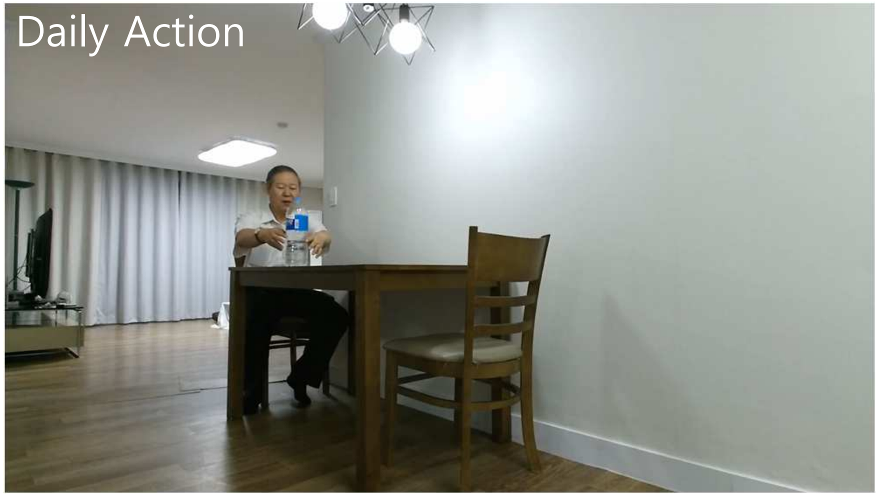
UST seminar

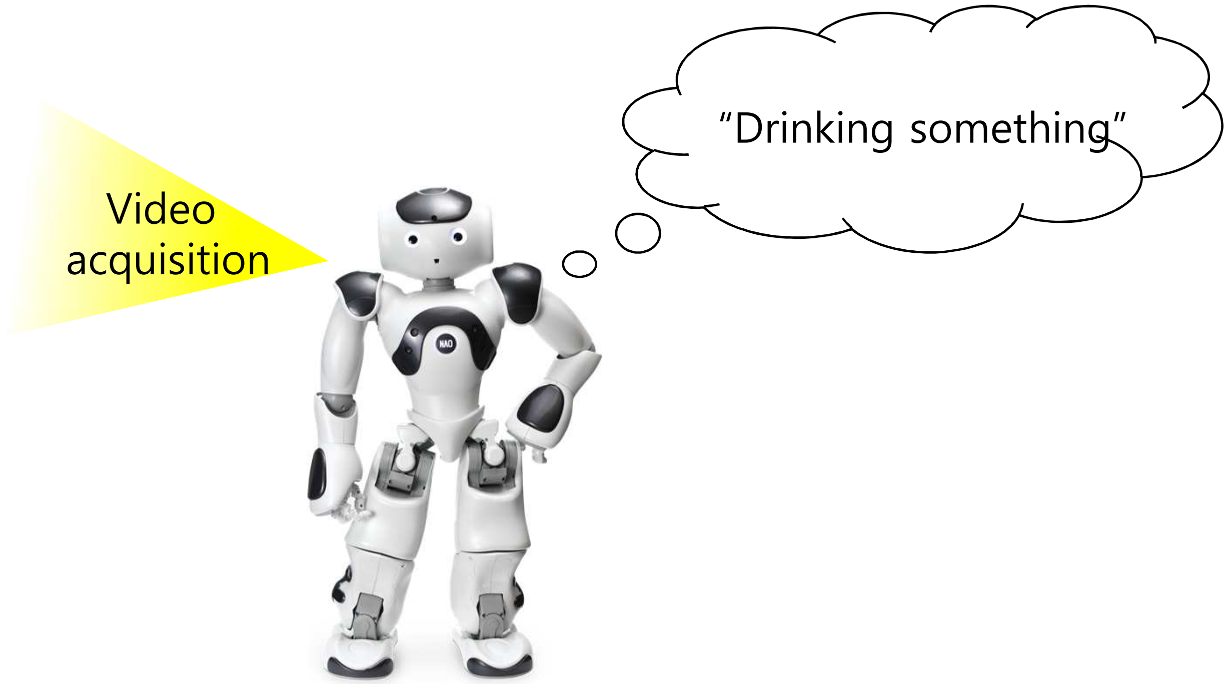
Towards domain-agnostic Video action recognition

Hyungmin Kim
UST-ETRI
Ph.D. student
khm159@etri.re.kr/ust.ac.kr
21th Sep, 2023

Human action recognition in Human-Robot Interaction



Video from [1]

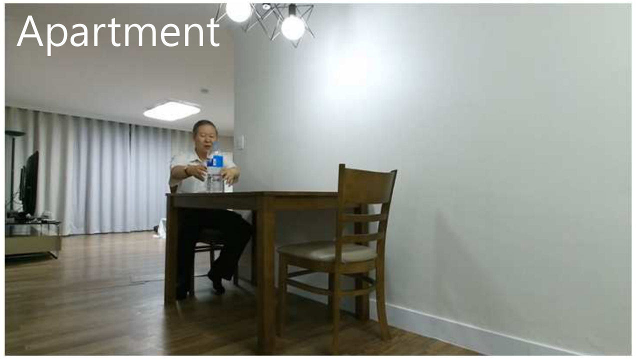


- **Problem definition :**

  Given input video v_i
  Model predicts corresponding action label y_i

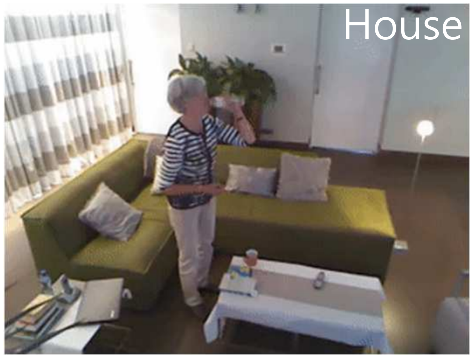
[1] Jang, Jinhyeok, et al. "ETRI-activity3D: A large-scale RGB-D dataset for robots to recognize daily activities of the elderly." *IROS*, 2020.

Service robot encounters numerous domains



Video from [1]

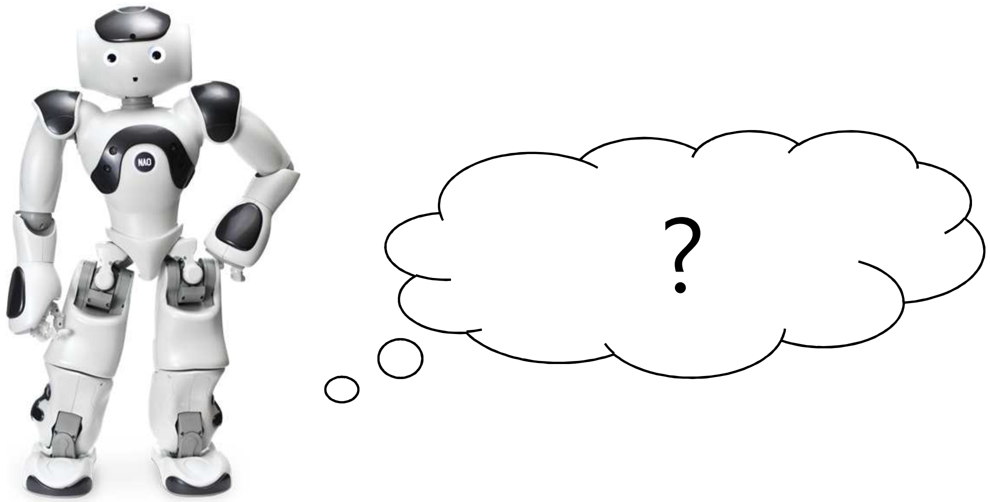
Video from [2]



Video from [3]

- Care-robot encounters numerous domains

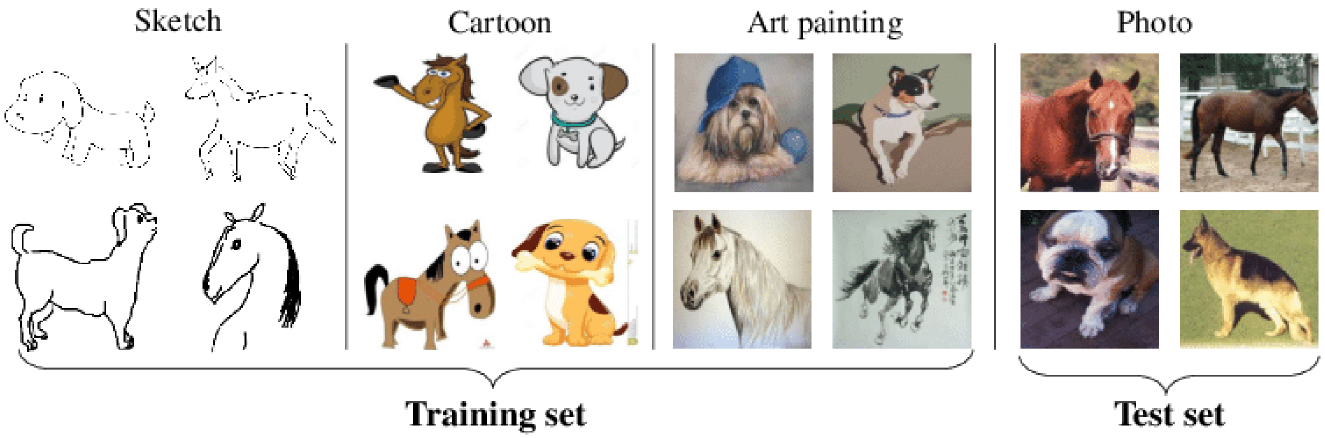
  Apartment, House, Building, Office ....



[1] Jang, Jinhyeok, et al. "ETRI-activity3D: A large-scale RGB-D dataset for robots to recognize daily activities of the elderly." *IROS*, 2020.
[2] Shahroudy, Amir, et al. "Ntu rgb+ d: A large scale dataset for 3d human activity analysis." *CVPR*. 2016.
[3] Das, Srijan, et al. "Toyota smarthome: Real-world activities of daily living." *ICCV*. 2019.

What is Domain Generalization?

Domain Generalization?



Train a generalized model with datasets from different domains that share the same semantic to improve performance on Un-seen domain.

Wang, Jindong, et al. "Generalizing to unseen domains: A survey on domain generalization." *IEEE Transactions on Knowledge and Data Engineering* (2022).

Motivation

- **Why DG is required?**
  Most of the existing deep learning/machine learning tasks operated under **i.i.d. assumption**.
  However, this is an impossible in real-world.

ICML21

Consider a classification task where the learning algorithm has access to i.i.d. data from m domains, $\{(d_i, \mathbf{x}_i, y_i)\}_{i=1}^n \sim (D_m, \mathcal{X}, \mathcal{Y})^n$ where $d_i \in D_m$ and $D_m \subset \mathcal{D}$ is a set of m domains. Each training in-

AAAI-22

In most statistical machine learning algorithms, a fundamental assumption is that the training data and test data are *independently and identically distributed* (i.i.d.). However, the data we have in many real-world applications are not i.i.d. Distributional shifts are ubiquitous. Under such

arXiv20

Convolutional Neural Networks (CNNs) show impressive performance in the standard classification setting where training and testing data are drawn i.i.d. from a given domain. However, CNNs do not readily generalize to new domains with different statistics, a setting that is simple for humans. In this work, we address

ICLR21

success of CNNs heavily relies on the i.i.d. assumption, i.e. training and test data should be drawn from the same distribution; when such an assumption is violated even just slightly, as in most real-world application scenarios, severe performance degradation is expected (Hendrycks & Dietterich,

ICML13

Since in general $\mathbb{P}^i_{XY} \neq \mathbb{P}^j_{XY}$, the samples in $\mathcal{S}$ are not i.i.d. Let $\widehat{\mathbb{P}}^i$ denote empirical distribution associated

NeurIPS18

discrete set $\{1, 2, \ldots N_c\}$, where N_c denotes the number of classes. Let $\{\mathcal{D}_i\}_{i=1}^{p+q}$ represent the $p+q$ distributions, each of which exists on the joint space $\mathcal{X} \times \mathcal{Y}$. Let $D_i = \{(\mathbf{x}_j^{(i)}, y_j^{(i)})\}_{j=1}^{N_i}$ represent the dataset sampled from the i^{th} distribution, i.e. each $(x_j^i, y_j^i) \overset{i.i.d.}{\sim} \mathcal{D}_i$. In the rest of the paper,

NeurIPS21

Problem 3.1 (Domain generalization). Let $\mathcal{E}_{\text{train}} \subsetneq \mathcal{E}_{\text{all}}$ be a finite subset of training domains, and assume that for each $e \in \mathcal{E}_{\text{train}}$, we have access to a dataset $\mathcal{D}^e := \{(x_j^e, y_j^e)\}_{j=1}^{n_e}$ sampled i.i.d. from $\mathbb{P}(X^e, Y^e)$. Given a function class $\mathcal{F}$ and a loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_{>0}$, our goal is to learn

ECCV20

We expand our notation set for the theoretical analysis. As we study the domain-agnostic cross-domain setting, we no longer work with $i.i.d$ data. There-

NeuralPS21

Independent and identically distributed (i.i.d.) condition is the underlying assumption of machine learning experiments. However, this assumption may not hold in real-world scenarios, *i.e.*, the training and the test data distribution may differ significantly by *distribution shifts*. For example, a

IEEE Trans. On Knowledge and Data Engineering

to new (test) data. Traditional ML models are trained based on the *i.i.d.* assumption that training and testing data are identically and independently distributed. However, this assumption does not always hold in reality. When the prob-

What is the i.i.d. assumption?

Independently and Identically Distributed Assumption

(1) **Independent** (Observations that acquire each data point are independent of each other)

Random samples from an unknown joint probability distribution (A.K.A. training dataset)

Each event are **independent** each other

Sample num	Feature 1	Feature 2	…	Feature N
1	1.24	True	…	…
2	…	…	…	…
3	…	…	…	…
…	…	…	…	…
M-1	….	…	…	…
M	0.11	False	…	…

What is the i.i.d. assumption?

Independently and Identically Distributed Assumption

(1) **Independent** (Observations that acquire each data point are independent of each other)

Random samples from an unknown joint probability distribution (A.K.A. training dataset)

Each event are **independent** each other

Sample num	Feature 1	Feature 2	…	Feature N
1	1.24	True	…	…
2	…	…	…	…
3	…	…	…	…
…	…	…	…	…
M-1	….	…	…	…
M	0.11	False	…	…

Why independent each other?

7

What is the i.i.d. assumption?

Independently and Identically Distributed Assumption

(1) **Independent** (Observations that acquire each data point are independent of each other)

Purpose :

Likelihood?

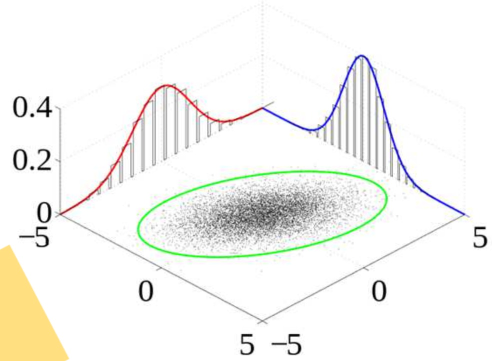
→ **To calculate Likelihood easily** 

8

Preliminary : Likelihood?

Likelihood :

The probability that the data is **derived from a particular probability distribution**



Unknown joint probability distribution

sampling

Sample num	Feature 1	Feature 2	…	Feature N
1	1.24	True	…	…
2	…	…	…	…
3	…	…	…	…
…	…	…	…	…
M-1	….	…	…	…
M	0.11	False	…	…

9

Preliminary : Likelihood?

