# Representation Learning with Contrastive Predictive Coding

School of Industrial and Management Engineering, Korea University

Sangmin Kim

KOREA UNIVERSITY

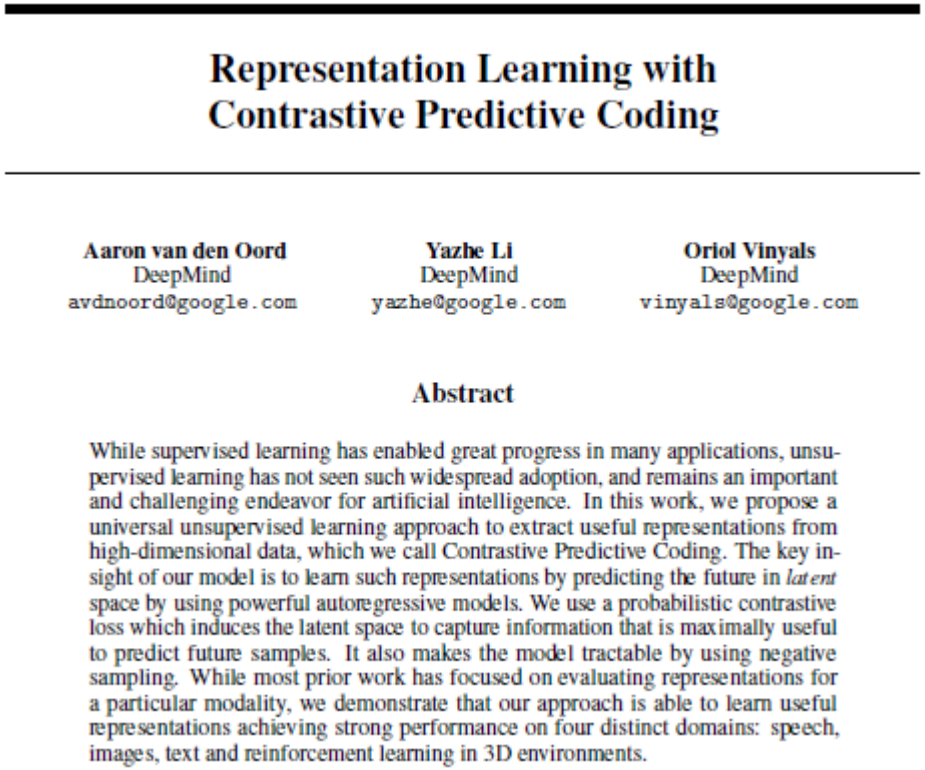
DMQA

# Contents

# Contrastive Predictive Coding

❖ **CPCv1**: Representation Learning with Contrastive Predictive Coding(2018, arXiv)

- Google DeepMind에서 연구된 논문이며, 2022년 5월 11일 기준 1,972회 인용됨



**Representation Learning with Contrastive Predictive Coding**

**Aaron van den Oord**
DeepMind
avdnoord@google.com

**Yazhe Li**
DeepMind
yazhe@google.com

**Oriol Vinyals**
DeepMind
vinyals@google.com

**Abstract**

While supervised learning has enabled great progress in many applications, unsupervised learning has not seen such widespread adoption, and remains an important and challenging endeavor for artificial intelligence. In this work, we propose a universal unsupervised learning approach to extract useful representations from high-dimensional data, which we call Contrastive Predictive Coding. The key insight of our model is to learn such representations by predicting the future in *latent* space by using powerful autoregressive models. We use a probabilistic contrastive loss which induces the latent space to capture information that is maximally useful to predict future samples. It also makes the model tractable by using negative sampling. While most prior work has focused on evaluating representations for a particular modality, we demonstrate that our approach is able to learn useful representations achieving strong performance on four distinct domains: speech, images, text and reinforcement learning in 3D environments.

