Generative Adversarial Network
and its Applications to Signal Processing and Natural Language Processing

Hung-yi Lee and Yu Tsao

Outline

Part I: General Introduction of Generative Adversarial Network (GAN)

Part II: Applications on Signal Processing

Part III: Applications on Natural Language Processing

Generative Adversarial Network
and its Applications to Signal Processing and Natural Language Processing
Part I: General Introduction

All Kinds of GAN …

https://github.com/hindupuravinash/the-gan-zoo

GAN

ACGAN

BGAN

CGAN

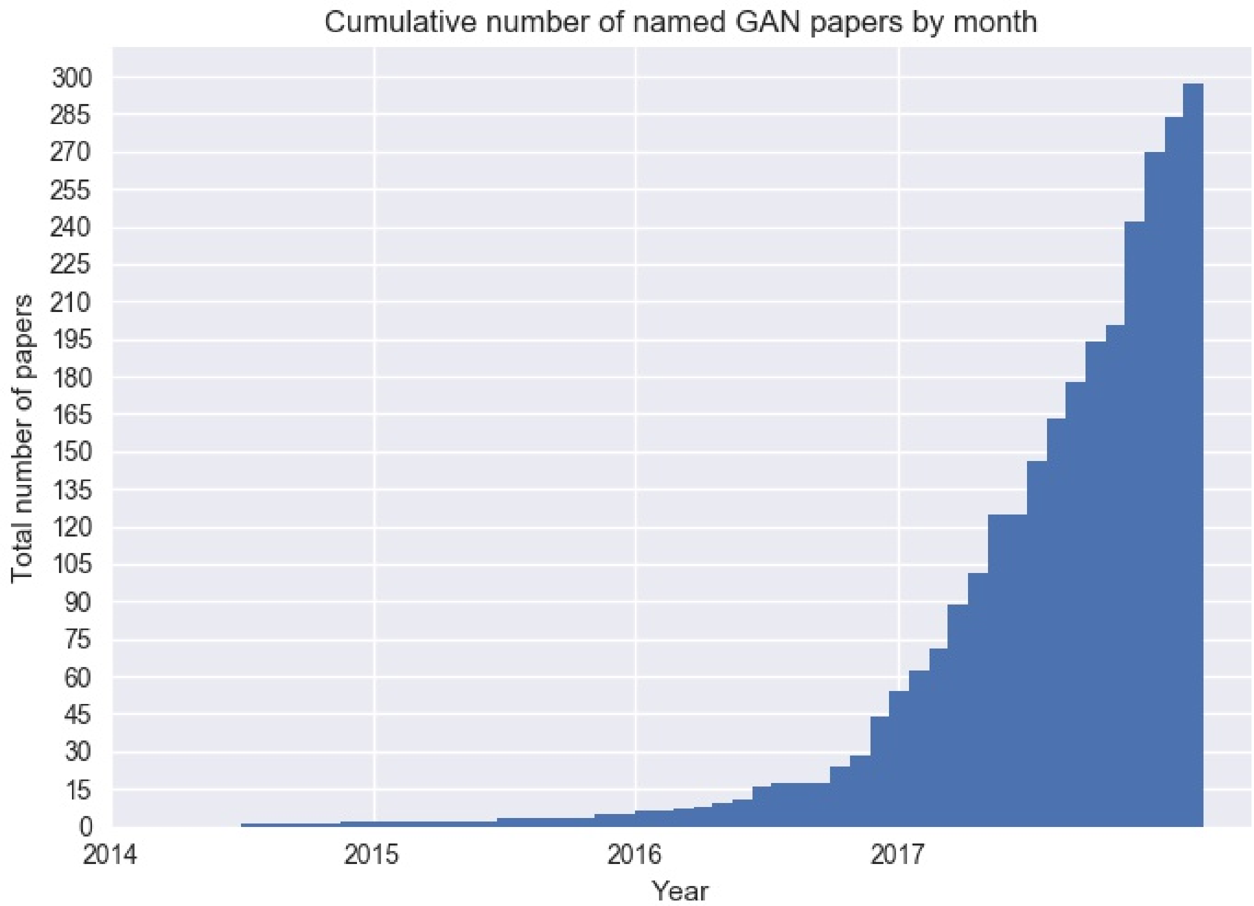
DCGAN

EBGAN

fGAN

GoGAN

⋮

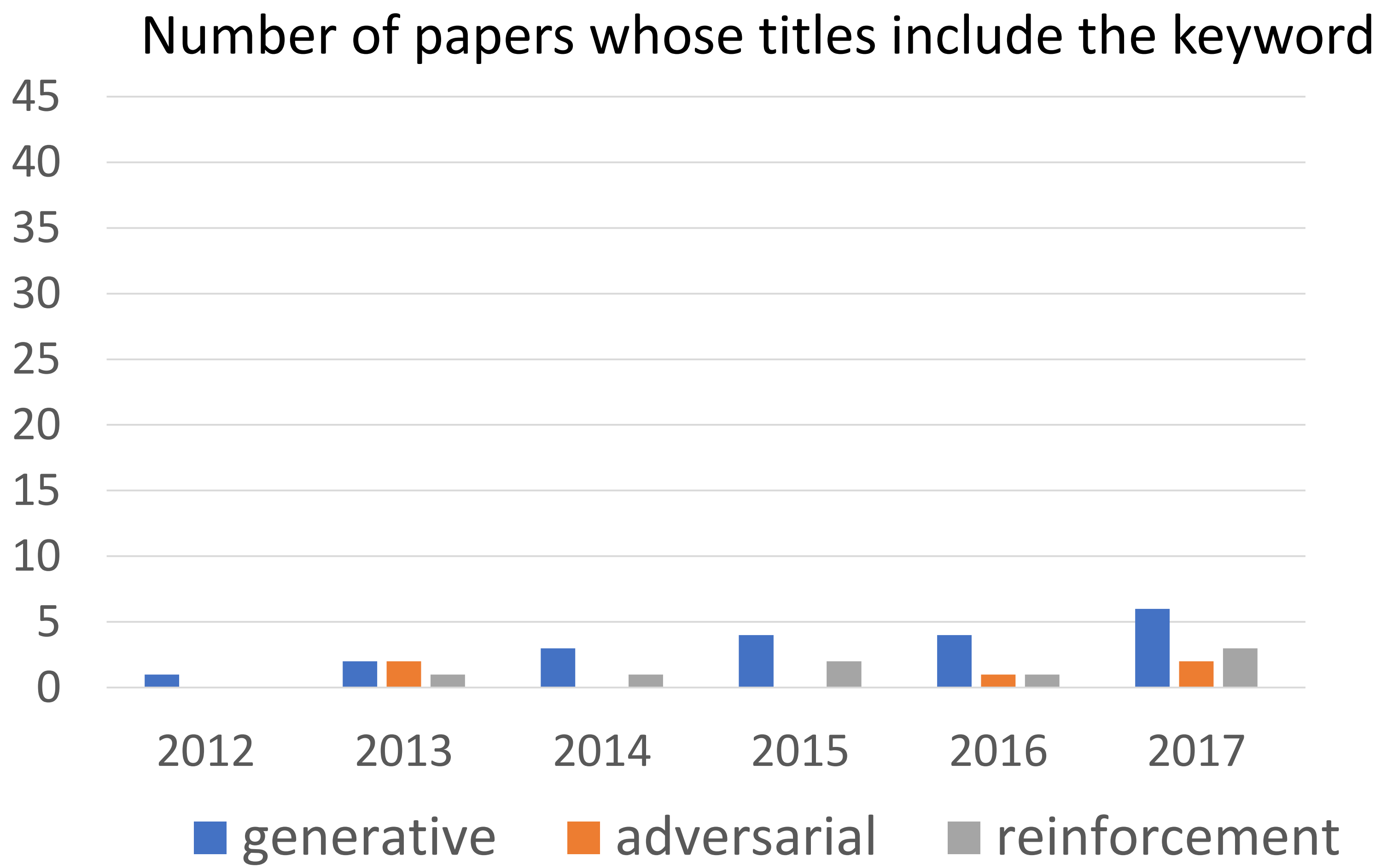


Mihaela Rosca, Balaji Lakshminarayanan, David Warde-Farley, Shakir Mohamed, "Variational Approaches for Auto-Encoding Generative Adversarial Networks", arXiv, 2017

[2]We use the Greek α prefix for α-GAN, as AEGAN and most other Latin prefixes seem to have been taken https://deephunt.in/the-gan-zoo-79597dc8c347.

ICASSP

Keyword search on session index page, so session name is included.