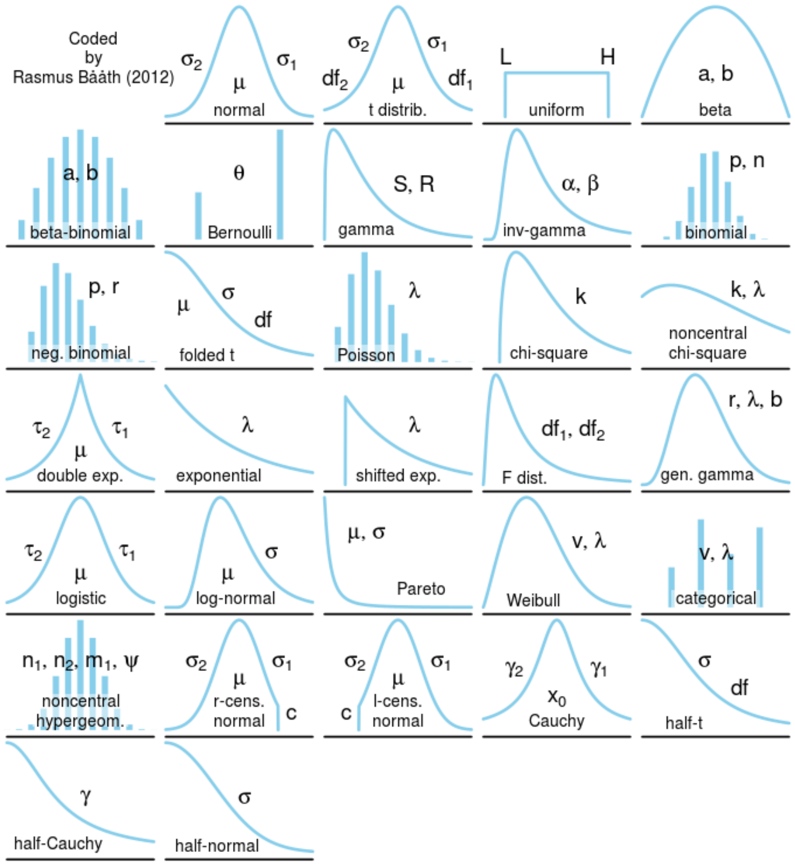
Likelihood :

Let there are the data : [1, 1, 1, 1]

To **which probability distribution** are the **given data points**

most likely to belong?

[1, 1, 1, 1]



Well defined distributions

Preliminary : Likelihood?

Likelihood :

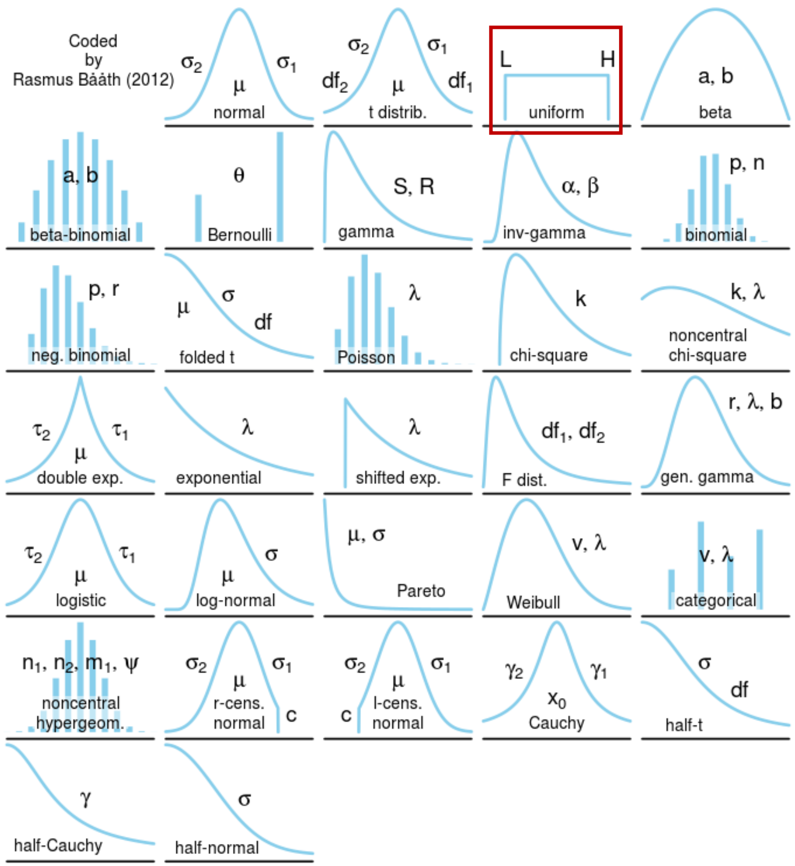
Let there are the data : [1, 1, 1, 1]

To **which probability distribution** are the **given data points**

most likely to belong?

[1, 1, 1, 1]

Uniform distribution



Well defined distributions

Preliminary : Likelihood?

Likelihood :

We basically **parameterize the unknown distribution** with parameters $\boldsymbol{\theta} = [\theta_1, \theta_2, \ldots, \theta_k]^T$

Preliminary : Likelihood?

Likelihood :

We basically **parameterize the unknown distribution** with parameters $\boldsymbol{\theta} = [\theta_1, \theta_2, \dots, \theta_k]^T$

For example gaussian …

$$g(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\left(-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right)} \approx \exp(\alpha x^2 + \beta x + \gamma)$$

Preliminary : Likelihood?

Likelihood :

We basically **parameterize the unknown distribution** with parameters $\boldsymbol{\theta} = [\theta_1, \theta_2, \dots, \theta_k]^T$

For example gaussian …

$$g(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\left(-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right)} \approx \exp(\alpha x^2 + \beta x + \gamma)$$

In this case…. The **Maximum Likelihood Estimation(MLE)** is to find optimal $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}$

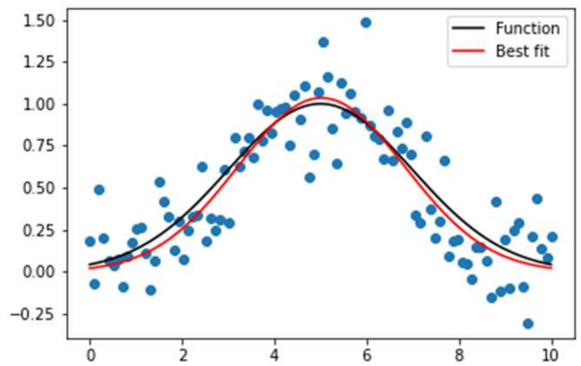
Preliminary : Likelihood?

Likelihood :

We basically **parameterize the unknown distribution** with parameters $\boldsymbol{\theta} = [\theta_1, \theta_2, \ldots, \theta_k]^T$

For example gaussian …

$$g(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\left(-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right)} \approx \exp(\alpha x^2 + \beta x + \gamma)$$



In this case…. The **Maximum Likelihood Estimation(MLE)** is to find optimal $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}$

To find **mostly likely belonging probability distribution** when given data points…

Preliminary : Likelihood?

Likelihood :

We basically **parameterize the unknown distribution** with parameters $\boldsymbol{\theta} = [\theta_1, \theta_2, \dots, \theta_k]^T$

Now we can define the likelihood with the given parameter $\boldsymbol{\theta}$ and dataset X like this…

$$Likelohood(\boldsymbol{\theta}) = p(X|\boldsymbol{\theta})$$

Preliminary : Likelihood?

Likelihood :

We basically **parameterize the unknown distribution** with parameters $\boldsymbol{\theta} = [\theta_1, \theta_2, \dots, \theta_k]^T$

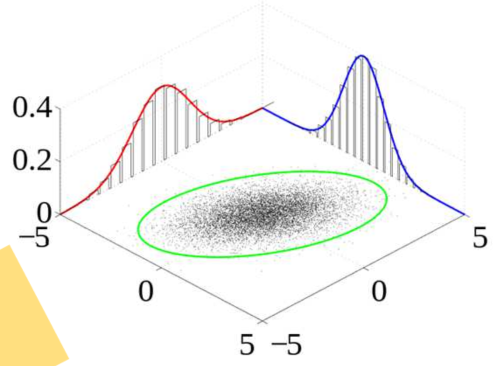
Now we can define the likelihood with the given parameter $\boldsymbol{\theta}$ and dataset X like this…

$$Likelohood(\boldsymbol{\theta}) = p(X|\boldsymbol{\theta})$$

If $\boldsymbol{p}(\boldsymbol{X}|\boldsymbol{\theta})$ is high, then we say that the estimated $\boldsymbol{\theta}$ **effectively explain Unknown (original) distribution of $\boldsymbol{X}$**

So.. Now we know the likelihood

We can model transfer function with some parameters $\boldsymbol{\theta}$
The probability that the data is **derived from a particular probability distribution**



Unknown joint probability distribution

sampling

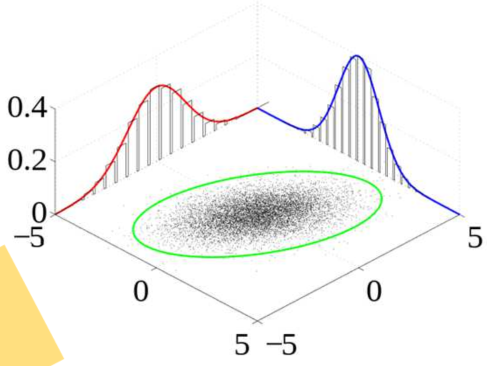
Sample num	Feature 1	Feature 2	…	Feature N
1	1.24	True	…	…
2	…	…	…	…
3	…	…	…	…
…	…	…	…	…
M-1	….	…	…	…
M	0.11	False	…	…

18

So.. Now we know the likelihood

We can model transfer function with some parameters $\boldsymbol{\theta}$
The probability that the data is **derived from a particular probability distribution**

We estimate $\boldsymbol{\theta}$ when likelihood is maximized.



Unknown joint probability distribution

sampling

MLE

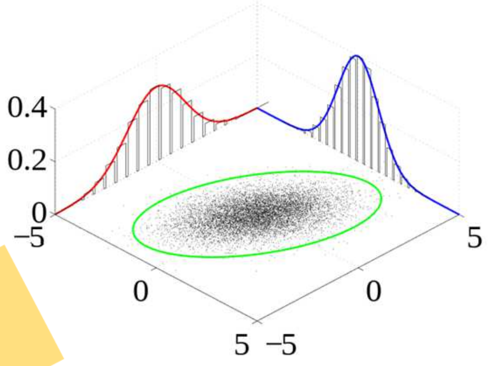
Sample num	Feature 1	Feature 2	…	Feature N
1	1.24	True	…	…
2	…	…	…	…
3	…	…	…	…
…	…	…	…	…
M-1	….	…	…	…
M	0.11	False	…	…

Remember, we need to find the model from dataset

So.. Now we know the likelihood

We can model transfer function with some parameters $\boldsymbol{\theta}$
The probability that the data is **derived from a particular probability distribution**

We estimate $\boldsymbol{\theta}$ when likelihood is maximized.



Unknown joint probability distribution

sampling

MLE

Sample num	Feature 1	Feature 2	…	Feature N
1	1.24	True	…	…
2	…	…	…	…
3	…	…	…	…
…	…	…	…	…
M-1	….	…	…	…
M	0.11	False	…	…

$$Likelohood(\boldsymbol{\theta}) = p(X|\boldsymbol{\theta}) \uparrow$$

20

What is the i.i.d. assumption?

Independently and Identically Distributed Assumption

(1) **Independent** (Observations that acquire each data point are independent of each other)

Purpose :

→ **To calculate Likelihood easily**

it can be expressed **in the form of a product**, so the complexity of the operation can be minimized by adding log.

I don’t get it…

21

What is the i.i.d. assumption?

Independently and Identically Distributed Assumption

(1) **Independent** (Observations that acquire each data point are independent of each other)

Purpose :

→ **To calculate Likelihood easily**

it can be expressed **in the form of a product**, so the complexity of the operation can be minimized by adding log.

Likelihood

INDEPENDENT!

$$l(\theta) = \underline{P(x_1, x_2, x_3, \ldots, x_n|\theta) = P(x_1|\theta)P(x_2|\theta)P(x_3|\theta)\ldots P(x_n|\theta)} = \prod_{n=1}^{N} P(x_n|\theta)$$

22

What is the i.i.d. assumption?

Independently and Identically Distributed Assumption

(1) **Independent** (Observations that acquire each data point are independent of each other)

Purpose :

→ **To calculate Likelihood easily**

it can be expressed **in the form of a product**, so the complexity of the operation can be minimized by adding log.

Likelihood

INDEPENDENT!

$$l(\theta) = \underline{P(x_1, x_2, x_3, \ldots, x_n|\theta) = P(x_1|\theta)P(x_2|\theta)P(x_3|\theta)\ldots P(x_n|\theta)} = \prod_{n=1}^{N} P(x_n|\theta)$$

$$= \boldsymbol{log} \prod_{n=1}^{N} P(x_n|\theta)$$

$$= \underline{\sum_{n=1}^{N} P(x_n|\theta)}$$ **SUMMATION!**

23

What is the i.i.d. assumption?

Independently and Identically Distributed Assumption

If you are interested in more **relationship between MLE and cross-entropy or MSE loss**...
Then I recommend following materials.. (because of the presentation time limit I'm not explain this)
I was able to get a good insight from the materials below.

(1) Miranda and Lester James, "Understanding softmax and the negative log-likelihood", 2017.
(2) Hwalseok Lee, "Everything about autoencoder", 2018 (Korean)
(3) Doersch, Carl. "Tutorial on variational autoencoders." *arXiv preprint,* 2016).
(4) Scaling Up Deep Learning - Yoshua Bengio, *ICML*, 2014.

What is the i.i.d. assumption?

Independently and Identically Distributed Assumption

If not independent?

$$l(\theta) = P(x_1, x_2, x_3, \ldots, x_n | \theta)$$

The joint probability are **extremely hard** to define and calculate....

Remember.. No dependent....

What is the i.i.d. assumption?

Independently and Identically Distributed Assumption

(2) **Identically distributed**

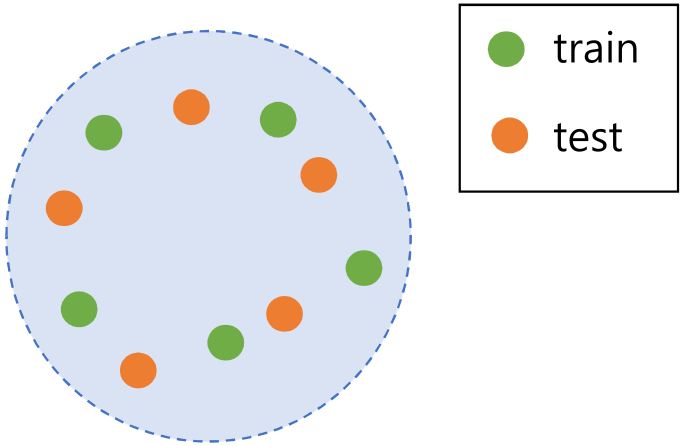
There are no overall trends the distribution **doesn't fluctuate** and
all items in the sample are **taken from the same probability distribution.**

What is the i.i.d. assumption?

Independently and Identically Distributed Assumption

(2) **Identically distributed**

There are no overall trends the distribution **doesn't fluctuate** and
all items in the sample are **taken from the same probability distribution.**



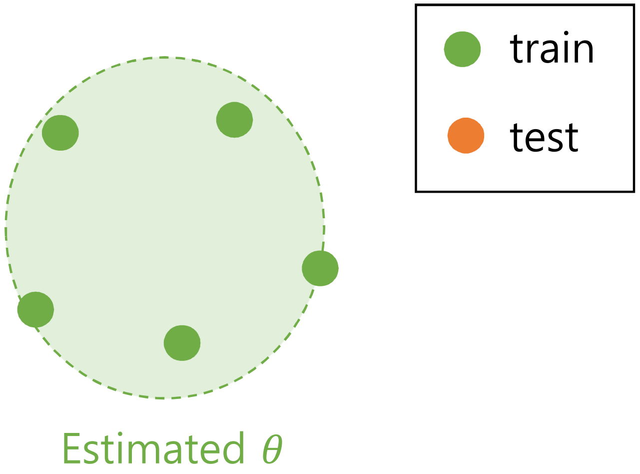
Ex. gaussian

What is the i.i.d. assumption?