# Contrastive Predictive Coding

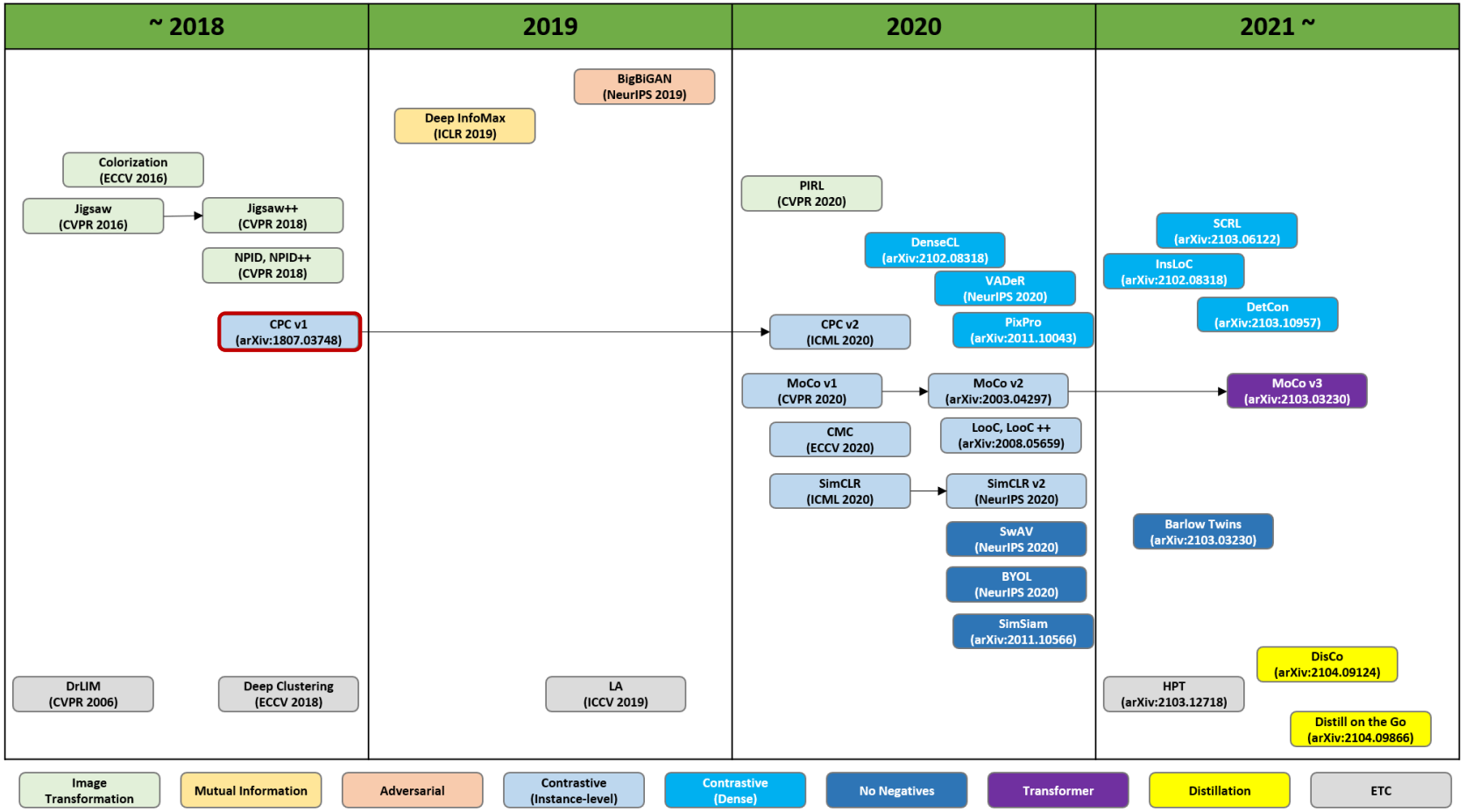
## ❖ Brief summary

- Contrastive Predictive Coding (CPC) is a **unsupervised learning approach to extract useful representations from high-dimensional data**
- **A noise-Contrastive Estimation Loss(InfoNCE Loss)** induces the latent space to capture information that is maximally useful to predict future samples
- **Four distinct domains are used for evaluation: speech, images, text and reinforcement learning** in 3D environments

