $I(x, c) = \sum_{x,c} p(x, c) \log \frac{p(x|c)}{p(x)}$

CPC (Oord et al., 2018):

1. Encode $z_t = g_{\text{enc}}(x_t)$; summarize context $c_t = g_{\text{ar}}(z_{\leq t})$
2. Model density ratio $f_k(x_{t+k}, c_t) \propto \frac{p(x_{t+k}|\ c_t)}{p(x_{t+k})}$ as $f_k(x_{t+k}, c_t) := \exp(z_{t+k}^\top W_k c_t)$
3. Noise-Contrastive Estimation:

$$\mathcal{L}_{\text{InfoNCE}} = -\mathbb{E}_X \left[\log \frac{f_k(x_{t+k}, c_t)}{\sum_{x_j \in X} f_k(x_j, c_t)} \right]$$

InfoNCE loss

The InfoNCE loss optimizes the negative log probability of classifying the positive sample correctly

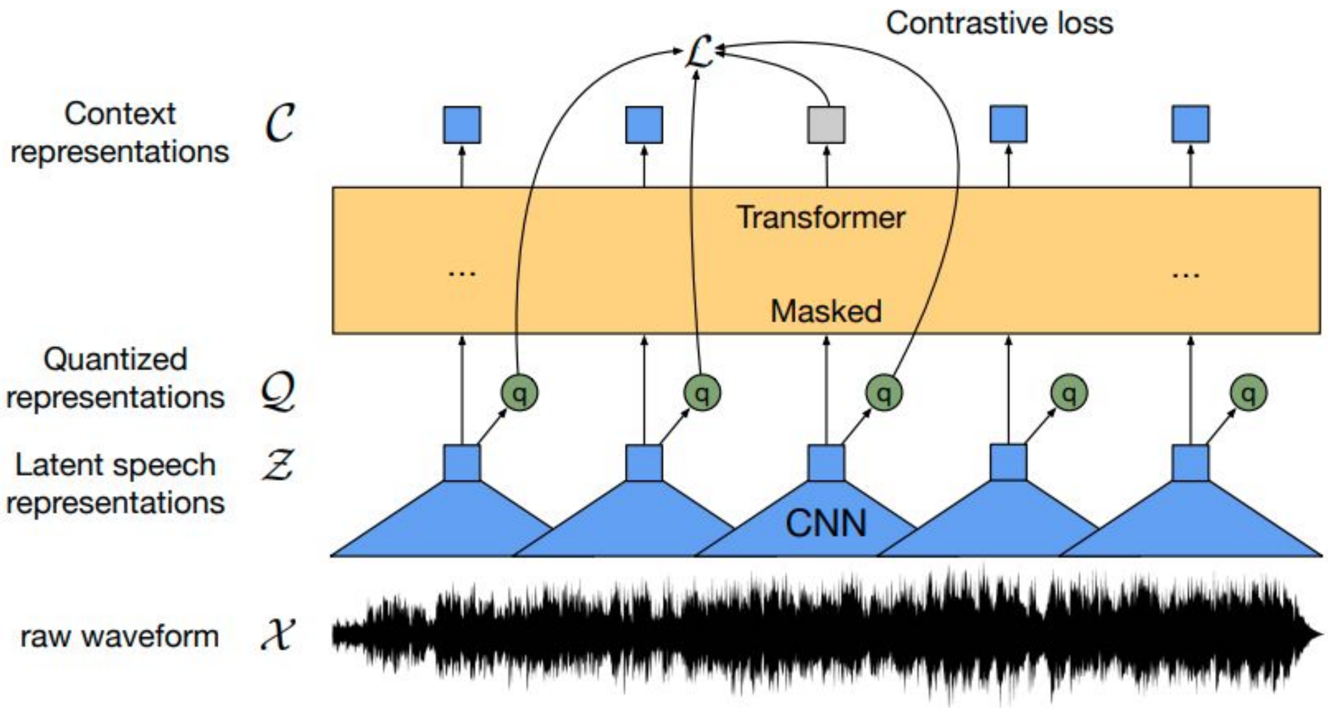
The probability of detecting the positive sample correctly is:

$$p(C = \text{pos} \mid X, \mathbf{c}) = \frac{p\left(x_{\text{pos}} \mid \mathbf{c}\right) \prod_{i=1,\ldots,N; i \neq \text{pos}} p\left(\mathbf{x}_i\right)}{\sum_{j=1}^{N} \left[p\left(\mathbf{x}_j \mid \mathbf{c}\right) \prod_{i=1,\ldots,N; i \neq j} p\left(\mathbf{x}_i\right)\right]} =$$

$$= \frac{\frac{p(\mathbf{x}_{\text{pos}} \mid \mathbf{c})}{p(\mathbf{x}_{\text{pos}})}}{\sum_{j=1}^{N} \frac{p(\mathbf{x}_j \mid \mathbf{c})}{p(\mathbf{x}_j)}} = \frac{f\left(\mathbf{x}_{\text{pos}}, \mathbf{c}\right)}{\sum_{j=1}^{N} f\left(\mathbf{x}_j, \mathbf{c}\right)}$$