Independently and Identically Distributed Assumption

(2) **Identically distributed**

There are no overall trends the distribution **doesn't fluctuate** and
all items in the sample are **taken from the same probability distribution.**



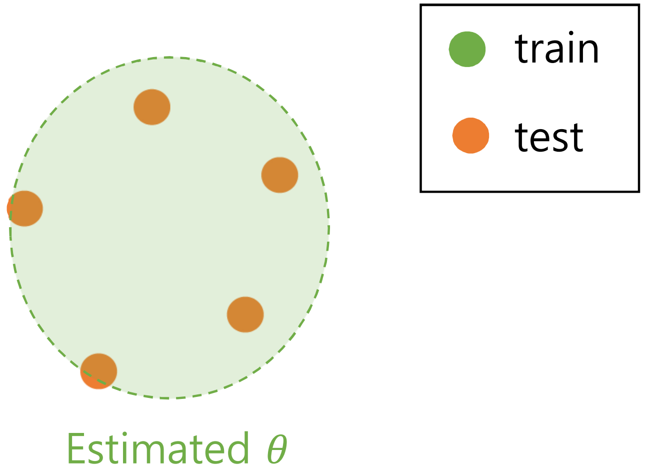
<step1>
Do MLE for optimized model
On training dataset

What is the i.i.d. assumption?

Independently and Identically Distributed Assumption

(2) **Identically distributed**

There are no overall trends the distribution **doesn't fluctuate** and
all items in the sample are **taken from the same probability distribution.**



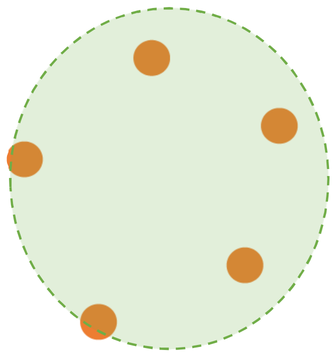
<step2>
The optimized model can effectively explain Testing dataset
if train and test dataset are identically distributed

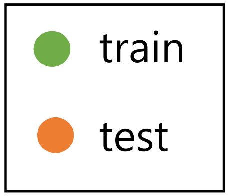
What is the i.i.d. assumption?

Independently and Identically Distributed Assumption

(2) **Identically distributed**

There are no overall trends the distribution **doesn't fluctuate** and
all items in the sample are **taken from the same probability distribution.**





Estimated θ

<step2>
The optimized model can effectively explain Testing dataset
if train and test dataset are identically distributed

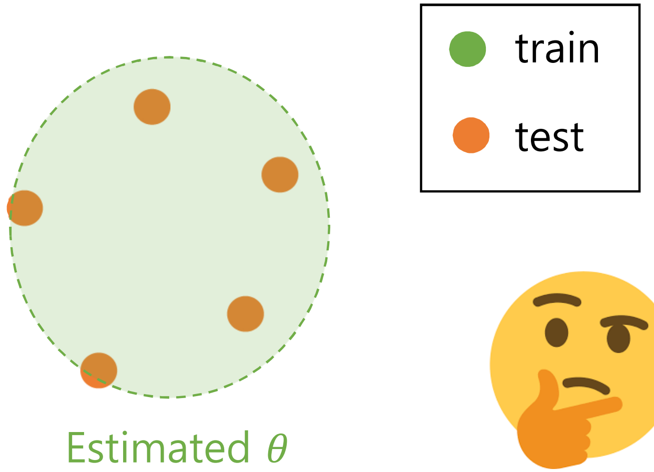
Is it possible?

What is the i.i.d. assumption?

Independently and Identically Distributed Assumption

(2) **Identically distributed**

There are no overall trends the distribution **doesn't fluctuate** and
all items in the sample are **taken from the same probability distribution.**



<step2>
The optimized model can effectively explain Testing dataset
if train and test dataset are identically distributed

Is it possible?

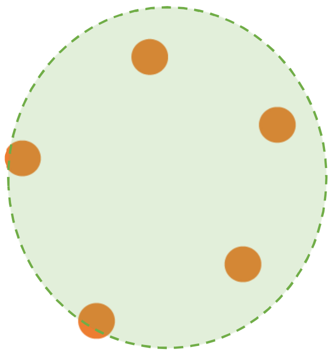
Yes!

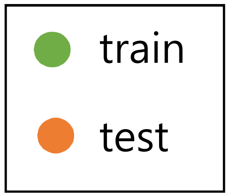
What is the i.i.d. assumption?

Independently and Identically Distributed Assumption

(2) **Identically distributed**

There are no overall trends the distribution **doesn't fluctuate** and
all items in the sample are **taken from the same probability distribution.**





Estimated θ

<step2>
The optimized model can effectively explain Testing dataset
if train and test dataset are identically distributed

Is it possible?

Yes!

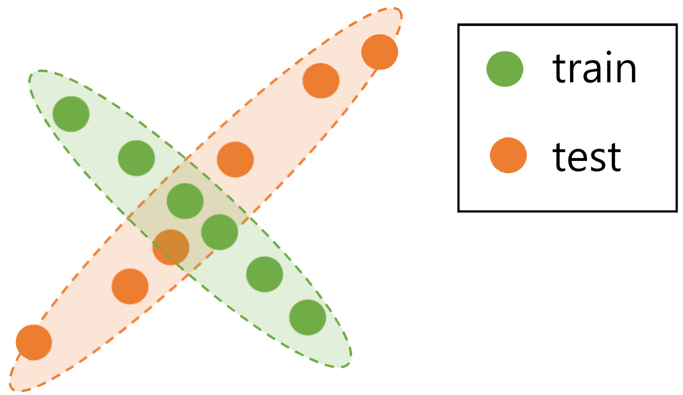
If the data obtained is representative of same population

What is the i.i.d. assumption?

Independently and Identically Distributed Assumption

(2) **Identically distributed**

If not? **The training is useless**



The training set and the test set represent **different distributions.**

Summary

Independently and Identically Distributed Assumption

(1) Independent

To calculate likelihood easily

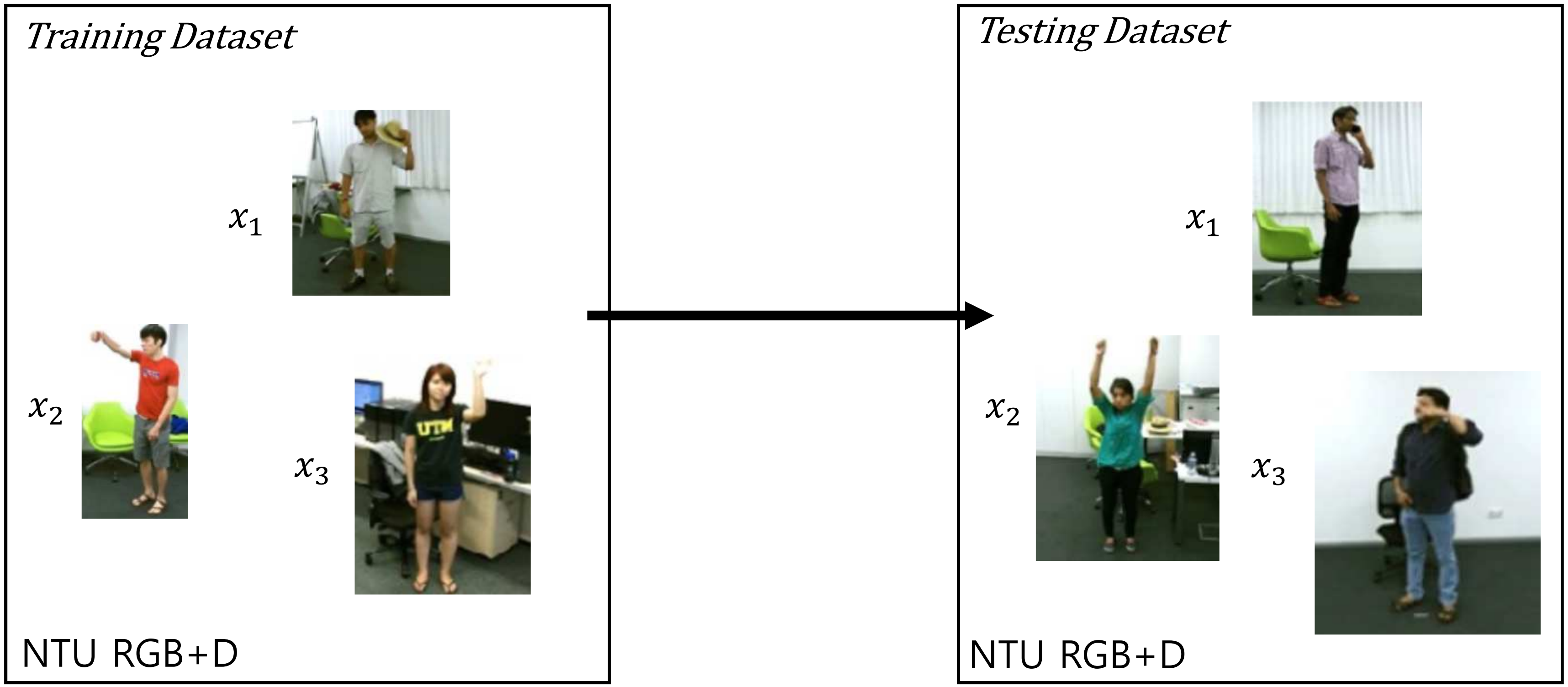
(2) Identically distributed

The population represented by the learning set and the test set is the same.

In i.i.d. assumption...

- Usually, the training set and test set are obtained in the same or similar way.
  Each sample is Independent Identically distributed.

Similar back ground, similar camera view point, etc...



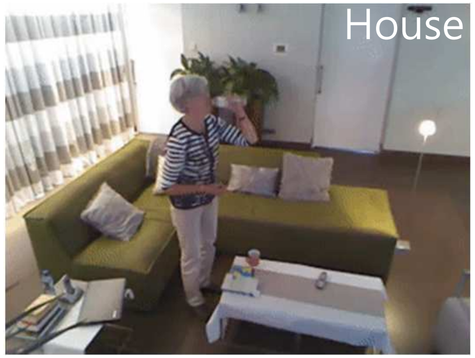
i.i.d. assumption is easily violated in real world

- Care-robot encounters numerous domains

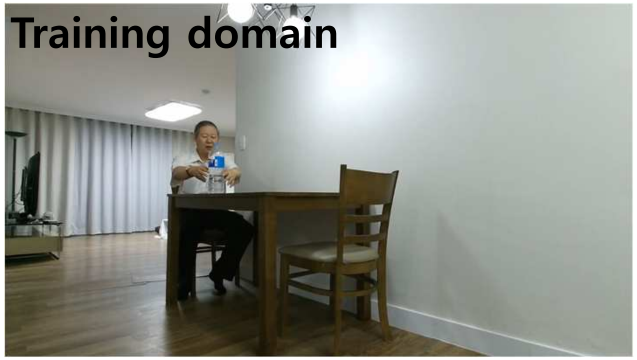
Apartment, House, Building, Office ….

Inferenced domains

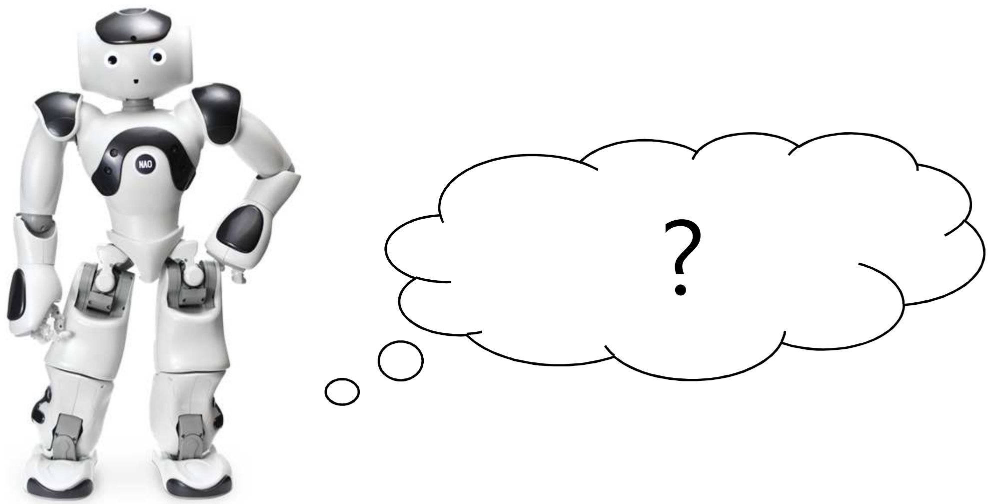
Video from [2]



Video from [3]



Video from [1]



[1] Jang, Jinhyeok, et al. "ETRI-activity3D: A large-scale RGB-D dataset for robots to recognize daily activities of the elderly." *IROS*, 2020.
[2] Shahroudy, Amir, et al. "Ntu rgb+ d: A large scale dataset for 3d human activity analysis." *CVPR*. 2016.
[3] Das, Srijan, et al. "Toyota smarthome: Real-world activities of daily living." *ICCV*. 2019.

The goal of DG

Using many training domains well
to train models that are **robust to domain differences**.

ETRI Activity-3D
domain-specific : old man, dinning room…

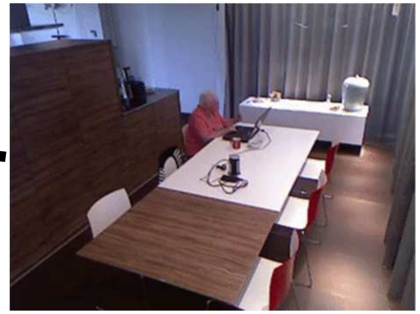
Towards OOD!!

NTU-RGB+D
Domain-specific : girl, office…

Semantic :
“something to eat"

generative
model
$f_\theta(\cdot)$

Toyota Smarthome
Domain-specific : old woman, kitchen…



Evaluation protocol

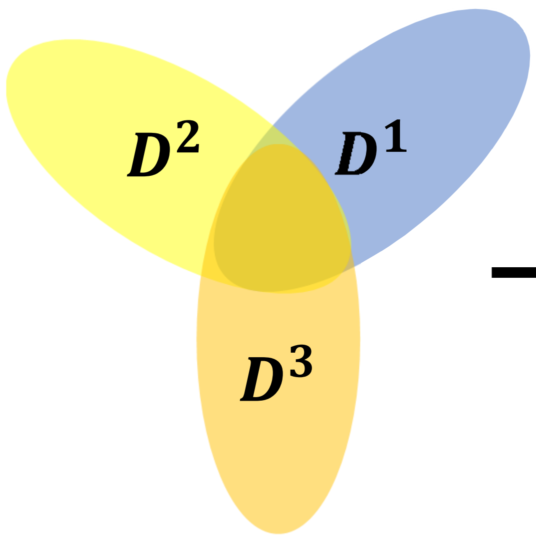
- **Leave-one-domain-out protocol**

  Set of domains $D = \{D^i | i = 1, \dots M\}$

  Each domain $D^i = \{(x^{i,j}, y^{i,j})\}_{j=1}^{n_i}$

  one domain from D is excluded or left out during training,
  while the remaining domains of D are used for training the model.