# Contrastive Predictive Coding

## ❖ Chronicle

- Contrastive learning 계열 기점이 되는 논문으로 향후, 다양한 도메인에 적용됨



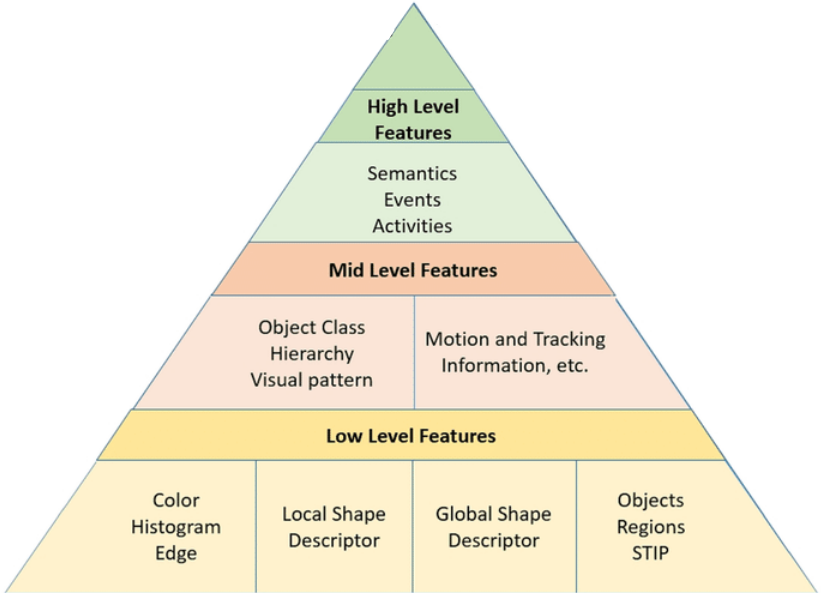



DMQA

# Research Purpose

## ❖ Introduction

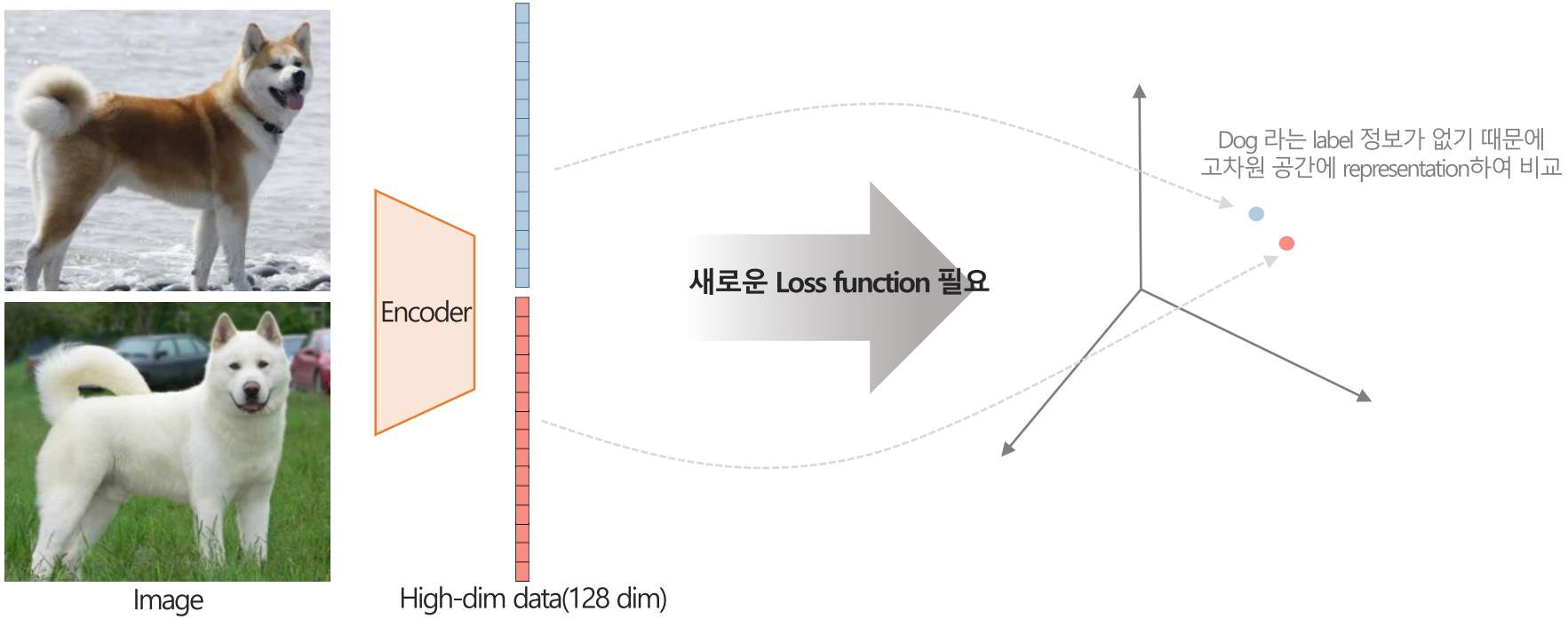
- Representation learning은 해결하는데, less specialized (=robust and generic representation = high-level representation=high-dimensional) feature 를 필요로 함
- 예를 들어, 이미지 분류(single supervised task)를 위해 pre-training된 모델의 feature들은 분류를 위한 feature(= specialized )일 뿐, 분류와 무관한 image captioning 에 필요한 feature (= specialized ) 정보는 포함하지 않음
- 즉, pre-training을 통해 여러 task를 가능케 하는 feature 추출하고자 함



# Research Purpose

## ❖ Motivation & Intuition

- Intuition: 고차원 시그널 속 다른 부분들 사이에 있는 shared information을 인코딩한 representation을 학습하는 것
- 여기서 다른 부분들 사이에 있는 shared information은 high-level information(=slow feature) 을 의미
- 그러나, MSE 나 Cross-entropy 와 같은 unimodal losses 를 사용하면 고차원 정보를 예측하기 어려움
  - 즉, 고차원 data 를 MSE나 Cross-entropy loss 로 비교하면 비효율적, 새로운 loss가 필요함