Outline of Part 1

Generation by GAN

Conditional Generation

Unsupervised Conditional Generation

Relation to Reinforcement Learning

Outline of Part 1

Generation by GAN

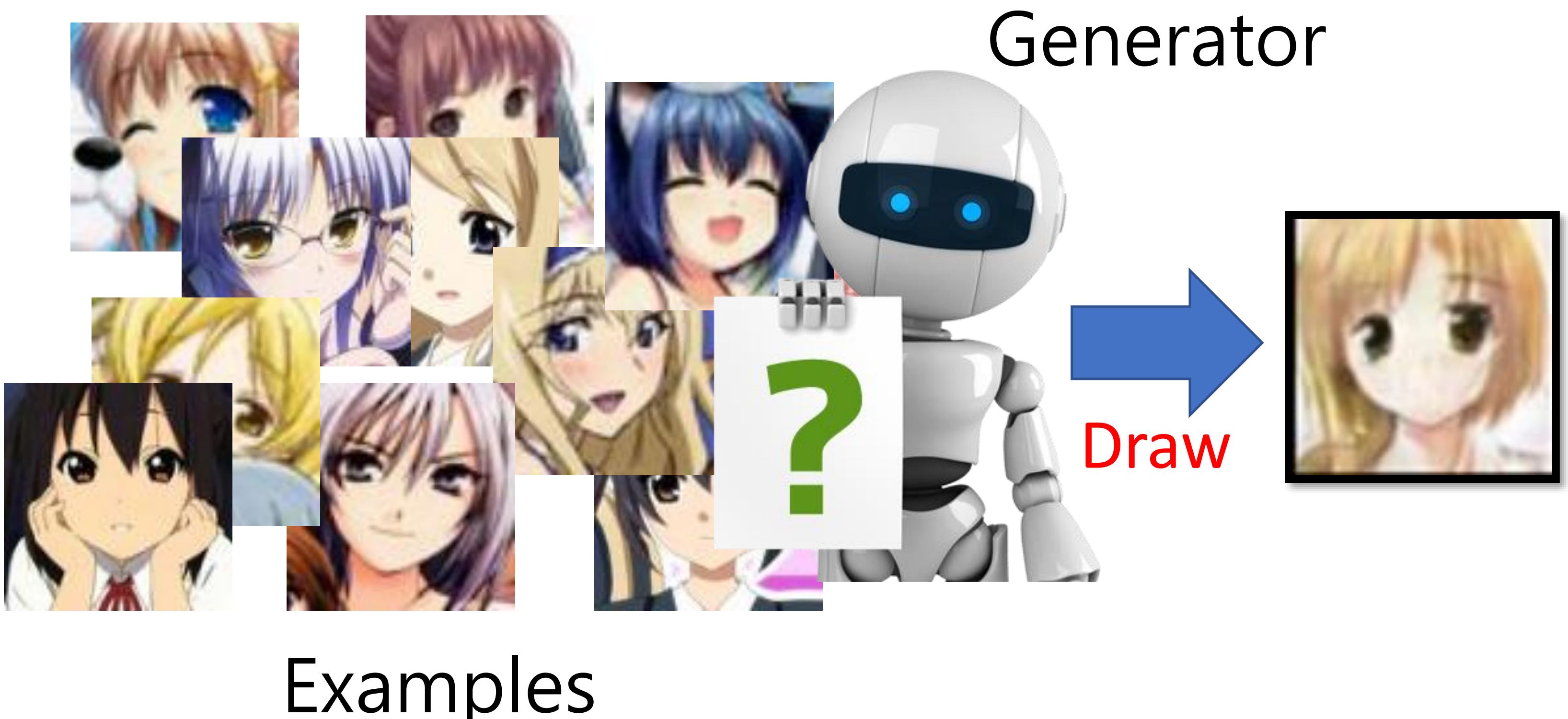
- Image Generation as Example
- Theory behind GAN
- Issues and Possible Solutions

Conditional Generation

Unsupervised Conditional Generation

Relation to Reinforcement Learning

Anime Face Generation



Powered by: http://mattya.github.io/chainer-DCGAN/

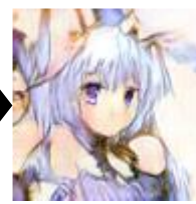
Basic Idea of GAN

It is a neural network (NN), or a function.

vector → Generator → image

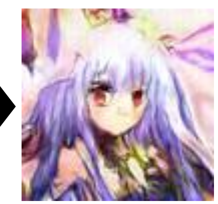
high dimensional vector

$$\begin{bmatrix} 0.1 \\ -3 \\ \vdots \\ 2.4 \\ 0.9 \end{bmatrix}$$ → Generator →



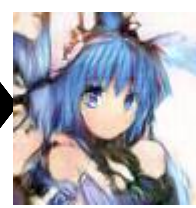
Each dimension of input vector represents some characteristics.

$$\begin{bmatrix} \boxed{3} \\ -3 \\ \vdots \\ 2.4 \\ 0.9 \end{bmatrix}$$ → Generator →



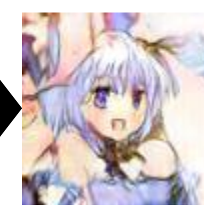
Longer hair

$$\begin{bmatrix} 0.1 \\ 2.1 \\ \vdots \\ \boxed{5.4} \\ 0.9 \end{bmatrix}$$ → Generator →



blue hair

$$\begin{bmatrix} 0.1 \\ -3 \\ \vdots \\ 2.4 \\ \boxed{3.5} \end{bmatrix}$$ → Generator →



Open mouth

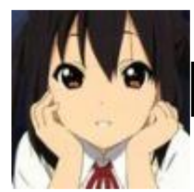
Basic Idea of GAN

It is a neural network (NN), or a function.