  - n_j : number of samples of D^i
  - M : number of domains



If $D = \{D^i | i = 1, \dots 4\}$

Evaluate D^4

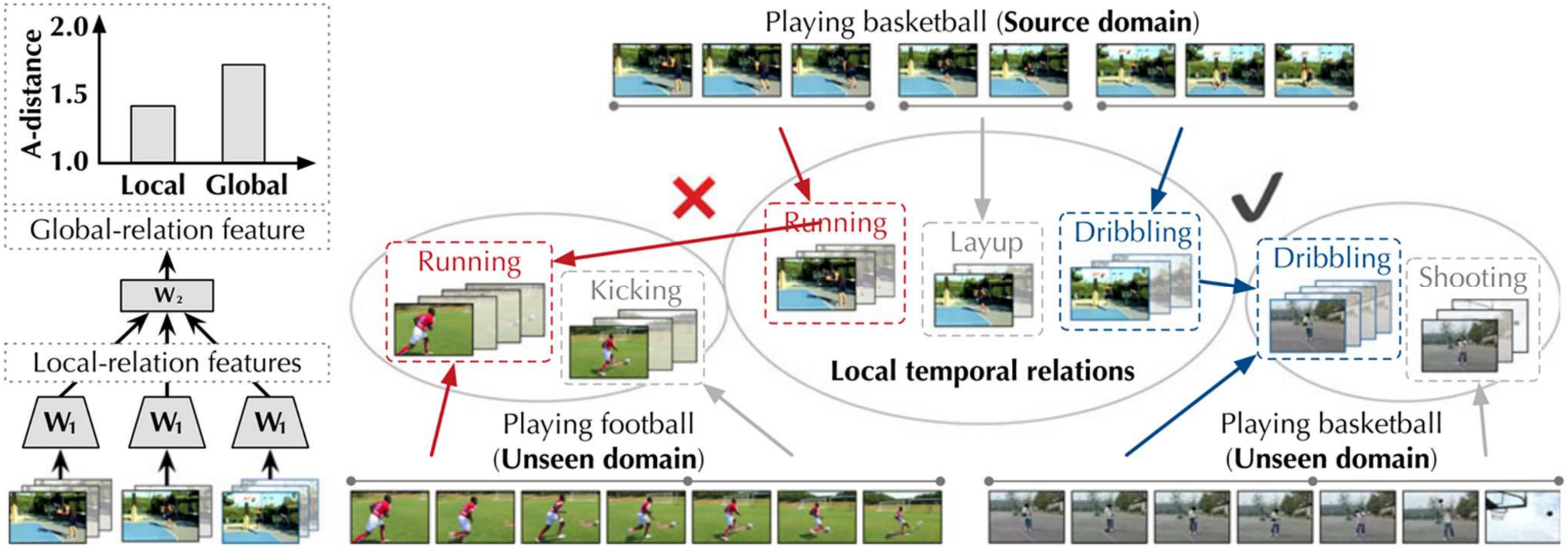
DG in video domain

Unfortunately, there are few papers and little research.

So, I will introduce a pioneer paper.
(best of my knowledge, **in real domain**, **only two(and one is my work…**)).

VideoDG review

Motivation : **The "temporal" discrepancy** is **co-existed** with "spatial" discrepancy in video domain.



(a) Feature discrepancy (b) A negative and a positive example of video domain generalization

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021.*

VideoDG review

(1) Spatial domain difference

Caused by the variations of the appearance of video frames
Partially solve by image-based domain generalization methods

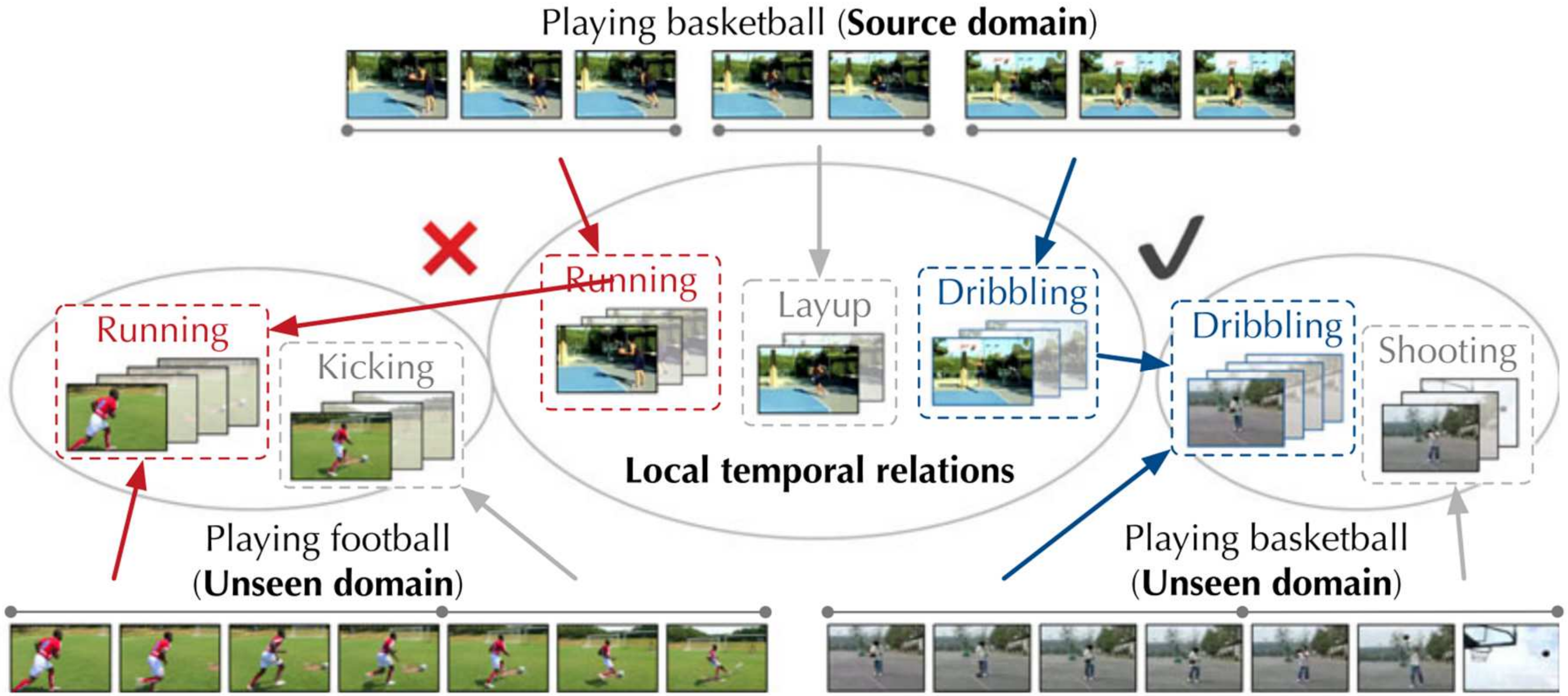
Example of the spatial domain difference in video (ENT dataset*)

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021.*
*Hyungmin, Kim et al. "A simple baseline for domain generalization of action recognition and a realistic out-of-domain scenario." *International Conference on Ubiquitous Robots (UR).* IEEE, 2023.

VideoDG review

(2) Temporal domain difference

The unexpected absence or misalignment of short-term video events



(b) A negative and a positive example of video domain generalization

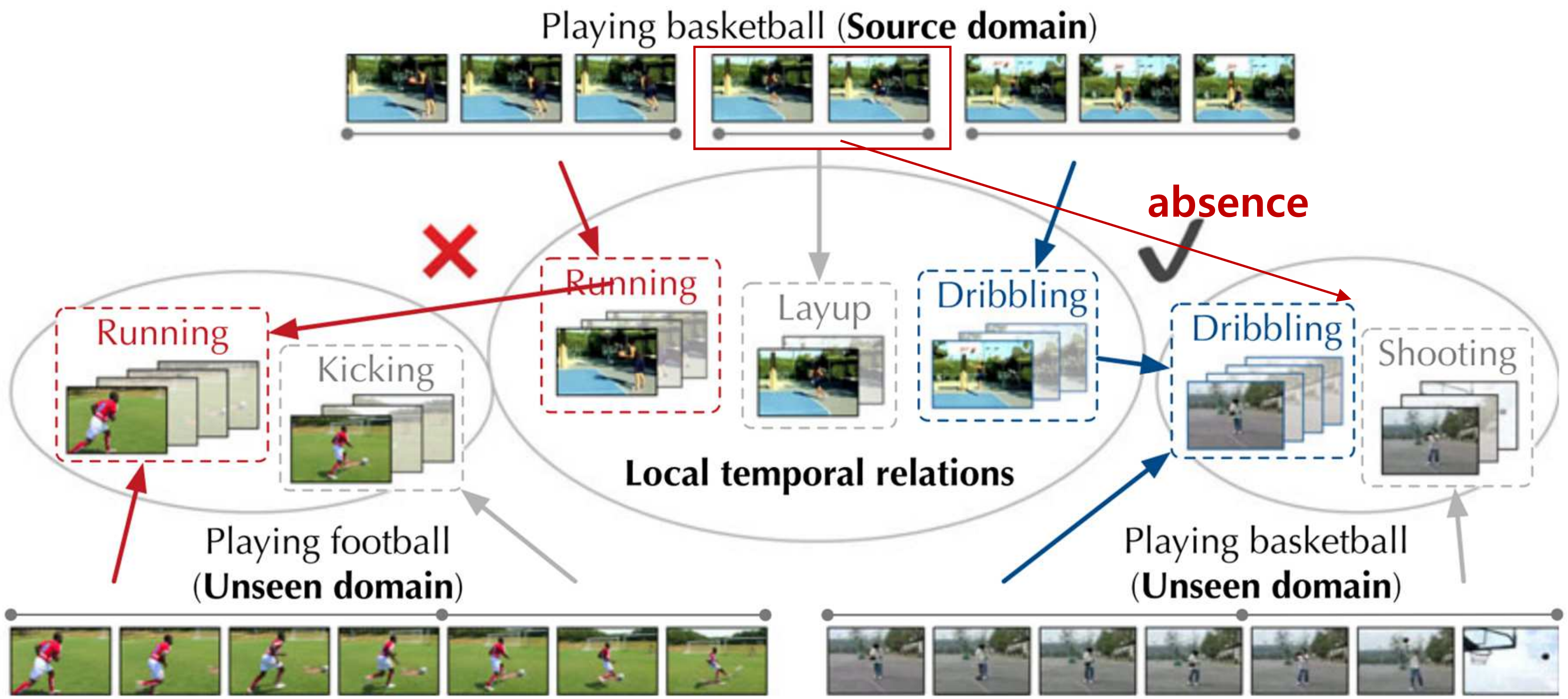
Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021.*

VideoDG review

(2) Temporal domain difference

Short-term events can be existed or confused When domain-shift is occurred.

The unexpected **absence** or misalignment of short-term video events



(b) A negative and a positive example of video domain generalization

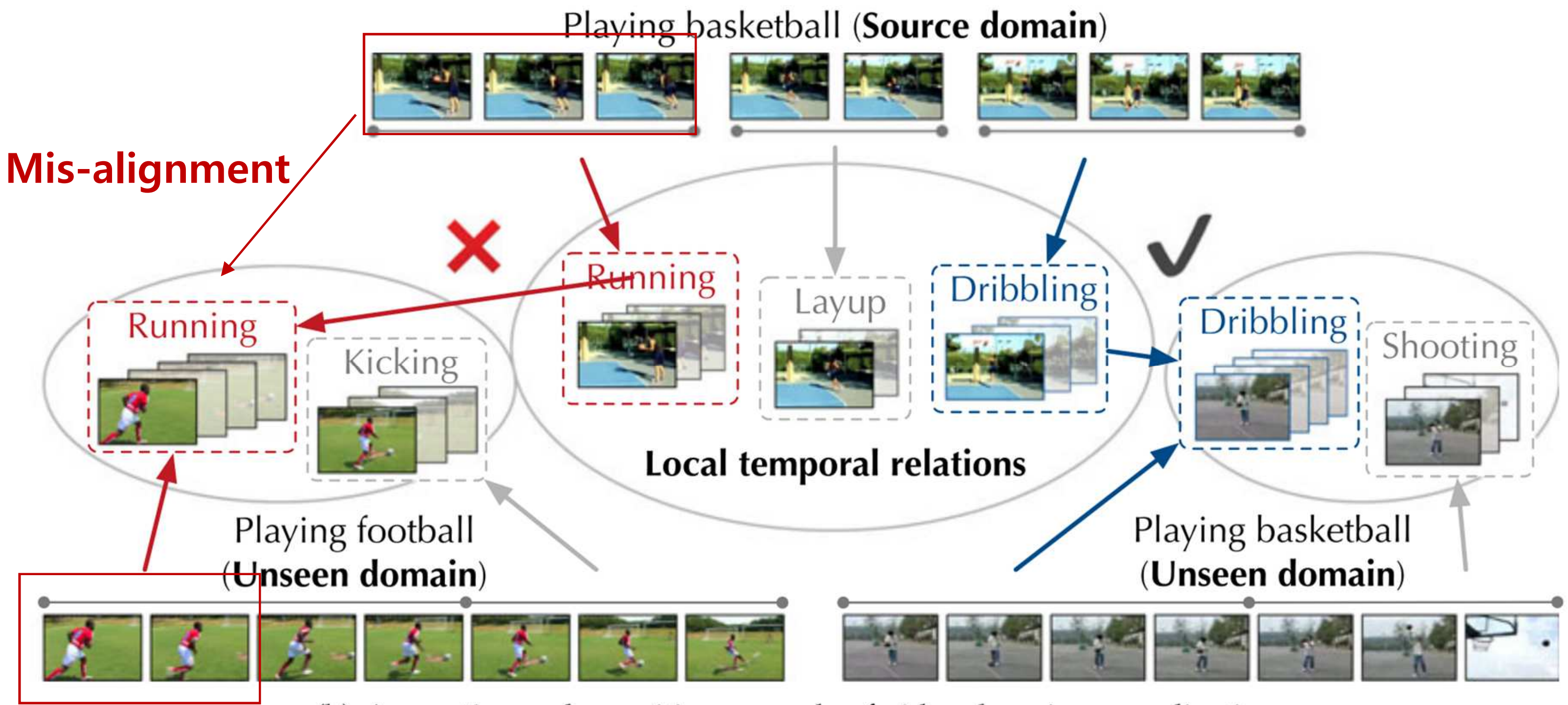
Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021.*

VideoDG review

(2) Temporal domain difference

Short-term events can be existed or confused When domain-shift is occurred.

The unexpected absence or **misalignment** of short-term video events



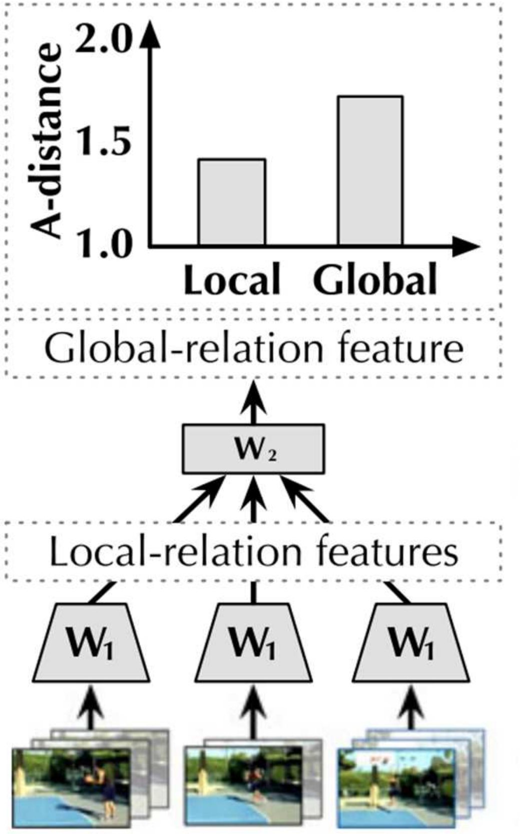
(b) A negative and a positive example of video domain generalization

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021.*

VideoDG review

In case of UCF→HMDB

Motivation : **The "temporal" discrepancy** is **co-existed** with "spatial" discrepancy in video domain.