DMQA

# Background

## ❖ Background

- Predictive Coding
- Mutual Information
- NCELoss

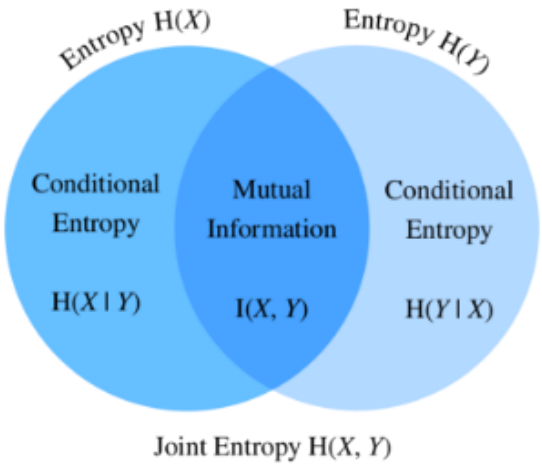
# Background

## ❖ Mutual Information

- 정보 이론의 개념으로 하나의 확률 변수가 다른 하나의 확률 변수에 대해 제공하는 정보의 양
- Joint distribution p(x,y)가 p(x)·p(y) 와 얼마나 비슷한지를 측정하는 척도
- X 와 Y 가 independent 하면, p(x,y)가 p(x)·p(y) 같아지므로 log 내 식 값이 1이 되어 Mutual Information 값은 0이 되며,서로 dependent 할수록 Mutual Information 값이 커짐(X와 Y의 순서를 바꾸어도 동일)

Mutual Information

$$\mathbb{I}(X;Y) \triangleq \mathbb{KL}(p(x,y) \| p(x)p(y)) = \sum_{y \in Y} \sum_{x \in X} p(x,y) \log \frac{p(x,y)}{p(x)\,p(y)}$$






DMQA

# Background

## ❖ Predictive Coding

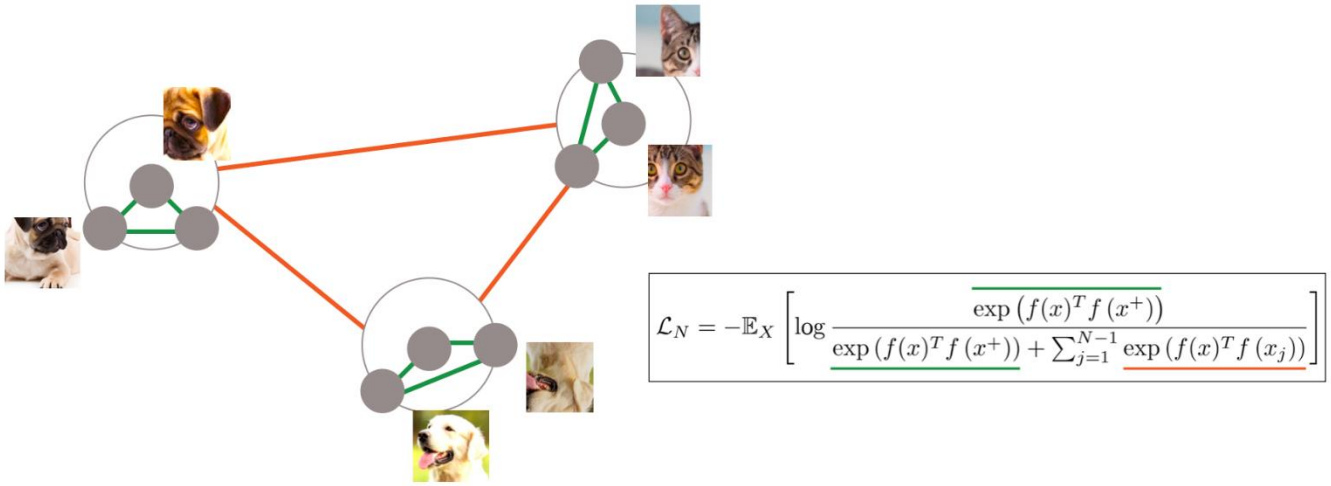
- Predictive Coding 이란 말 그대로 예측 코딩, 예측 처리라고 하며, Neuroscience에서의 개념
- Predictive Coding 이론에 따르면, 인간의 뇌는 여러 추상적인 단계로부터 관측치를 예측
- 예를 들어, 특정 단어를 예측하기 위해 주변 단어를 예측하거나, 회색 이미지를 보고 색상을 예측하거나, 이미지 패치에서 서로의 상대적인 위치를 활용하는 경우

Representation Learning with
Contrastive Predictive Coding

# Background

## ❖ InfoNCE Loss

- Positive sample 끼리 가깝게, negative sample 끼리 멀게 representation 을 정의하기 위해 사용
- 초록색 부분은 최대화, 빨간색 부분은 최소화되도록 학습



$$\mathcal{L}_N = -\mathbb{E}_X \left[ \log \frac{\exp\left(f(x)^T f(x^+)\right)}{\exp\left(f(x)^T f(x^+)\right) + \sum_{j=1}^{N-1} \exp\left(f(x)^T f(x_j)\right)} \right]$$