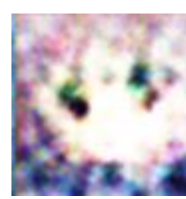
image → Discri-minator → scalar

Larger value means real, smaller value means fake.

 → Discri-minator → 1.0

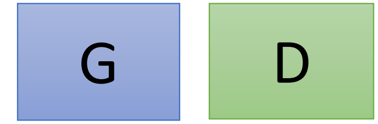
 → Discri-minator → 1.0

 → Discri-minator → 0.1

 → Discri-minator → 0.1

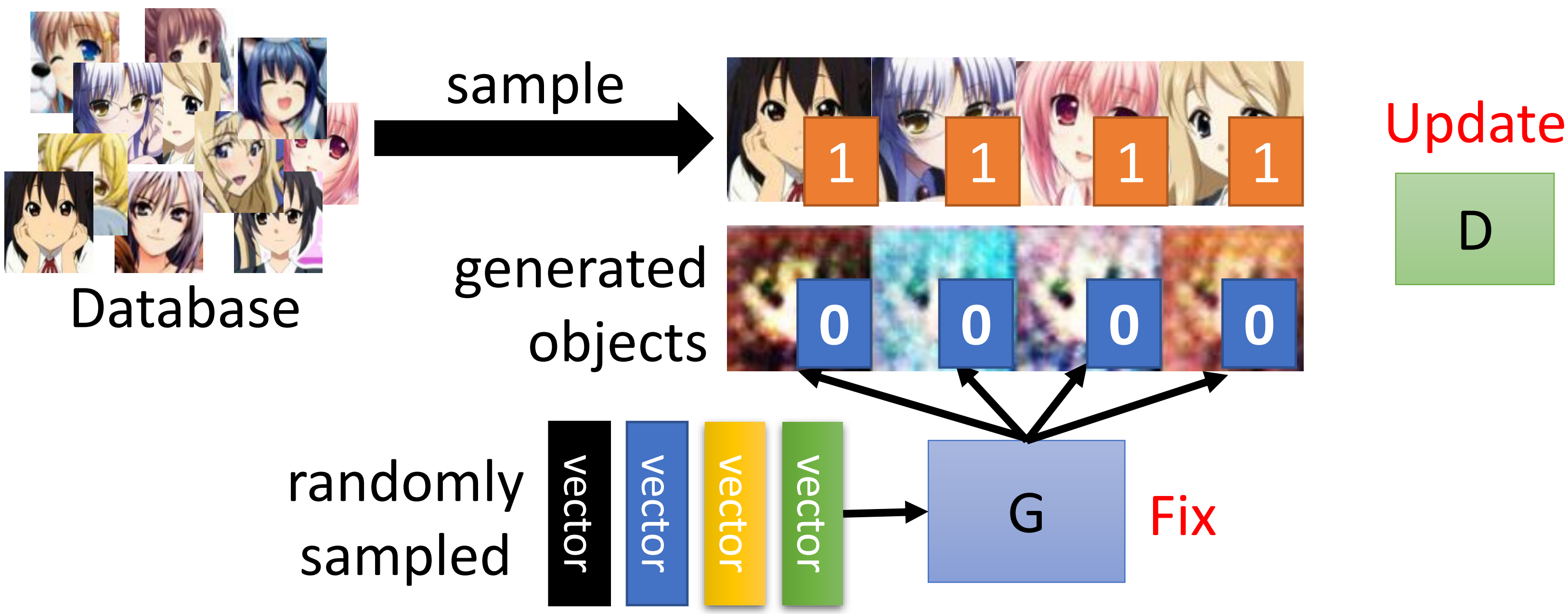
Algorithm

- Initialize generator and discriminator



- In each training iteration:

Step 1: Fix generator G, and update discriminator D



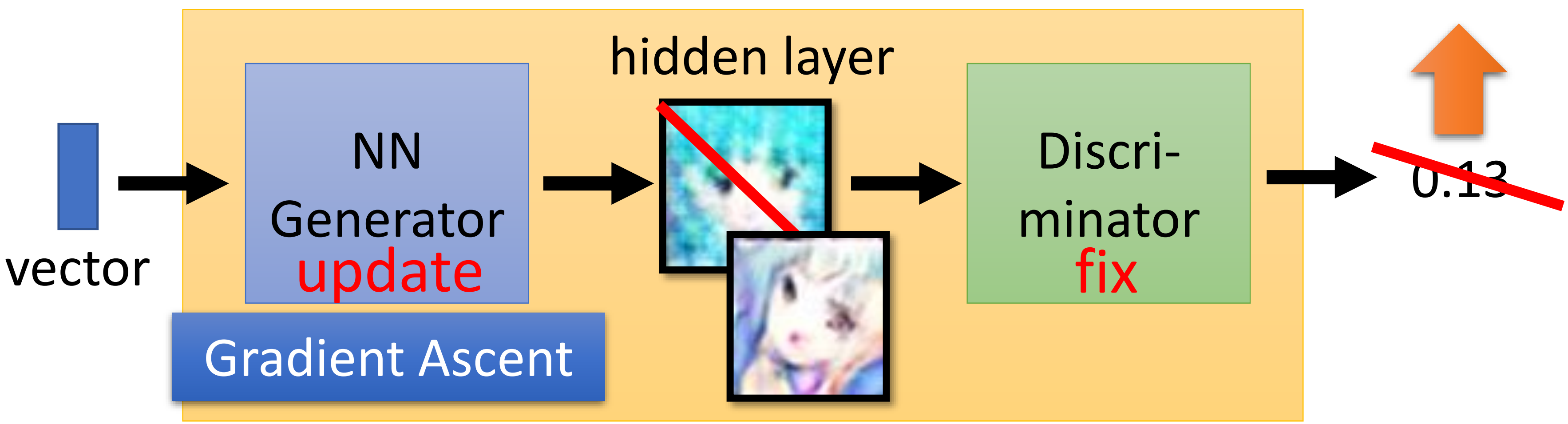
Discriminator learns to assign high scores to real objects and low scores to generated objects.

Algorithm

- Initialize generator and discriminator G D
- In each training iteration:

Step 2: Fix discriminator D, and update generator G

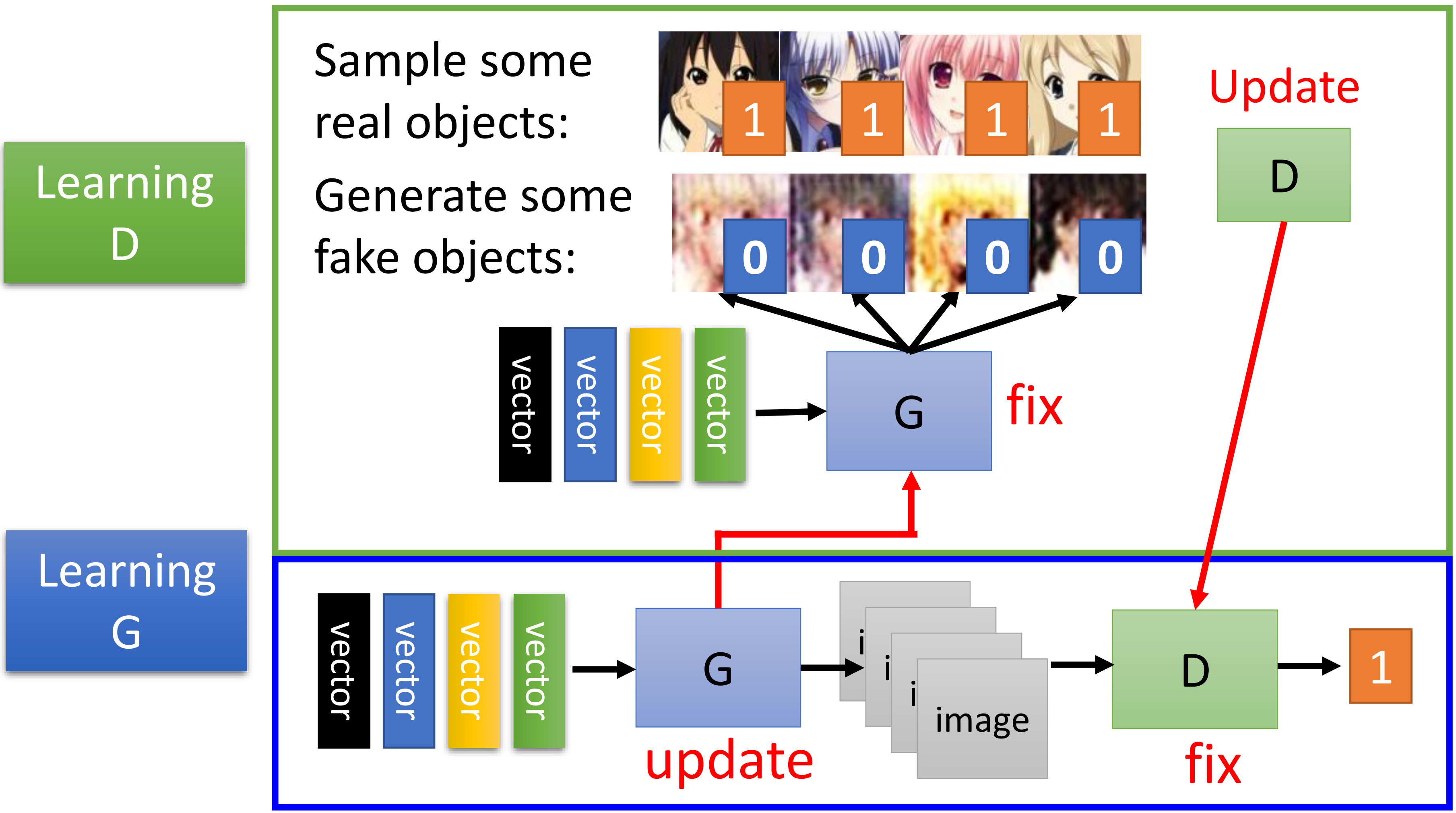
Generator learns to “fool” the discriminator