(a) Feature discrepancy

A-distance*

If A-distance large : two distribution far apart

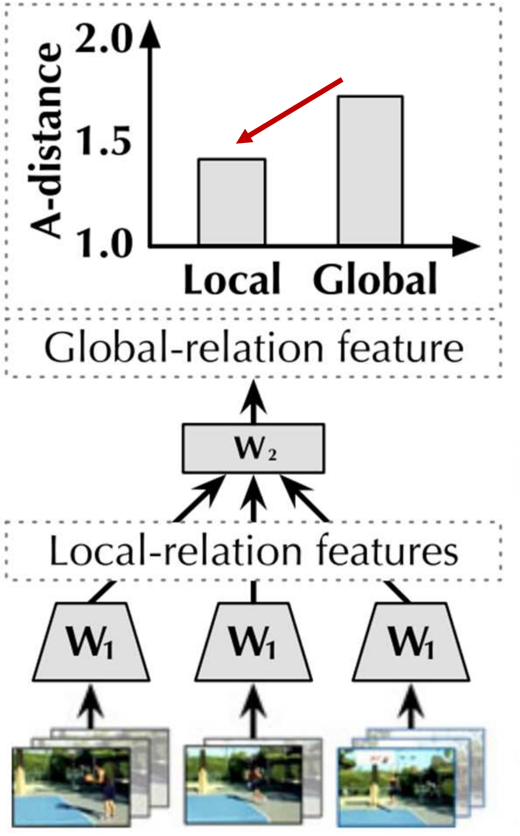
If A-distance small : two distribution close to each other

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021.*
*Ben-David, Shai, et al. "Analysis of representations for domain adaptation." *Advances in neural information processing systems* 19 (2006).

VideoDG review

In case of UCF→HMDB

Motivation : **The “temporal” discrepancy** is **co-existed** with “spatial” discrepancy in video domain.



(a) Feature discrepancy

A-distance*

If A-distance large : two distribution far apart

If A-distance small : two distribution close to each other

The local feature are more generalizable then global feature

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021.*
*Ben-David, Shai, et al. "Analysis of representations for domain adaptation." *Advances in neural information processing systems* 19 (2006).

VideoDG review

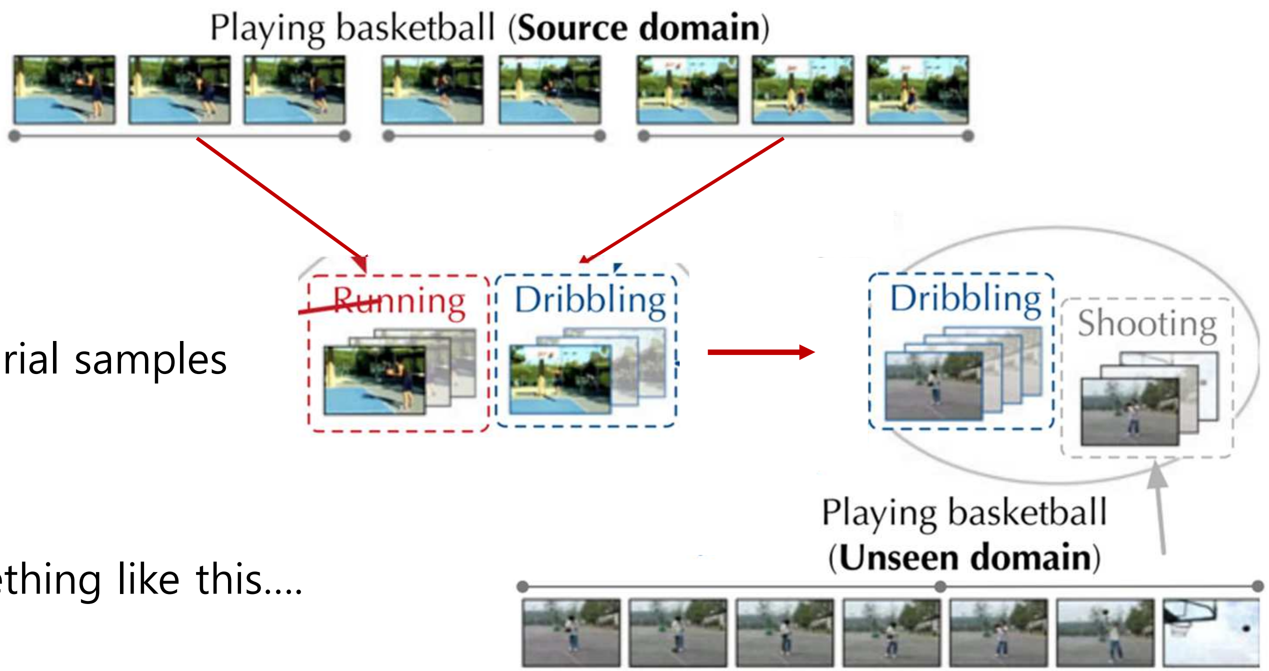
Main Idea

The local feature are more generalizable then global feature

Using local features **to generate adversarial samples** to augment training dataset

Adversarial samples

Something like this....



Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021.*

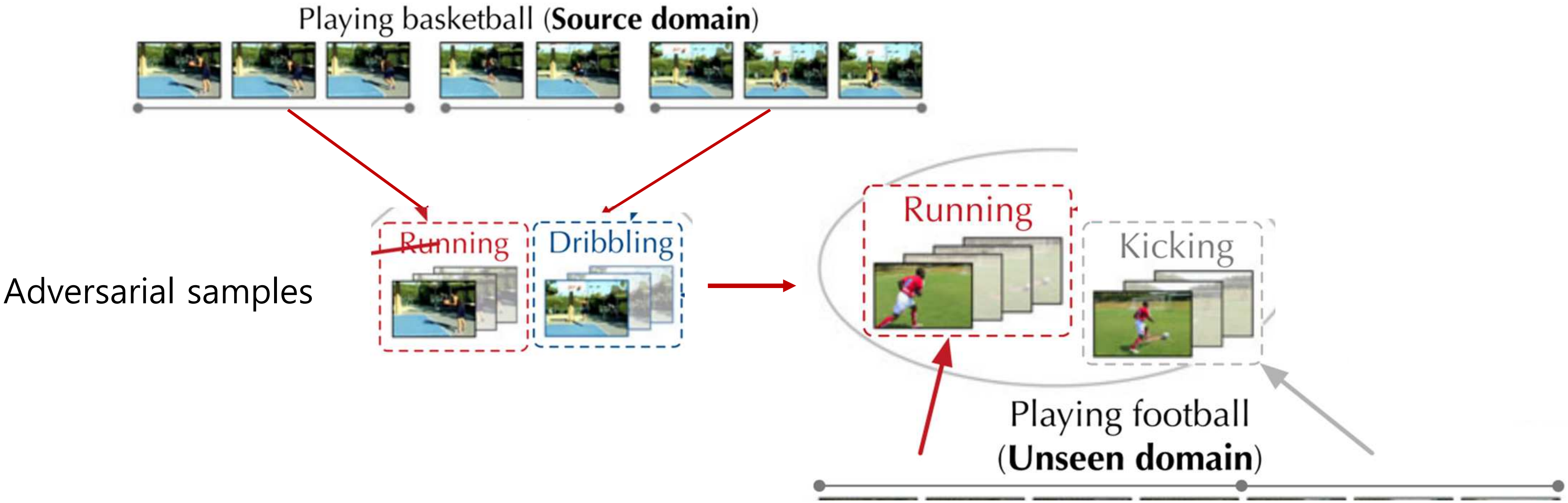
*Ben-David, Shai, et al. "Analysis of representations for domain adaptation." *Advances in neural information processing systems* 19 (2006).

VideoDG review

Main Idea

The local feature are more generalizable then global feature

Using local features **to generate adversarial samples** to augment training dataset



But also **degrade the discriminability** and can miss classify into similar categories

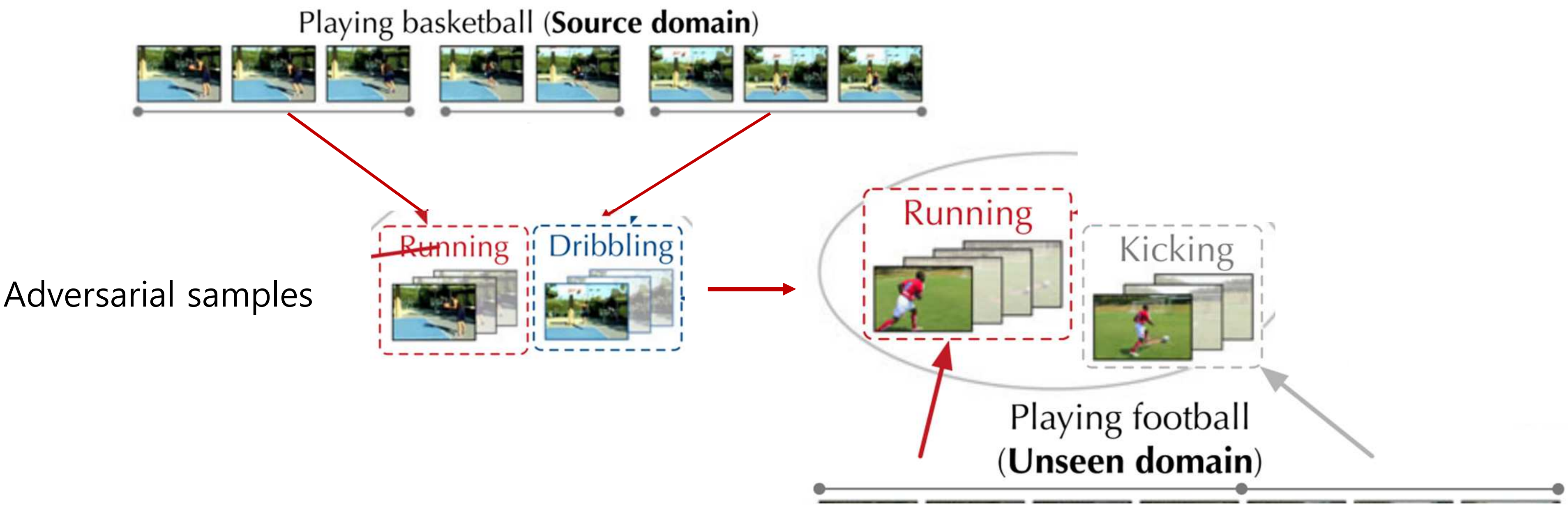
Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021.*
*Ben-David, Shai, et al. "Analysis of representations for domain adaptation." *Advances in neural information processing systems* 19 (2006).

VideoDG review

Main Idea

The local feature are more generalizable then global feature

Using local features **to generate adversarial samples** to augment training dataset



But also degrade the discriminability and can miss classify into similar categories

The global feature are also important

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021.*

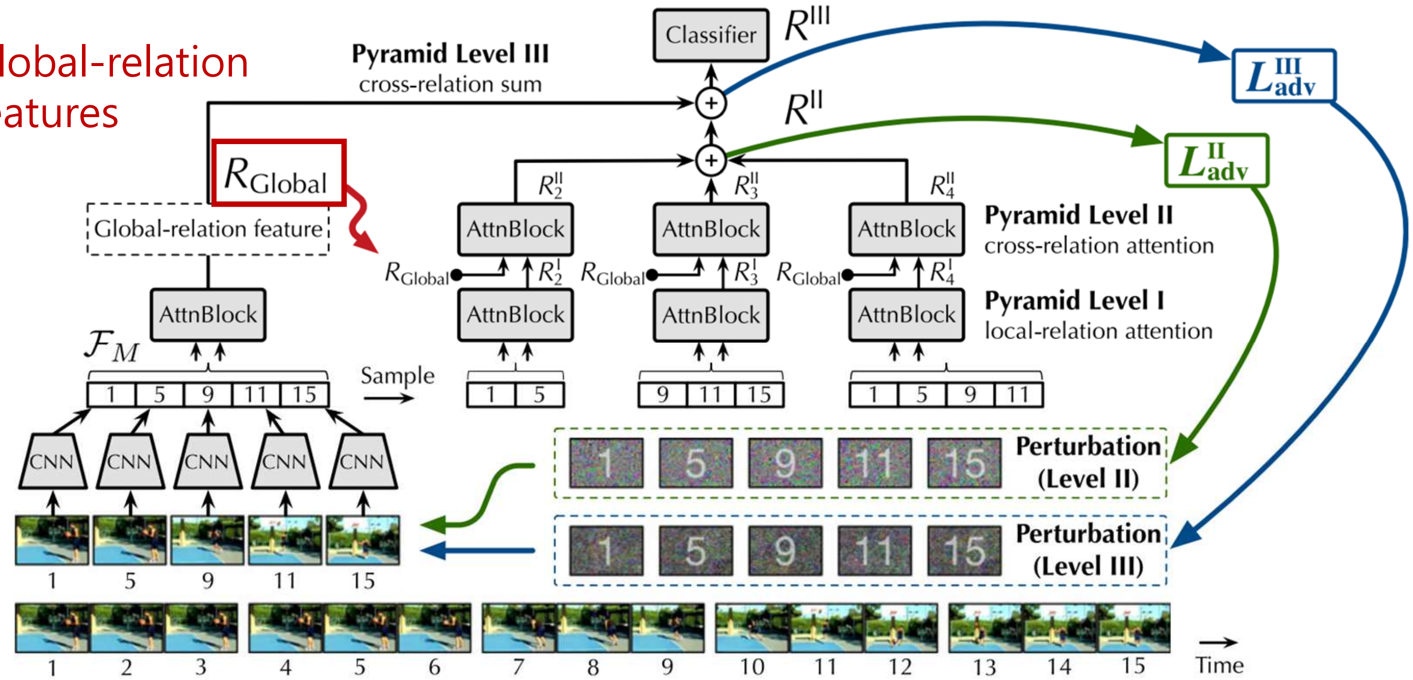
*Ben-David, Shai, et al. "Analysis of representations for domain adaptation." *Advances in neural information processing systems* 19 (2006).

VideoDG review : APN

APN(Adversarial Pyramid Network)

To use the **global-relation features to guide the generalization of local events**

Global-relation features

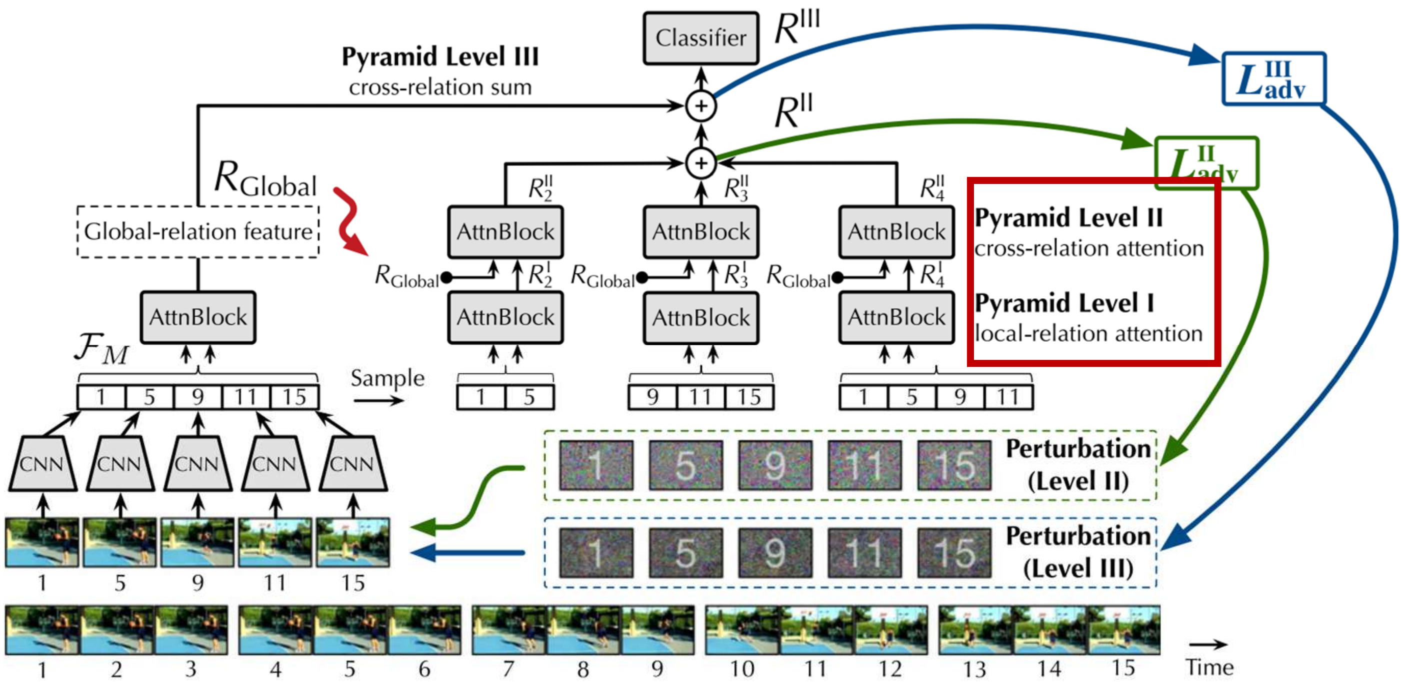


Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : APN

APN(Adversarial Pyramid Network)