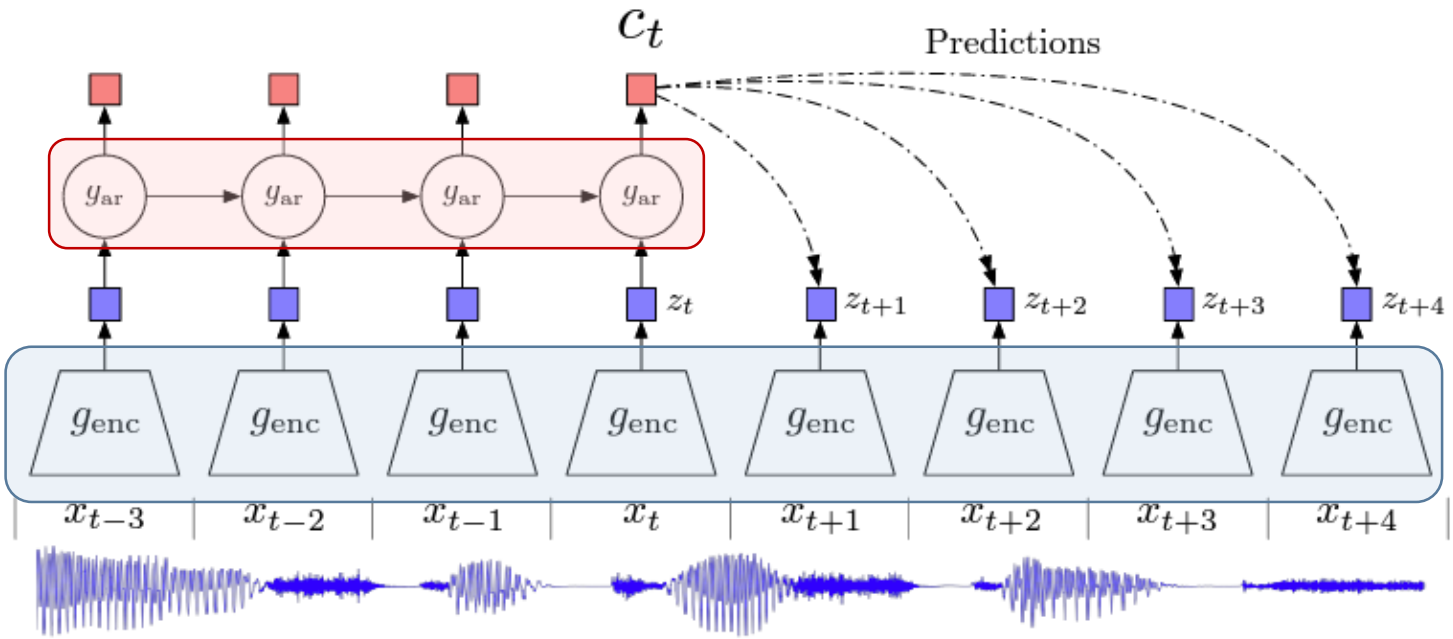
$$\underbrace{\text{score}(f(x), f(x^+))}_{\text{Positive samples}} >> \underbrace{\text{score}(f(x), f(x^-))}_{\text{Negative samples}}$$

Info Noise Contrastive Estimation Loss

# Contrastive Predictive Coding

## ❖ Architecture

- 크게 $g_{enc}$ 와 $g_{ar}$ 로 구성
- 입력 데이터(x)를 latent vector(z) 로 변형하는 $g_{enc}(nonlinear\ encoder)$
- z 를 context vector(c) 로 변형하는 $g_{ar}(autoregressive\ model)$