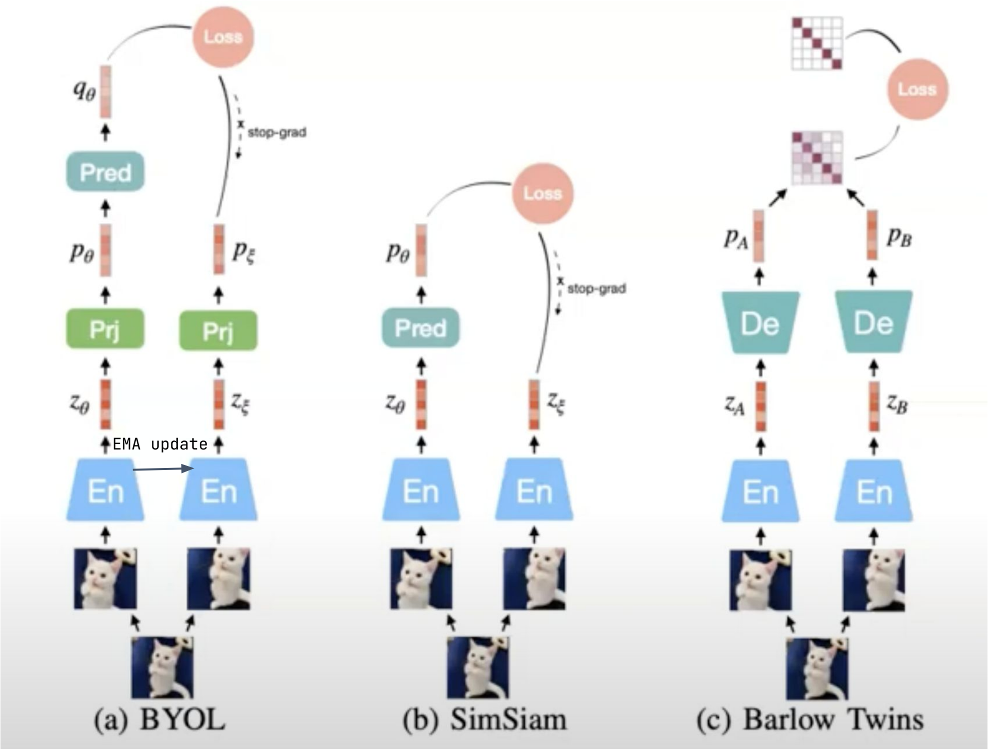
where the scoring function is $f(\mathbf{x}, \mathbf{c}) \propto \frac{p(\mathbf{x} \mid \mathbf{c})}{p(\mathbf{x})}$.

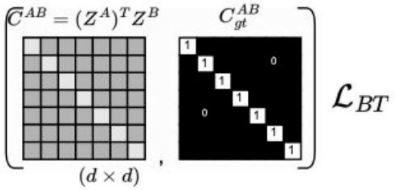
Wav2Vec 2.0



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

Non-Contrastive Learning





Why BYOL and SimSiam don't collapse?

Update online network θ at each training step, use EMA updates for target network ξ:

$$\theta \leftarrow \text{optimizer}\left(\theta, \nabla_\theta \mathcal{L}_{\theta,\xi}^{\text{BYOL}}, \eta\right),$$
$$\xi \leftarrow \tau\xi + (1-\tau)\theta,$$

Intuitions behind absense of collapse:

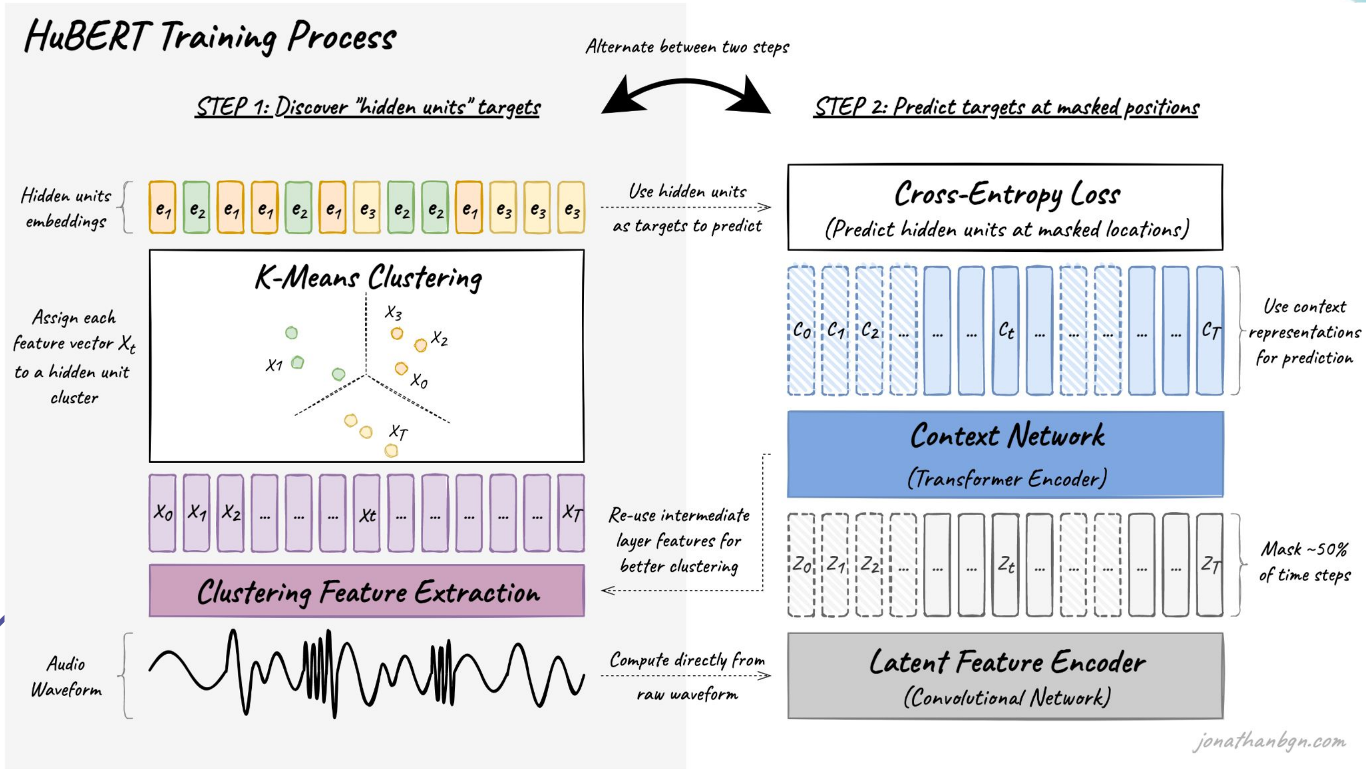
- ξ updates are **not** in the direction of $\nabla_\xi \mathcal{L}_{\theta,\xi}^{\text{BYOL}}$
- Collapsed constant solutions are unstable due to variance induced by asymmetric design / training dynamics

BYOL

- With **stop-grad**, the trivial solution has **zero gradient** w.r.t. encoder weights, but it's a **saddle point, not a stable minimum.**
- Any noise or SGD fluctuation pushes the model away, and the predictor-stopgrad asymmetry amplifies differences instead of collapsing them.
- The predictor plays the same role as BYOL's EMA target: introducing an asymmetry so the system can't just synchronize into trivial constant vectors.