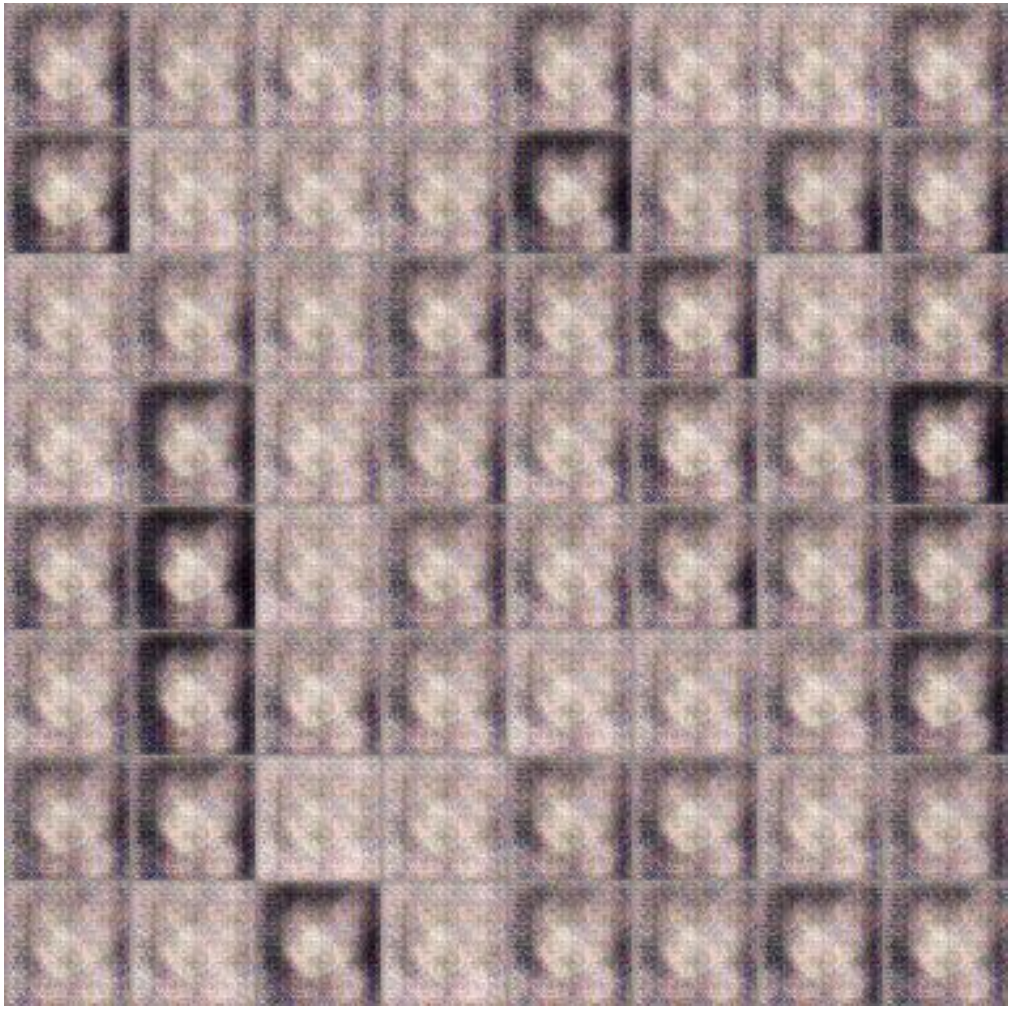
large network

Algorithm

- Initialize generator and discriminator G D
- In each training iteration:



Anime Face Generation



100 updates

Source of training data: https://zhuanlan.zhihu.com/p/24767059

Anime Face Generation

1000 updates

Anime Face Generation

2000 updates

Anime Face Generation

5000 updates

Anime Face Generation

10,000 updates

Anime Face Generation

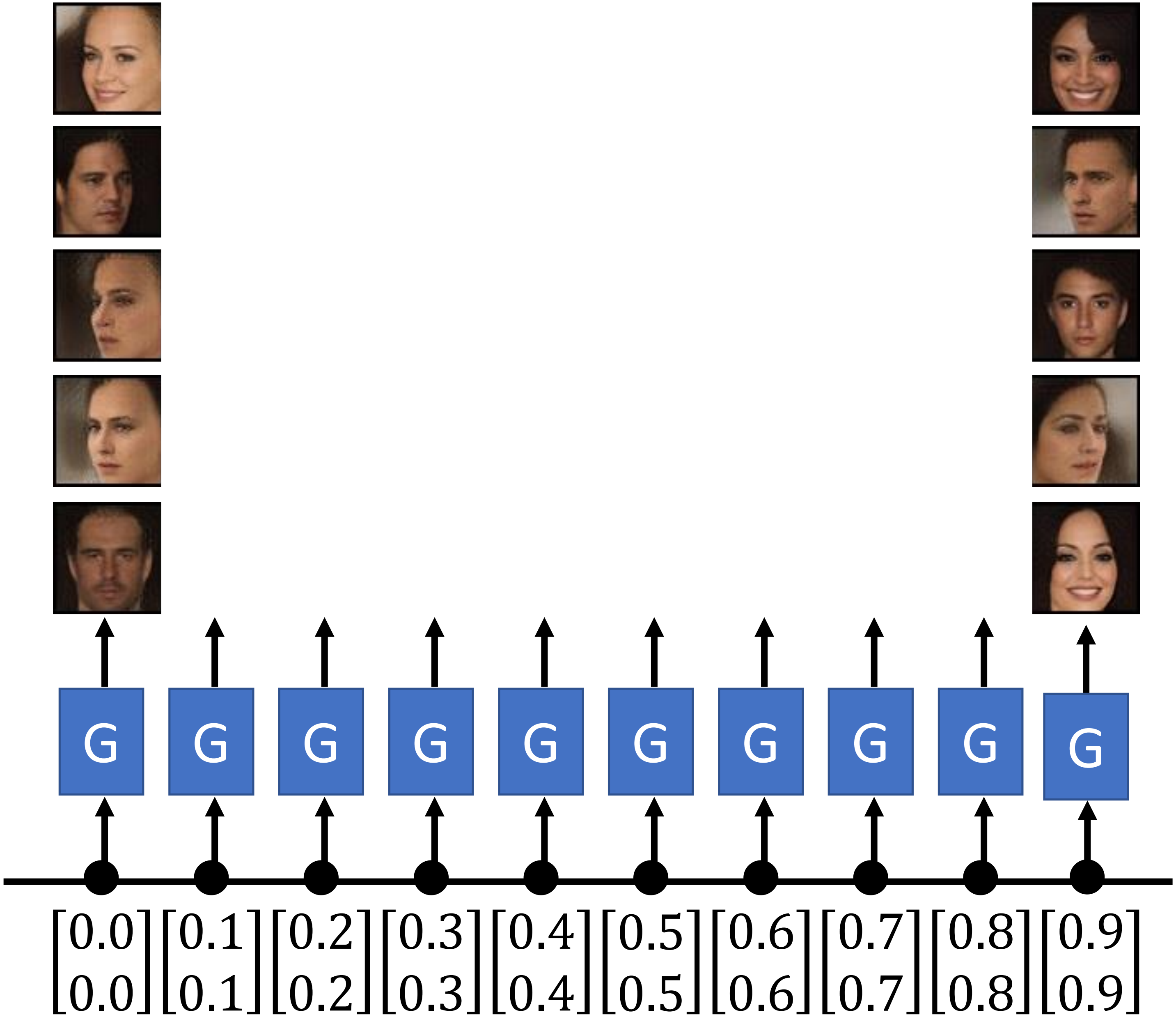
20,000 updates