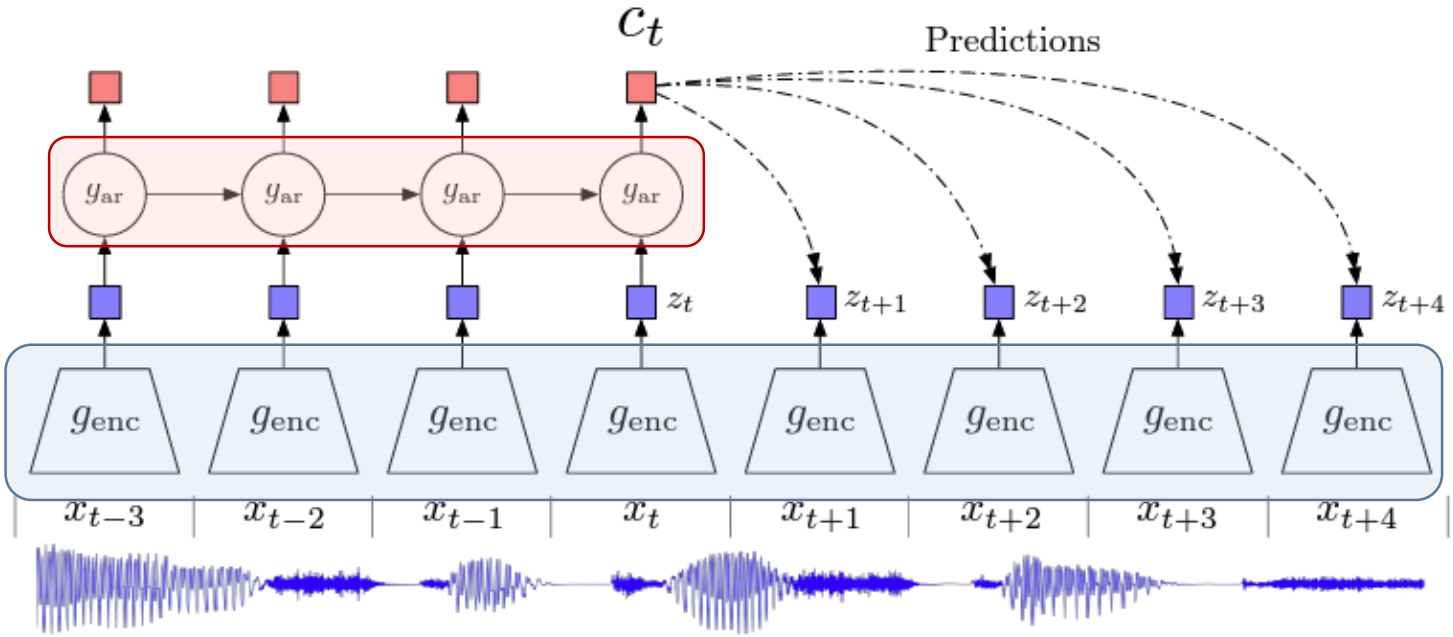



Overview of Contrastive Predictive Coding

# Contrastive Predictive Coding

## ❖ Architecture

1. $g_{enc}$ 을 통해 raw input을 잘라서 각각 latent vector(z)로 encoding
   - $g_{enc}$은 convolution layers 로 구성된 ResNet 구조 활용
2. $g_{ar}$은 encoding된 latent vector($z$)를 high-level Information인 context vector($c$)로 변환
   - $t$ 시점 이전 *latent vector*($z_t, z_{t-1}, z_{t-2}, z_{t-3}, \ldots, z_1$ ) 들을 통해 $C_t$ 추출
   - $g_{ar}$ 은 RNN 모델을 활용하여 구성(GRU, LSTM 등)




Overview of Contrastive Predictive Coding

# Contrastive Predictive Coding

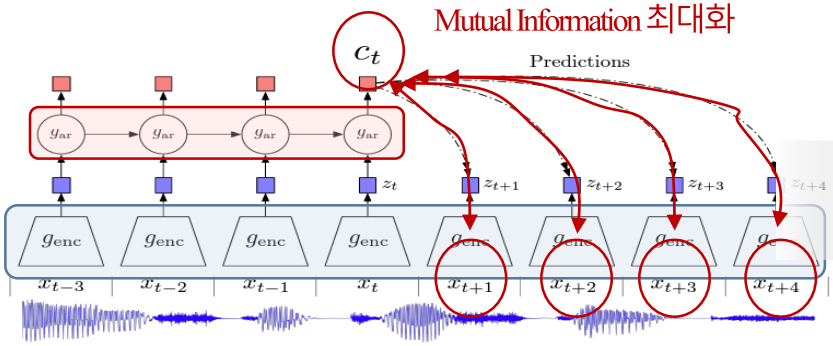
## ❖ Contrastive Predictive Coding

- 저자들은 generative model인 $p(x_{t+k}|c_t)$에서 k시점 이후, 미래의 관찰 값인 $x_{t+k}$를 직접적으로 생성하는 것이 아닌 $x_{t+k}$와 $c_t$사이에 존재하는 mutual information을 담고있는 density ratio $f$를 모델링
  - Mutual Information는 density ratio에 비례하며, 이를 최대화 하는 손실 함수를 고안
- $f$ 는 다양한 식으로 사용이 될 수 있는데, 저자들은 단순한 log-bilinear model을 사용

Definition of Mutual Information

비례

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




Density ratio

$$f_k(x_{t+k}, c_t) \propto \frac{p(x_{t+k}|c_t)}{p(x_{t+k})}$$

Simple log-bilinear model

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

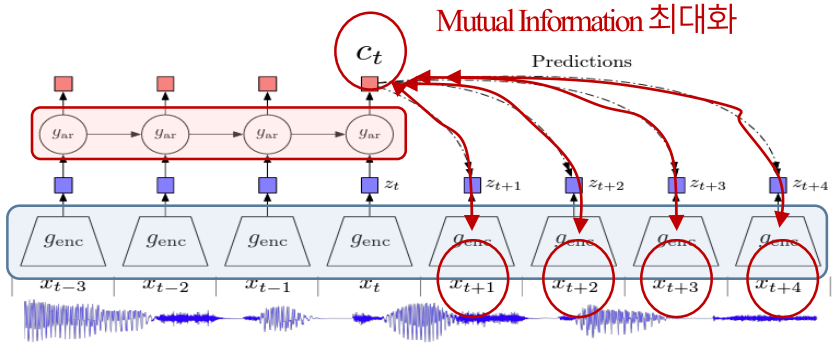
학습할 parameter

# Contrastive Predictive Coding

## ❖ Contrastive Predictive Coding

- 저자들은 generative model인 $p(x_{t+k}|c_t)$에서 k시점 이후, 미래의 관찰 값인 $x_{t+k}$를 직접적으로 생성하는 것이 아닌 $x_{t+k}$와 $c_t$사이에 존재하는 mutual information을 담고있는 density ratio $f$를 모델링
  - Mutual Information는 density ratio에 비례하며, 이를 최대화 하는 손실 함수를 고안
- $f$ 는 다양한 식으로 사용이 될 수 있는데, 저자들은 단순한 log-bilinear model을 사용
  - $f$ 값이 클수록 $x_{t+k}$와 $c_t$ 사이의 mutual information 이 큼