To use the global-relation features to guide the generalization of local events
And dynamically **find the events that are highly relevant to the overall video representation**



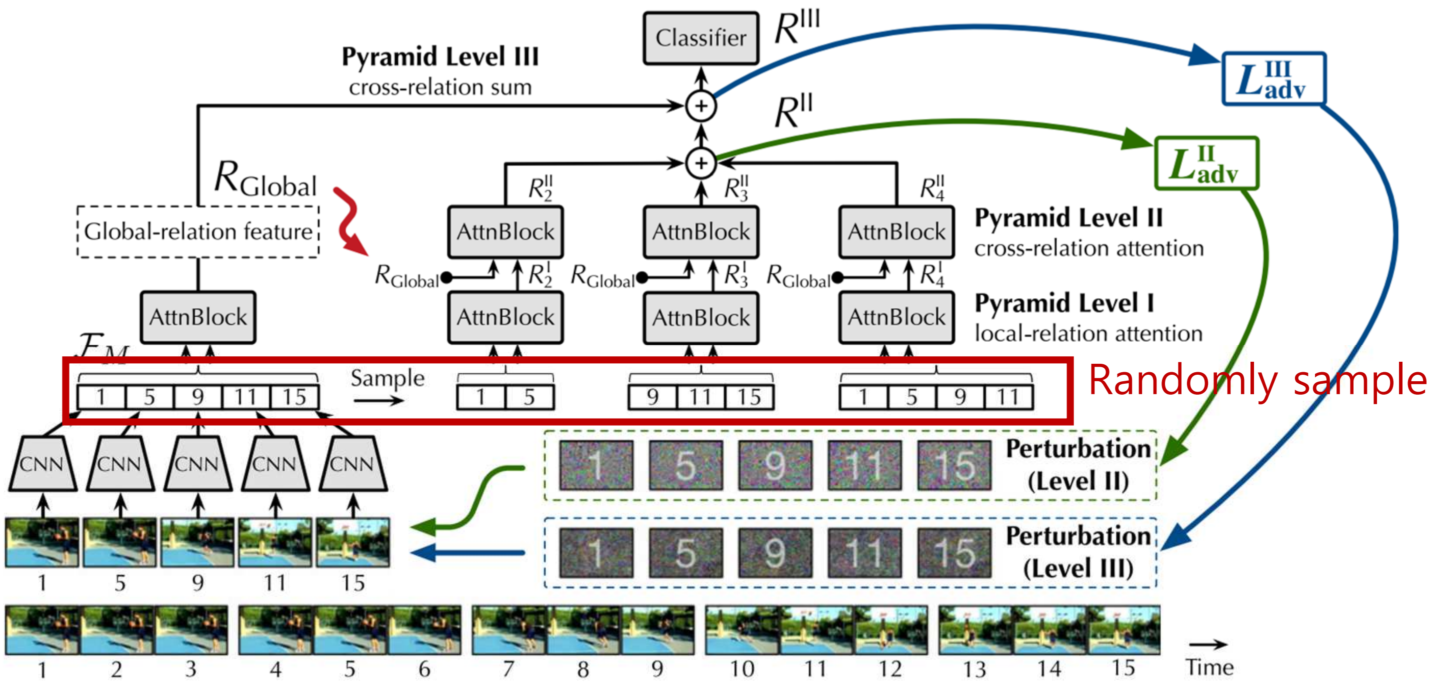
Various temporal relationship

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : APN

Pyramid I : Local-Relation Attention

Obtain $R_1^I, \ldots, R_4^I$: **attention in various local snippets**



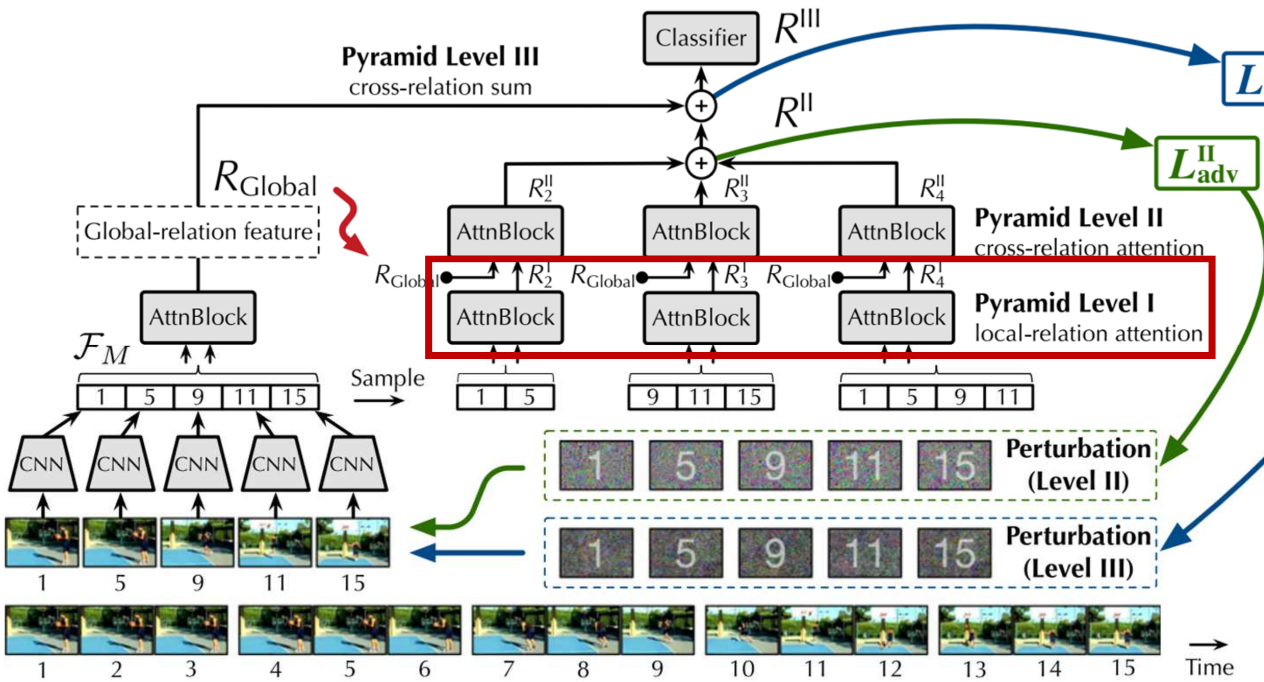
Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : APN

Pyramid I : Local-Relation Attention

Obtain $R_1^I, \dots, R_4^I$: **attention in various local snippets**

Mining local information (more generative)



Attention in snippets

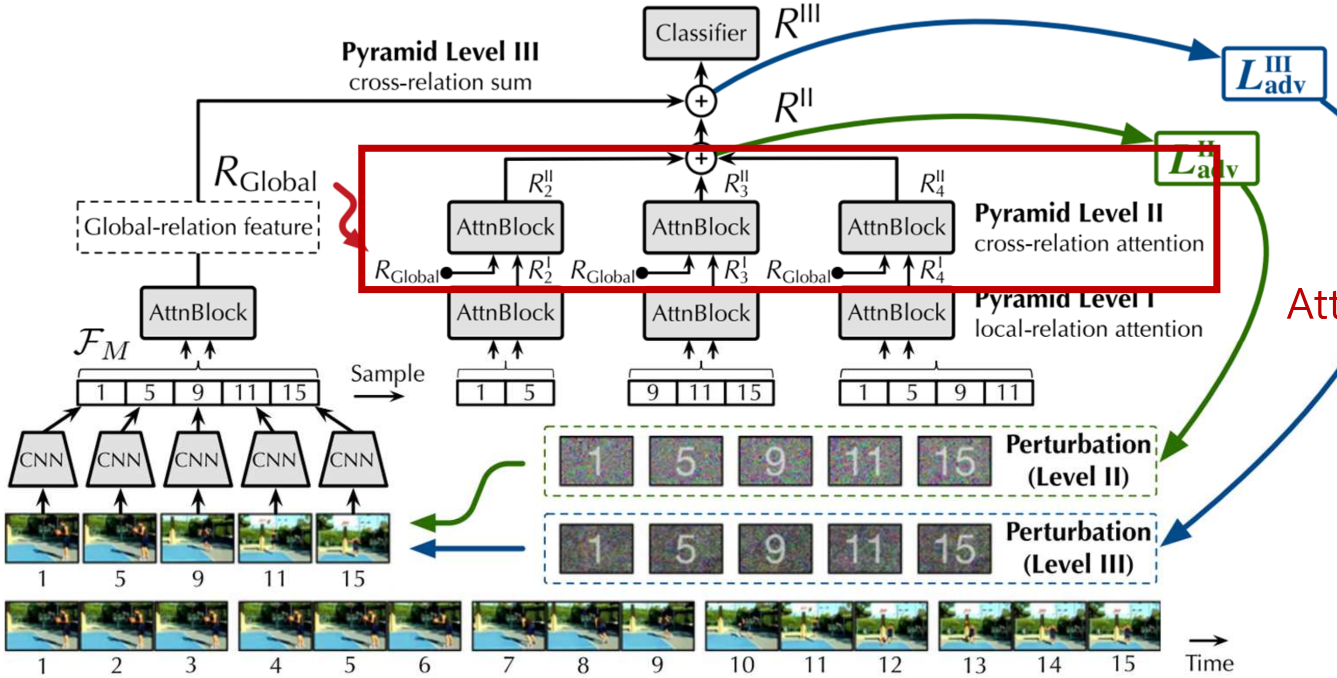
Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : APN

Pyramid II : Cross-Relation Attention

Obtain $R_1^{II}, \dots, R_4^{II}$: **attention in global and local**

Constraint local feature with categorical information
To more discriminative recognition



Attention in global and local

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : RADA

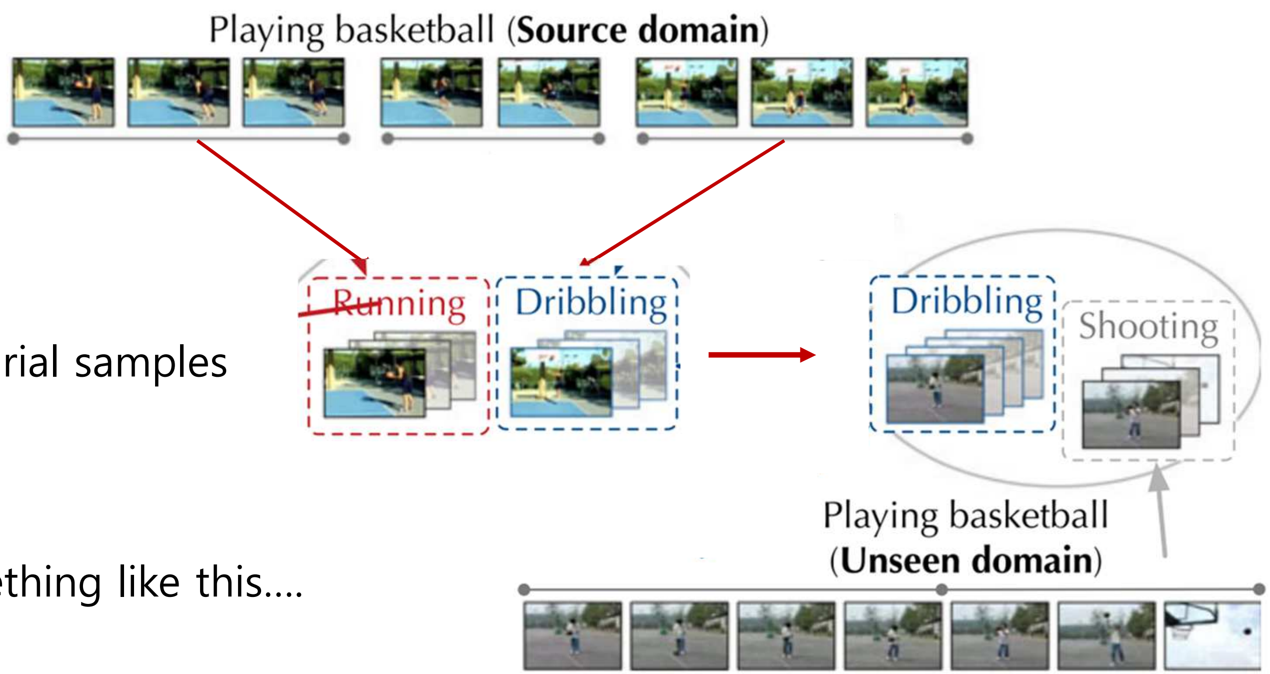
RADA algorithm

Purpose :

- Generate the adversarial samples using training domain samples
  close as possible as to that of the invisible target domain.

Adversarial samples

Something like this….



Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : RADA

RADA algorithm

Purpose :

- Generate the adversarial samples using training domain samples
  close as possible as to that of the invisible target domain.

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : RADA

RADA algorithm

Purpose :

- Generate the adversarial samples using training domain samples close as possible as to that of the invisible target domain.

Phases :

(1) T_max maximization phases :

Generate adversarial examples from the multi-level relational features

VideoDG review : RADA

RADA algorithm

Purpose :

- Generate the adversarial samples using training domain samples close as possible as to that of the invisible target domain.

Phases :

(1) T_max maximization phases :

Generate adversarial examples from the multi-level relational features

(2) **Minimization** phase of **classification error** with a robustness **regularization** :

Train generalizable features to be unaffected by overly divergent new data points

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : RADA

RADA algorithm

Purpose :

- Generate the adversarial samples using training domain samples close as possible as to that of the invisible target domain.