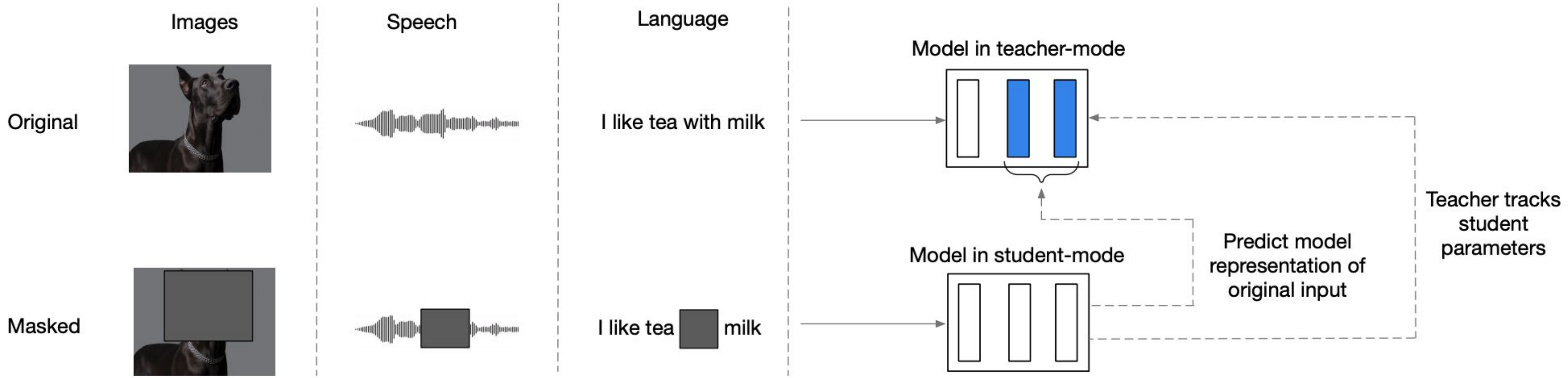
SimSiam

Predictive SSL. HuBERT

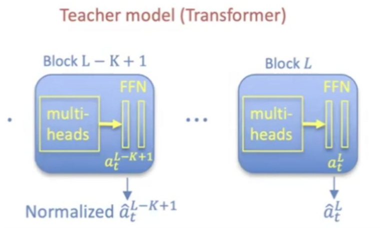


Data2Vec

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

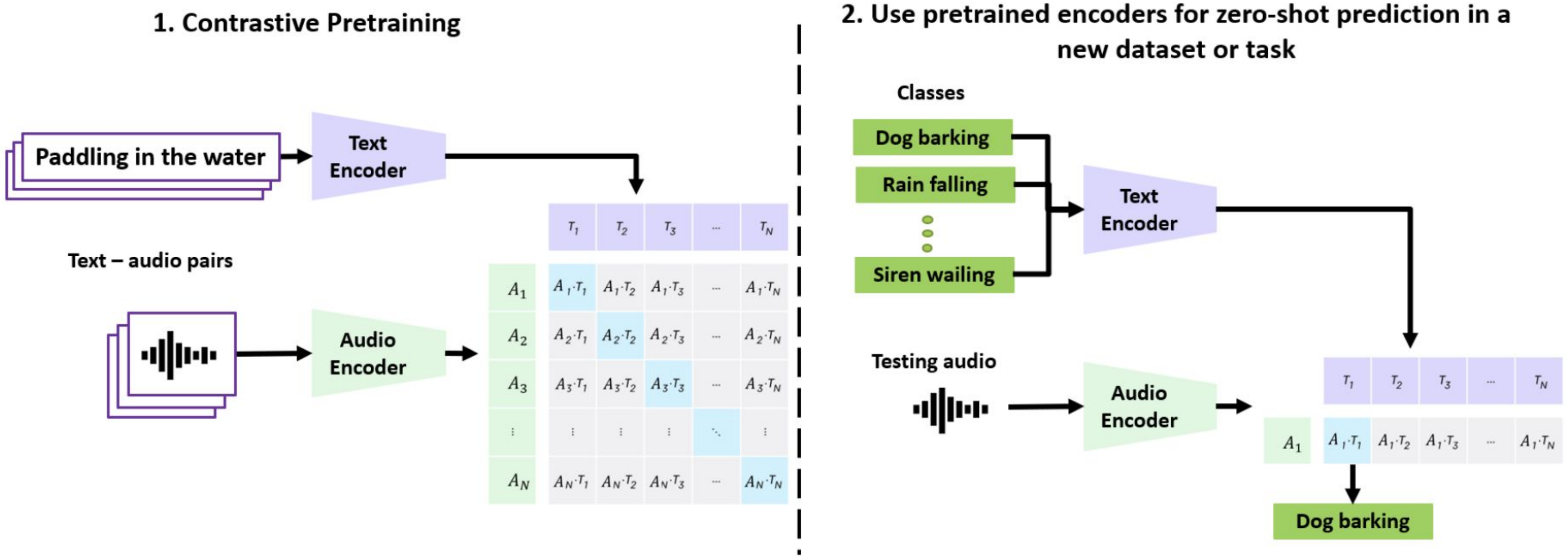


Trains by masking parts of the input and forcing a student network to regress the continuous latent representations produced by an EMA teacher

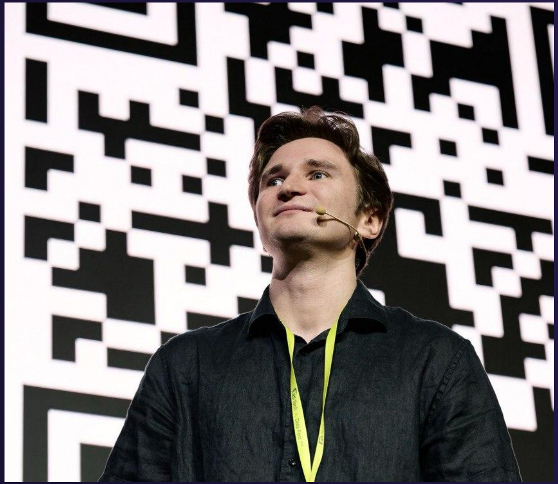


$$y_t = \frac{1}{K}\sum_{l=L-K+1}^{L} \hat{a}_t^l$$

Cross-Model learning



DLS



Razvorotnev Ivan

Thank you for attention!

tg: @razvor