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




$W_k c_t$: $\hat{Z}_{t+k}$

$\hat{Z}_{t+k}$: *prediction of latent vector at* $x_{t+k}$

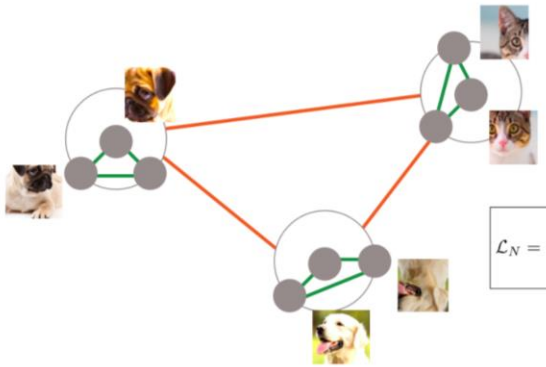
$Z_{t+k}$: *latent vector at* $x_{t+k}$

$Z_{t+k} \cdot \hat{Z}_{t+k}$(*dot product*): *similarity*

# Contrastive Predictive Coding

## ❖ Contrastive Predictive Coding

- 앞서 mutual information을 담고있는 density ratio $f$ 모델링을 통해 손실 함수인 InfoNCE Loss를 고안
- 최종적으로 아래 loss를 줄이도록 학습되며, log 안의 부분은 categorical cross-entropy 형태를 나타냄
  - *loss function* 의 분모에 해당하는 $x_{t+k}$와 $c_t$사이에 존재하는 mutual information 를 최대화 시키도록 학습



$$\mathcal{L}_N = -\mathbb{E}_X \left[ \log \frac{\exp(f(x)^T f(x^+))}{\exp(f(x)^T f(x^+)) + \sum_{j=1}^{N-1} \exp(f(x)^T f(x_j))} \right]$$

$$\text{score}(f(x), f(x^+)) >> \text{score}(f(x), f(x^-))$$

Positive samples　　Negative samples

*Postive sample* ↑

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

*Postive sample + Negative samples*

InfoNCELoss

# Experiments

## ❖ Speech

- Dataset
  - LibriSpeech Dataset(label이 없는 영어말하기 모음 데이터셋)에서 100시간 정도의 데이터를 샘플링
  - Kaldi toolkit을 활용하여 label 생성
- Phone classification: 음성에서 41개의 음소를 분류
  - 음소(音素): 더 이상 작게 나눌 수 없는 음운론상의 최소 단위
- Speaker classification: 251명의 화자를 구분
- Results:

Figure 2: t-SNE visualization of audio (speech) representations for a subset of 10 speakers (out of 251). Every color represents a different speaker.

| Method | ACC |
|---|---|
| **Phone classification** | |
| Random initialization | 27.6 |
| MFCC features | 39.7 |
| CPC | 64.6 |
| Supervised | 74.6 |
| **Speaker classification** | |
| Random initialization | 1.87 |
| MFCC features | 17.6 |
| CPC | 97.4 |
| Supervised | 98.5 |

Table 1: LibriSpeech phone and speaker classification results. For phone classification there are 41 possible classes and for speaker classification 251. All models used the same architecture and the same audio input sizes.

# Experiments

## ❖ Vision

- ILSVRC ImageNet Competition 데이터를 통해 이미지 분류 수행
- Architecture:
  - $g_{enc}$: ResNetv2 101
  - $g_{ar}$: Pixel CNN-style autoregressive model
- Results:

| Method | Top-1 ACC |
|---|---|
| **Using AlexNet conv5** | |
| Video [28] | 29.8 |
| Relative Position [11] | 30.4 |
| BiGan [35] | 34.8 |
| Colorization [10] | 35.2 |
| Jigsaw [29] * | 38.1 |
| **Using ResNet-V2** | |
| Motion Segmentation [36] | 27.6 |
| Exemplar [36] | 31.5 |
| Relative Position [36] | 36.2 |
| Colorization [36] | 39.6 |
| **CPC** | **48.7** |

Table 3: ImageNet top-1 unsupervised classification results. *Jigsaw is not directly comparable to the other AlexNet results because of architectural differences.