Phases :

(1) T_max maximization phases :

Generate adversarial examples from the multi-level relational features

(2) **Minimization** phase of **classification error** with a robustness **regularization** :

Train generalizable features to be unaffected by overly divergent new data points

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : RADA

RADA algorithm (1) minimize source classification loss

1.1 randomly sample data from batch

Algorithm 2. The RADA Framework for Training APN

Input: A source video dataset $\mathcal{S} = \{(X_i, Y_i)\}_{i=1}^{n}$, the penalty parameter of the transportation cost γ, the robustness regularization parameter λ, the number of maximization phases $T_{\max}$, and the learning rate α

Output: Learned APN weights θ

Initialize $\theta \leftarrow \theta_0$

repeat

$\quad \boxed{(X, Y) \sim \mathcal{S} \quad \triangleright \text{ Randomly sample a batch of data}}$

$\quad (\mathcal{R}_0^{\mathrm{II}}, \mathcal{R}_0^{\mathrm{III}}) = \mathrm{APN}(X)$

$\quad \theta \leftarrow \theta - \alpha \nabla_\theta \ell(h(\mathcal{R}_0^{\mathrm{III}}), Y)$ $\triangleright$ Minimize the classification loss of the source data

$\quad X^{\mathrm{II}} = X; X^{\mathrm{III}} = X$

$\quad$ **for** $t = 1, \ldots, T_{\max}$ **do** $\triangleright$ For each maximization phase

$\quad\quad (\mathcal{R}^{\mathrm{II}}, \_\_) = \mathrm{APN}(X^{\mathrm{II}})$

$\quad\quad X^{\mathrm{II}} \leftarrow X^{\mathrm{II}} + \nabla_X L_{\mathrm{adv}}^{\mathrm{II}}(\theta; (X, Y))$ $\triangleright$ Generate new data according to Eq. (9)

$\quad\quad (\_\_, \mathcal{R}^{\mathrm{III}}) = \mathrm{APN}(X^{\mathrm{III}})$

$\quad\quad X^{\mathrm{III}} \leftarrow X^{\mathrm{III}} + \nabla_X L_{\mathrm{adv}}^{\mathrm{III}}(\theta; (X, Y))$

$\quad$ **end for**

$\quad (\mathcal{R}^{\mathrm{II}}, \_\_) = \mathrm{APN}(X^{\mathrm{II}})$ $\triangleright$ For minimization phase with robust training

$\quad (\_\_, \mathcal{R}^{\mathrm{III}}) = \mathrm{APN}(X^{\mathrm{III}})$

$\quad \theta \leftarrow \theta - \alpha \nabla_\theta (L_{\mathrm{cls}}^{\mathrm{II}}(\theta; (X^{\mathrm{II}}, Y)) + \lambda L_{\mathrm{robust}}^{\mathrm{II}}(\theta; (X^{\mathrm{II}}, Y)))$ $\triangleright$ According to Eq. (12)

$\quad \theta \leftarrow \theta - \alpha \nabla_\theta (L_{\mathrm{cls}}^{\mathrm{III}}(\theta; (X^{\mathrm{III}}, Y)) + \lambda L_{\mathrm{robust}}^{\mathrm{III}}(\theta; (X^{\mathrm{III}}, Y)))$

until Convergence

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : RADA

RADA algorithm (1) minimize source classification loss

1.2 APN inference

Algorithm 2. The RADA Framework for Training APN

Input: A source video dataset $\mathcal{S} = \{(X_i, Y_i)\}_{i=1}^{n}$, the penalty parameter of the transportation cost γ, the robustness regularization parameter λ, the number of maximization phases $T_{\max}$, and the learning rate α

Output: Learned APN weights θ

Initialize $\theta \leftarrow \theta_0$

repeat

  $(X, Y) \sim \mathcal{S}$ ▷ Randomly sample a batch of data

  $\boxed{(\mathcal{R}_0^{\mathrm{II}}, \mathcal{R}_0^{\mathrm{III}}) = \mathrm{APN}(X)}$

  $\theta \leftarrow \theta - \alpha \nabla_\theta \ell(h(\mathcal{R}_0^{\mathrm{III}}), Y)$ ▷ Minimize the classification loss of the source data

  $X^{\mathrm{II}} = X; X^{\mathrm{III}} = X$

  for $t = 1, \ldots, T_{\max}$ **do** ▷ For each maximization phase

    $(\mathcal{R}^{\mathrm{II}}, \_\_) = \mathrm{APN}(X^{\mathrm{II}})$

    $X^{\mathrm{II}} \leftarrow X^{\mathrm{II}} + \nabla_X L_{\mathrm{adv}}^{\mathrm{II}}(\theta; (X, Y))$ ▷ Generate new data according to Eq. (9)

    $(\_\_, \mathcal{R}^{\mathrm{III}}) = \mathrm{APN}(X^{\mathrm{III}})$

    $X^{\mathrm{III}} \leftarrow X^{\mathrm{III}} + \nabla_X L_{\mathrm{adv}}^{\mathrm{III}}(\theta; (X, Y))$

  end for

  $(\mathcal{R}^{\mathrm{II}}, \_\_) = \mathrm{APN}(X^{\mathrm{II}})$ ▷ For minimization phase with robust training

  $(\_\_, \mathcal{R}^{\mathrm{III}}) = \mathrm{APN}(X^{\mathrm{III}})$

  $\theta \leftarrow \theta - \alpha \nabla_\theta (L_{\mathrm{cls}}^{\mathrm{II}}(\theta; (X^{\mathrm{II}}, Y)) + \lambda L_{\mathrm{robust}}^{\mathrm{II}}(\theta; (X^{\mathrm{II}}, Y)))$ ▷ According to Eq. (12)

  $\theta \leftarrow \theta - \alpha \nabla_\theta (L_{\mathrm{cls}}^{\mathrm{III}}(\theta; (X^{\mathrm{III}}, Y)) + \lambda L_{\mathrm{robust}}^{\mathrm{III}}(\theta; (X^{\mathrm{III}}, Y)))$

until Convergence

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : RADA

RADA algorithm (1) minimize source classification loss

1.3 classification loss backward
(on source domain)

Algorithm 2. The RADA Framework for Training APN

Input: A source video dataset $\mathcal{S} = \{(X_i, Y_i)\}_{i=1}^{n}$, the penalty parameter of the transportation cost γ, the robustness regularization parameter λ, the number of maximization phases $T_{\max}$, and the learning rate α

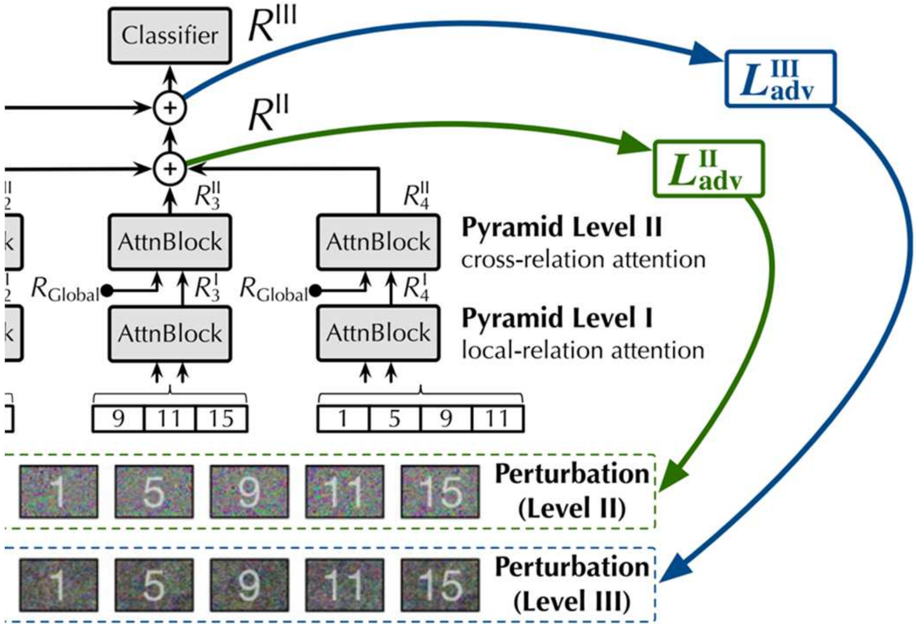
Output: Learned APN weights θ

Initialize $\theta \leftarrow \theta_0$
repeat
  $(X, Y) \sim \mathcal{S}$ ▷ Randomly sample a batch of data
  $(\mathcal{R}_0^{\mathrm{II}}, \mathcal{R}_0^{\mathrm{III}}) = \mathrm{APN}(X)$
  $\boxed{\theta \leftarrow \theta - \alpha \nabla_\theta \ell(h(\mathcal{R}_0^{\mathrm{III}}), Y)}$ ▷ Minimize the classification loss of the source data
  $X^{\mathrm{II}} = X; X^{\mathrm{III}} = X$
  for $t = 1, \ldots, T_{\max}$ **do** ▷ For each maximization phase
    $(\mathcal{R}^{\mathrm{II}}, \_\_) = \mathrm{APN}(X^{\mathrm{II}})$
    $X^{\mathrm{II}} \leftarrow X^{\mathrm{II}} + \nabla_X L_{\mathrm{adv}}^{\mathrm{II}}(\theta; (X, Y))$ ▷ Generate new data according to Eq. (9)
    $(\_\_, \mathcal{R}^{\mathrm{III}}) = \mathrm{APN}(X^{\mathrm{III}})$
    $X^{\mathrm{III}} \leftarrow X^{\mathrm{III}} + \nabla_X L_{\mathrm{adv}}^{\mathrm{III}}(\theta; (X, Y))$
  end for
  $(\mathcal{R}^{\mathrm{II}}, \_\_) = \mathrm{APN}(X^{\mathrm{II}})$ ▷ For minimization phase with robust training
  $(\_\_, \mathcal{R}^{\mathrm{III}}) = \mathrm{APN}(X^{\mathrm{III}})$
  $\theta \leftarrow \theta - \alpha \nabla_\theta (L_{\mathrm{cls}}^{\mathrm{II}}(\theta; (X^{\mathrm{II}}, Y)) + \lambda L_{\mathrm{robust}}^{\mathrm{II}}(\theta; (X^{\mathrm{II}}, Y)))$ ▷ According to Eq. (12)
  $\theta \leftarrow \theta - \alpha \nabla_\theta (L_{\mathrm{cls}}^{\mathrm{III}}(\theta; (X^{\mathrm{III}}, Y)) + \lambda L_{\mathrm{robust}}^{\mathrm{III}}(\theta; (X^{\mathrm{III}}, Y)))$
until Convergence

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : RADA

RADA algorithm (2) maximization phases



2.1 APN inference

Algorithm 2. The RADA Framework for Training APN

Input: A source video dataset $\mathcal{S} = \{(X_i, Y_i)\}_{i=1}^{n}$, the penalty parameter of the transportation cost γ, the robustness regularization parameter λ, the number of maximization phases $T_{\max}$, and the learning rate α

Output: Learned APN weights θ

Initialize $\theta \leftarrow \theta_0$

repeat

$(X, Y) \sim \mathcal{S}$ ▷ Randomly sample a batch of data

$(\mathcal{R}_0^{II}, \mathcal{R}_0^{III}) = \mathrm{APN}(X)$

$\theta \leftarrow \theta - \alpha \nabla_\theta \ell(h(\mathcal{R}_0^{III}), Y)$ ▷ Minimize the classification loss of the source data

$X^{II} = X; X^{III} = X$

for $t = 1, \ldots, T_{\max}$ **do** ▷ For each maximization phase

$\boxed{(\mathcal{R}^{II}, \_\_) = \mathrm{APN}(X^{II})}$

$X^{II} \leftarrow X^{II} + \nabla_X L_{adv}^{II}(\theta; (X, Y))$ ▷ Generate new data according to Eq. (9)

$(\_\_, \mathcal{R}^{III}) = \mathrm{APN}(X^{III})$

$X^{III} \leftarrow X^{III} + \nabla_X L_{adv}^{III}(\theta; (X, Y))$

end for

$(\mathcal{R}^{II}, \_\_) = \mathrm{APN}(X^{II})$ ▷ For minimization phase with robust training

$(\_\_, \mathcal{R}^{III}) = \mathrm{APN}(X^{III})$

$\theta \leftarrow \theta - \alpha \nabla_\theta (L_{cls}^{II}(\theta; (X^{II}, Y)) + \lambda L_{robust}^{II}(\theta; (X^{II}, Y)))$ ▷ According to Eq. (12)

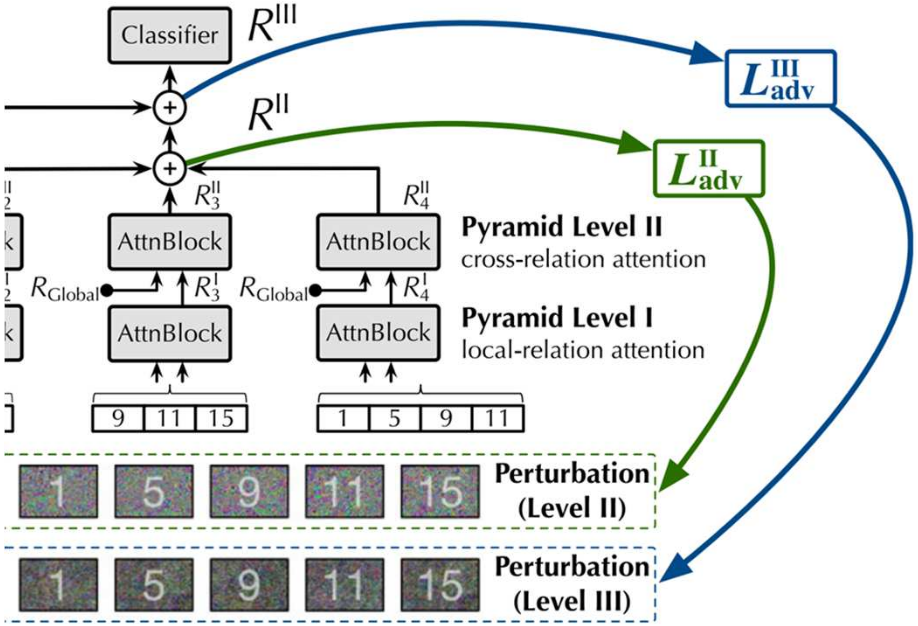
$\theta \leftarrow \theta - \alpha \nabla_\theta (L_{cls}^{III}(\theta; (X^{III}, Y)) + \lambda L_{robust}^{III}(\theta; (X^{III}, Y)))$

until Convergence

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : RADA

RADA algorithm (2) maximization phases



2.2 Adversarial sample generation!

Algorithm 2. The RADA Framework for Training APN

Input: A source video dataset $\mathcal{S} = \{(X_i, Y_i)\}_{i=1}^{n}$, the penalty parameter of the transportation cost γ, the robustness regularization parameter λ, the number of maximization phases $T_{\max}$, and the learning rate α

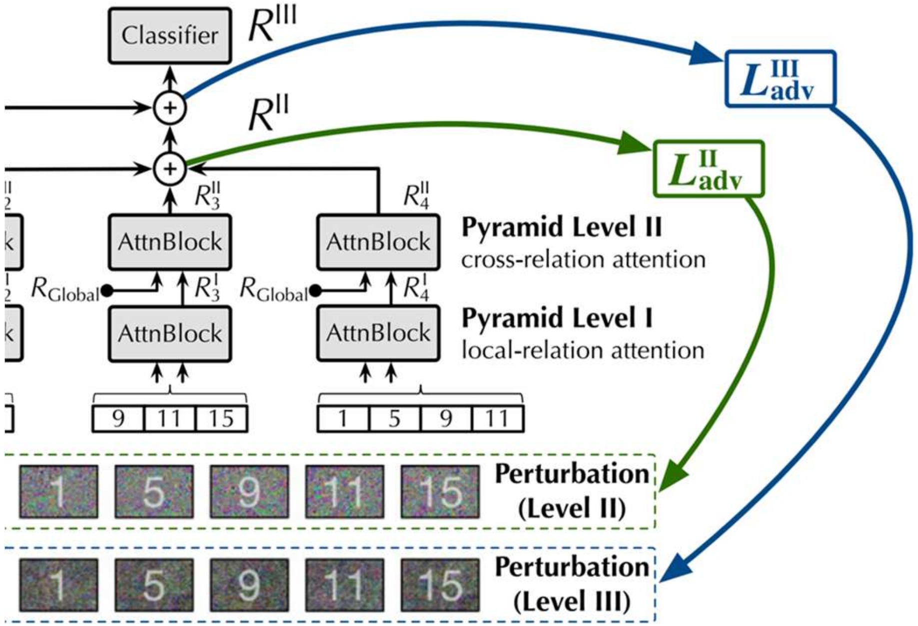
Output: Learned APN weights θ

Initialize $\theta \leftarrow \theta_0$
repeat
  $(X, Y) \sim \mathcal{S}$ ▷ Randomly sample a batch of data
  $(\mathcal{R}_0^{II}, \mathcal{R}_0^{III}) = \mathrm{APN}(X)$
  $\theta \leftarrow \theta - \alpha \nabla_\theta \ell(h(\mathcal{R}_0^{III}), Y)$ ▷ Minimize the classification loss of the source data
  $X^{II} = X; X^{III} = X$
  for $t = 1, \ldots, T_{\max}$ **do** ▷ For each maximization phase
    $(\mathcal{R}^{II}, \_\_) = \mathrm{APN}(X^{II})$
    $X^{II} \leftarrow X^{II} + \nabla_X L^{II}_{adv}(\theta; (X, Y))$ ▷ Generate new data according to Eq. (9)
    $(\_\_, \mathcal{R}^{III}) = \mathrm{APN}(X^{III})$
    $X^{III} \leftarrow X^{III} + \nabla_X L^{III}_{adv}(\theta; (X, Y))$
  end for
  $(\mathcal{R}^{II}, \_\_) = \mathrm{APN}(X^{II})$ ▷ For minimization phase with robust training
  $(\_\_, \mathcal{R}^{III}) = \mathrm{APN}(X^{III})$
  $\theta \leftarrow \theta - \alpha \nabla_\theta (L^{II}_{cls}(\theta; (X^{II}, Y)) + \lambda L^{II}_{robust}(\theta; (X^{II}, Y)))$ ▷ According to Eq. (12)
  $\theta \leftarrow \theta - \alpha \nabla_\theta (L^{III}_{cls}(\theta; (X^{III}, Y)) + \lambda L^{III}_{robust}(\theta; (X^{III}, Y)))$
until Convergence