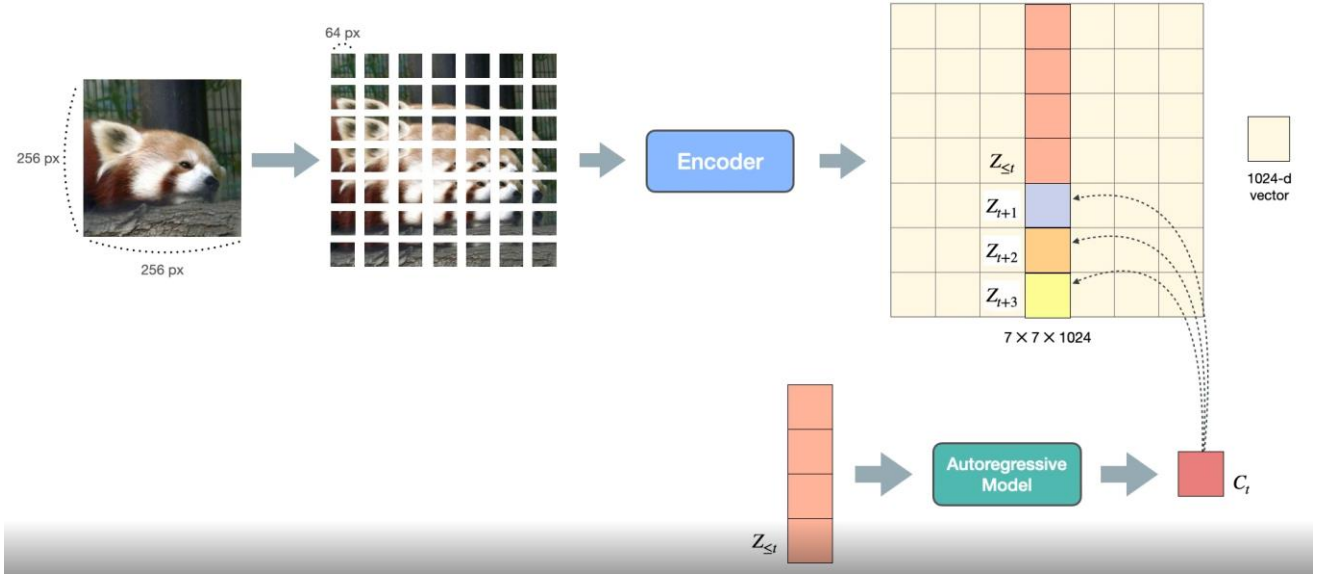
| Method | Top-5 ACC |
|---|---|
| Motion Segmentation (MS) | 48.3 |
| Exemplar (Ex) | 53.1 |
| Relative Position (RP) | 59.2 |
| Colorization (Col) | 62.5 |
| Combination of MS + Ex + RP + Col | 69.3 |
| **CPC** | **73.6** |

Table 4: ImageNet top-5 unsupervised classification results. Previous results with MS, Ex, RP and Col were taken from [36] and are the best reported results on this task.

# Experiments

## ❖ Vision

- Process(https://2-chae.github.io/category/2.papers/30 내 영상 자료 참고)
  1. Input image를 cropping, resize 하여 256 X 256 이미지 생성.
  2. 256 X 256 이미지를 64 X 64 사이즈 패치로 잘라 7X7 개의 이미지를 생성, 이때, overlap 허용
  3. ResNet v2(encoder) 모델을 활용하여 각 이미지로부터 latent vector 추출
  4. 각 이미지로부터 latent vector 추출한 output을 pooling시켜 1024-dim vector로 변환(7X7 X 1024 tensor)
  5. Pixel CNN-style autoregressive model을 사용하여 $c_t$를 출력, $c_t$를 통해 같은 column에 존재하는 다음 step time의 $z_{t+k}$들을 예측 → $\hat{z}_{t+k}$ 와 $z_{t+k}$이 비슷하게 되도록 학습하라는 뜻



# Experiments

## ❖ NLP

- Dataset
  - Pre-training: Book Corpus dataset
  - Fine-Tuning: Movie review sentiment, Customer product reviews, Opinion polarity (MPQA), TREC 등
- Results:

| Method | MR | CR | Subj | MPQA | TREC |
|---|---|---|---|---|---|
| Paragraph-vector [40] | 74.8 | 78.1 | 90.5 | 74.2 | 91.8 |
| Skip-thought vector [26] | 75.5 | 79.3 | 92.1 | 86.9 | 91.4 |
| Skip-thought + LN [41] | 79.5 | 82.6 | 93.4 | 89.0 | - |
| CPC | 76.9 | 80.1 | 91.2 | 87.7 | 96.8 |

Table 5: Classification accuracy on five common NLP benchmarks. We follow the same transfer learning setup from Skip-thought vectors [26] and use the BookCorpus dataset as source. [40] is an unsupervised approach to learning sentence-level representations. [26] is an alternative unsupervised learning approach. [41] is the same skip-thought model with layer normalization trained for 1M iterations.

# Reference

## ❖ Reference


- Van den Oord, A., Li, Y., & Vinyals, O. (2018). Representation learning with contrastive predictive coding. arXiv e-prints, arXiv-1807.
- https://2-chae.github.io/category/2.papers/30
- https://www.youtube.com/watch?v=X4fwPhGaR8Q


Thank You