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : RADA

RADA algorithm (2) maximization phases



2.2 Adversarial sample generation!

Algorithm 2. The RADA Framework for Training APN

Input: A source video dataset $\mathcal{S} = \{(X_i, Y_i)\}_{i=1}^{n}$, the penalty parameter of the transportation cost γ, the robustness regularization parameter λ, the number of maximization phases $T_{\max}$, and the learning rate α

Output: Learned APN weights θ

Initialize $\theta \leftarrow \theta_0$

repeat

 $(X, Y) \sim \mathcal{S}$ ▷ Randomly sample a batch of data

 $(\mathcal{R}_0^{II}, \mathcal{R}_0^{III}) = \text{APN}(X)$

 $\theta \leftarrow \theta - \alpha \nabla_\theta \ell(h(\mathcal{R}_0^{III}), Y)$ ▷ Minimize the classification loss of the source data

 $X^{II} = X; X^{III} = X$

 for $t = 1, \ldots, T_{\max}$ **do** ▷ For each maximization phase

  $(\mathcal{R}^{II}, \_) = \text{APN}(X^{II})$

  $X^{II} \leftarrow X^{II} + \nabla_X L_{adv}^{II}(\theta; (X, Y))$ ▷ Generate new data according to Eq. (9)

  $(\_, \mathcal{R}^{III}) = \text{APN}(X^{III})$

  $X^{III} \leftarrow X^{III} + \nabla_X L_{adv}^{III}(\theta; (X, Y))$

 end for

 $(\mathcal{R}^{II}, \_) = \text{APN}(X^{II})$ ▷ For minimization phase with robust training

 $(\_, \mathcal{R}^{III}) = \text{APN}(X^{III})$

 $\theta \leftarrow \theta - \alpha \nabla_\theta (L_{cls}^{II}(\theta; (X^{II}, Y)) + \lambda L_{robust}^{II}(\theta; (X^{II}, Y)))$ ▷ According to Eq. (12)

 $\theta \leftarrow \theta - \alpha \nabla_\theta (L_{cls}^{III}(\theta; (X^{III}, Y)) + \lambda L_{robust}^{III}(\theta; (X^{III}, Y)))$

until Convergence

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : RADA

RADA algorithm (2) maximization phases

- Make perturbations

$$X^{\mathrm{II}} \leftarrow \underbrace{X^{\mathrm{II}}}_{\text{input}} + \underbrace{\nabla_X L^{\mathrm{II}}_{\mathrm{adv}}(\theta; (X, Y))}_{\text{perturbations}}$$

2.1 Add perturbations to adversarial samples

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : RADA

RADA algorithm (2) maximization phases

- Make perturbations

$$X^{\mathrm{II}} \leftarrow \underbrace{X^{\mathrm{II}}}_{\text{input}} + \underbrace{\nabla_X L^{\mathrm{II}}_{\mathrm{adv}}(\theta; (X, Y))}_{\text{perturbations}}$$

$$L^k_{\mathrm{adv}}(\theta; (X, Y)) := \sup_{X \in \mathcal{X}} \left\{ \ell\left(h(\mathcal{R}^k), Y\right) - \gamma c\left(\mathcal{R}^k, \mathcal{R}^k_0\right) \right\}, \qquad (9)$$

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : RADA

RADA algorithm (2) maximization phases

- Make perturbations

$$X^{\mathrm{II}} \leftarrow \underbrace{X^{\mathrm{II}}}_{\text{input}} + \underbrace{\nabla_X L^{\mathrm{II}}_{\mathrm{adv}}(\theta; (X, Y))}_{\text{perturbations}}$$

$$L^k_{\mathrm{adv}}(\theta; (X, Y)) := \sup_{X \in \mathcal{X}} \left\{ \boxed{\ell\big(h(\mathcal{R}^k), Y\big)} - \gamma c\big(\mathcal{R}^k, \mathcal{R}^k_0\big) \right\}, \qquad (9)$$

K=1,2

$l(h(R^k), Y)$

Just cross entropy loss With FC layer h

: The **added perturbation** **must not move away from semantic**.

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : RADA

RADA algorithm (2) maximization phases
- Make perturbations

$$X^{\mathrm{II}} \leftarrow \underbrace{X^{\mathrm{II}}}_{\text{input}} + \underbrace{\nabla_X L^{\mathrm{II}}_{\mathrm{adv}}(\theta; (X, Y))}_{\text{perturbations}}$$

$$L^k_{\mathrm{adv}}(\theta; (X, Y)) := \sup_{X \in \mathcal{X}} \{\ell(h(\mathcal{R}^k), Y) - \boxed{\gamma c(\mathcal{R}^k, \mathcal{R}^k_0)}\}, \qquad (9)$$

K=1,2

$\gamma c(R^k, R^k_0)$

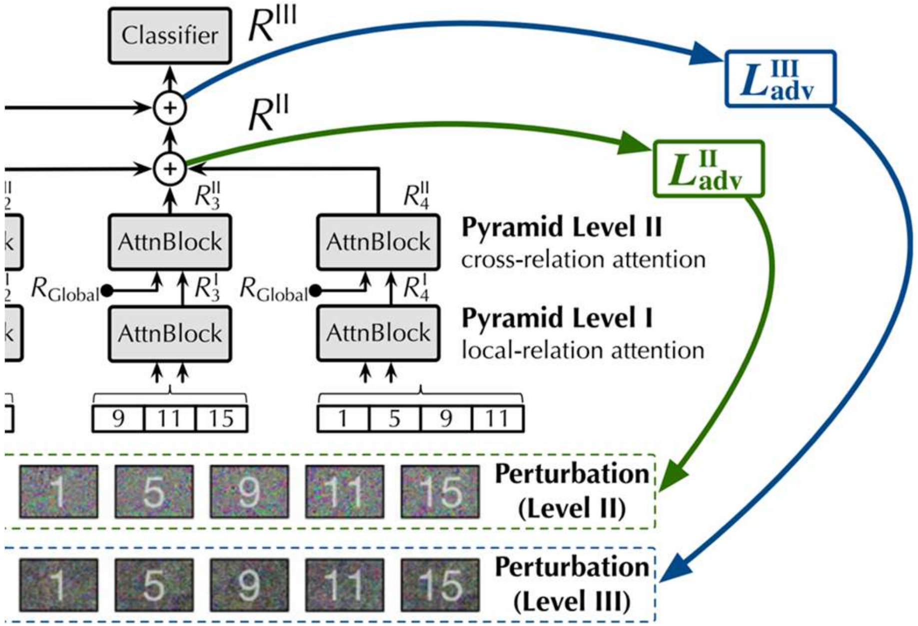
Just weighted mse loss between R^k, R^k_0
: Regulates features so that they are not too far from aggregated features at each level **to maintain consistency in advertising samples**

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : RADA

RADA algorithm (2) maximization phases

- Make perturbations



2.2 Add perturbation few times to adversarial sample

Algorithm 2. The RADA Framework for Training APN

Input: A source video dataset $\mathcal{S} = \{(X_i, Y_i)\}_{i=1}^n$, the penalty parameter of the transportation cost γ, the robustness regularization parameter λ, the number of maximization phases $T_{\max}$, and the learning rate α

Output: Learned APN weights θ

Initialize $\theta \leftarrow \theta_0$

repeat

$(X, Y) \sim \mathcal{S}$ ▷ Randomly sample a batch of data

$(\mathcal{R}_0^{\mathrm{II}}, \mathcal{R}_0^{\mathrm{III}}) = \mathrm{APN}(X)$

$\theta \leftarrow \theta - \alpha \nabla_\theta \ell(h(\mathcal{R}_0^{\mathrm{III}}), Y)$ ▷ Minimize the classification loss of the source data

$X^{\mathrm{II}} = X; X^{\mathrm{III}} = X$

for $t = 1, \ldots, T_{\max}$ **do** ▷ For each maximization phase

$(\mathcal{R}^{\mathrm{II}}, \_) = \mathrm{APN}(X^{\mathrm{II}})$

$X^{\mathrm{II}} \leftarrow X^{\mathrm{II}} + \nabla_X L_{\mathrm{adv}}^{\mathrm{II}}(\theta; (X, Y))$ ▷ Generate new data according to Eq. (9)

$(\_, \mathcal{R}^{\mathrm{III}}) = \mathrm{APN}(X^{\mathrm{III}})$

$X^{\mathrm{III}} \leftarrow X^{\mathrm{III}} + \nabla_X L_{\mathrm{adv}}^{\mathrm{III}}(\theta; (X, Y))$

end for

$(\mathcal{R}^{\mathrm{II}}, \_) = \mathrm{APN}(X^{\mathrm{II}})$ ▷ For minimization phase with robust training

$(\_, \mathcal{R}^{\mathrm{III}}) = \mathrm{APN}(X^{\mathrm{III}})$

$\theta \leftarrow \theta - \alpha \nabla_\theta (L_{\mathrm{cls}}^{\mathrm{II}}(\theta; (X^{\mathrm{II}}, Y)) + \lambda L_{\mathrm{robust}}^{\mathrm{II}}(\theta; (X^{\mathrm{II}}, Y)))$ ▷ According to Eq. (12)

$\theta \leftarrow \theta - \alpha \nabla_\theta (L_{\mathrm{cls}}^{\mathrm{III}}(\theta; (X^{\mathrm{III}}, Y)) + \lambda L_{\mathrm{robust}}^{\mathrm{III}}(\theta; (X^{\mathrm{III}}, Y)))$

until Convergence

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

VideoDG review : RADA

RADA algorithm (3) minimization phases
- Compensate overly divergent generated samples

$$L^k_{\text{cls}}(\theta; (X^k, Y)) := \underbrace{\ell(h(\mathcal{R}^k), Y)}_{\text{classify adversarial examples}}, \tag{10}$$

$$L^k_{\text{robust}}(\theta; (X^k, Y)) := \underbrace{\ell\Big(h(\mathcal{R}^k), h\Big(\mathcal{R}^{\text{III}}_0\Big)\Big)}_{\text{robustness regularization}}, \tag{11}$$

$$L_{\min} := L^k_{\text{cls}}(\theta; (X^k, Y)) + \lambda L^k_{\text{robust}}(\theta; (X^k, Y)), \tag{12}$$

3.1 Cross entropy loss on adversarial example

Algorithm 2. The RADA Framework for Training APN

Input: A source video dataset $\mathcal{S} = \{(X_i, Y_i)\}_{i=1}^n$, the penalty parameter of the transportation cost γ, the robustness regularization parameter λ, the number of maximization phases $T_{\max}$, and the learning rate α

Output: Learned APN weights θ

Initialize $\theta \leftarrow \theta_0$
repeat
  $(X, Y) \sim \mathcal{S}$ ▷ Randomly sample a batch of data
  $(\mathcal{R}^{\text{II}}_0, \mathcal{R}^{\text{III}}_0) = \text{APN}(X)$
  $\theta \leftarrow \theta - \alpha \nabla_\theta \ell(h(\mathcal{R}^{\text{III}}_0), Y)$ ▷ Minimize the classification loss of the source data
  $X^{\text{II}} = X; X^{\text{III}} = X$
  for $t = 1, \ldots, T_{\max}$ **do** ▷ For each maximization phase
    $(\mathcal{R}^{\text{II}}, \_) = \text{APN}(X^{\text{II}})$
    $X^{\text{II}} \leftarrow X^{\text{II}} + \nabla_X L^{\text{II}}_{\text{adv}}(\theta; (X, Y))$ ▷ Generate new data according to Eq. (9)
    $(\_, \mathcal{R}^{\text{III}}) = \text{APN}(X^{\text{III}})$
    $X^{\text{III}} \leftarrow X^{\text{III}} + \nabla_X L^{\text{III}}_{\text{adv}}(\theta; (X, Y))$
  end for
  $(\mathcal{R}^{\text{II}}, \_) = \text{APN}(X^{\text{II}})$ ▷ For minimization phase with robust training
  $(\_, \mathcal{R}^{\text{III}}) = \text{APN}(X^{\text{III}})$
  $\theta \leftarrow \theta - \alpha \nabla_\theta (L^{\text{II}}_{\text{cls}}(\theta; (X^{\text{II}}, Y)) + \lambda L^{\text{II}}_{\text{robust}}(\theta; (X^{\text{II}}, Y)))$ ▷ According to Eq. (12)
  $\theta \leftarrow \theta - \alpha \nabla_\theta (L^{\text{III}}_{\text{cls}}(\theta; (X^{\text{III}}, Y)) + \lambda L^{\text{III}}_{\text{robust}}(\theta; (X^{\text{III}}, Y)))$
until Convergence

VideoDG review : RADA

RADA algorithm (3) minimization phases
- Compensate overly divergent generated samples

$$L^k_{\text{cls}}(\theta; (X^k, Y)) := \underbrace{\ell(h(\mathcal{R}^k), Y)}_{\text{classify adversarial examples}}, \tag{10}$$

$$\boxed{L^k_{\text{robust}}(\theta; (X^k, Y)) := \underbrace{\ell\Big(h(\mathcal{R}^k), h\Big(\mathcal{R}^{\text{III}}_0\Big)\Big)}_{\text{robustness regularization}},} \tag{11}$$

$$L_{\min} := L^k_{\text{cls}}(\theta; (X^k, Y)) + \lambda L^k_{\text{robust}}(\theta; (X^k, Y)), \tag{12}$$

3.2 robustness regularization
Cross-entropy loss between
predicted distribution from Adversarial sample and
Predicted distribution from source(original) sample

Algorithm 2. The RADA Framework for Training APN

Input: A source video dataset $\mathcal{S} = \{(X_i, Y_i)\}_{i=1}^n$, the penalty parameter of the transportation cost γ, the robustness regularization parameter λ, the number of maximization phases $T_{\max}$, and the learning rate α
Output: Learned APN weights θ

```
Initialize θ ← θ_0
repeat
    (X, Y) ~ S                         ▷ Randomly sample a batch of data
    (R_0^II, R_0^III) = APN(X)
    θ ← θ − α∇_θ ℓ(h(R_0^III), Y)      ▷ Minimize the classification loss of the source data
    X^II = X; X^III = X
    for t = 1, ..., T_max do           ▷ For each maximization phase
        (R^II, _) = APN(X^II)
        X^II ← X^II + ∇_X L_adv^II(θ; (X, Y))   ▷ Generate new data according to Eq. (9)
        (_, R^III) = APN(X^III)
        X^III ← X^III + ∇_X L_adv^III(θ; (X, Y))
    end for
    (R^II, _) = APN(X^II)              ▷ For minimization phase with robust training
    (_, R^III) = APN(X^III)
    θ ← θ − α∇_θ(L_cls^II(θ; (X^II, Y)) + λL_robust^II(θ; (X^II, Y)))   ▷ According to Eq. (12)
    θ ← θ − α∇_θ(L_cls^III(θ; (X^III, Y)) + λL_robust^III(θ; (X^III, Y)))
until Convergence
```

Yao, Zhiyu, et al. "Videodg: Generalizing temporal relations in videos to novel domains." *IEEE TPAMI, 2021*

Conclusion

- The i.i.d. assumption are the fundamental but crucial assumption when training machine learning model
- However the i.i.d. assumption are easily violated in real world.
- The Domain Generalization(DG) has been proposed for solving these problem.
- However, the Video Domain Generalization is an un-charted area
- In Video, the temporal and spatial domain shifts simultaneously occurs
- Therefore temporal domain shifts are main challenge in video domain generalization
- In previous work, the temporal attention on diverse relation in time axis is tried.
- The adversarial samples are one of the method to solve VDG.

UST seminar

Towards domain-agnostic Video action recognition

Hyungmin Kim
UST-ETRI
Ph.D. student
khm159@etri.re.kr/ust.ac.kr
21th Sep, 2023