Anime Face Generation

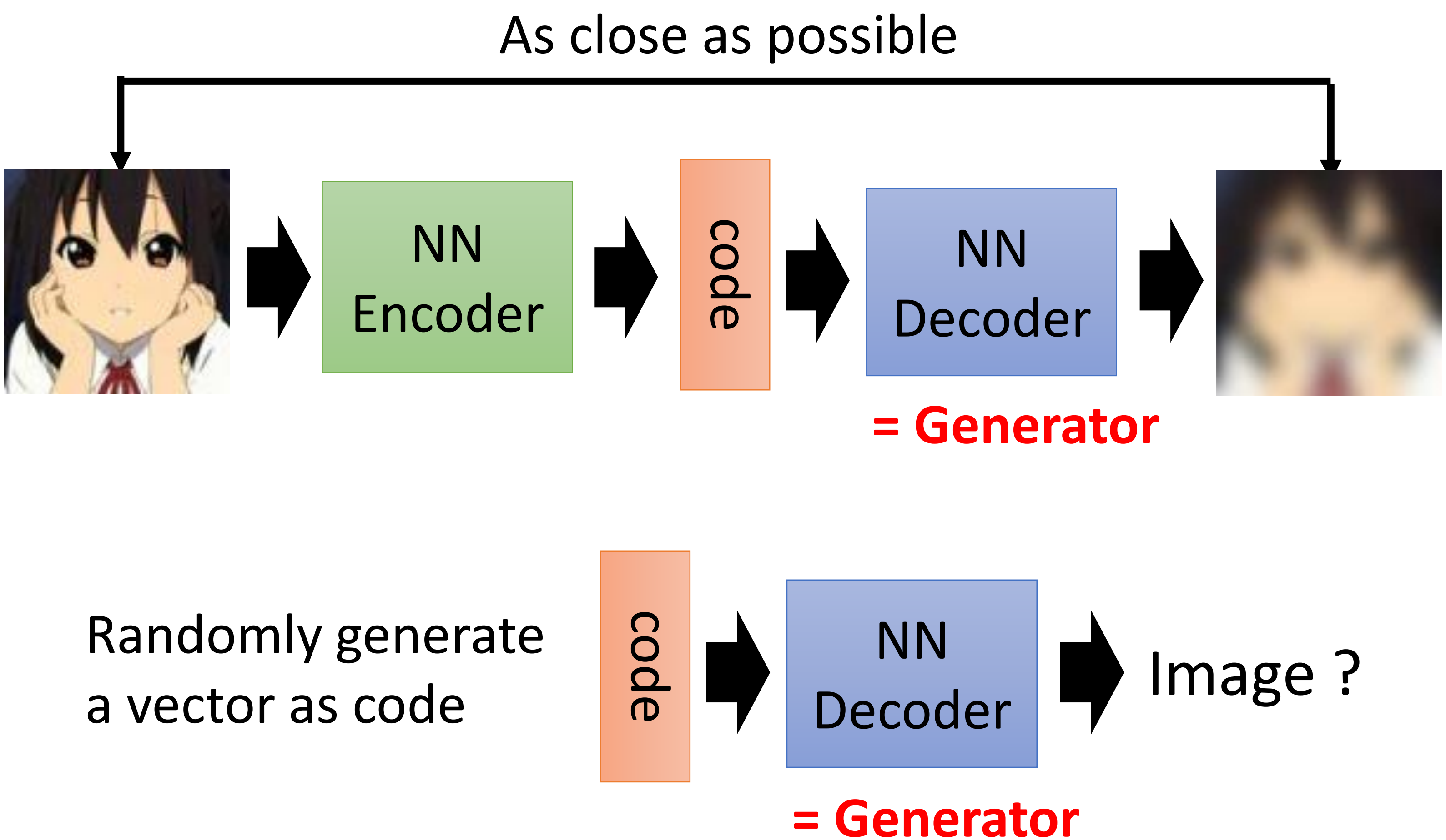
50,000 updates

The faces generated by machine.

The images are generated by Yen-Hao Chen, Po-Chun Chien, Jun-Chen Xie, Tsung-Han Wu.

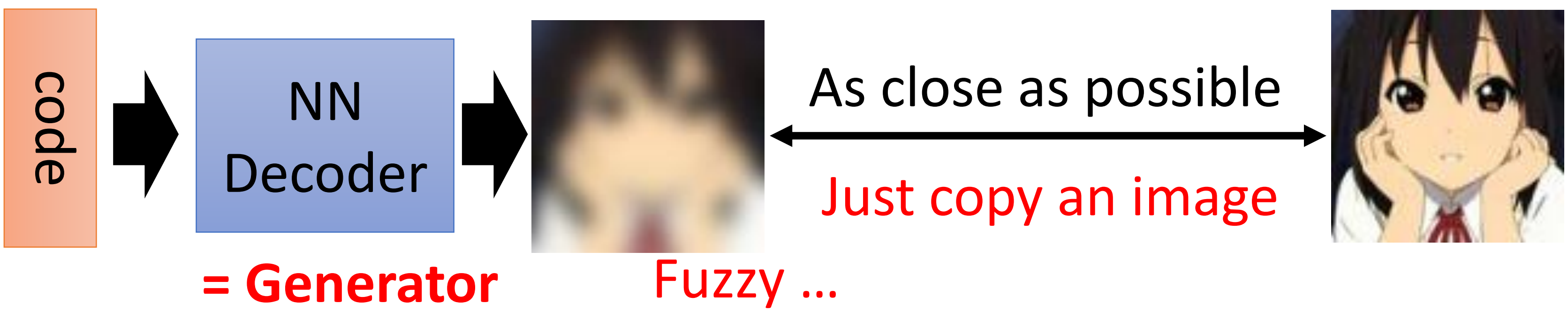

G
[0.0 0.0]
[0.1 0.1]
[0.2 0.2]
[0.3 0.3]
[0.4 0.4]
[0.5 0.5]
[0.6 0.6]
[0.7 0.7]
[0.8 0.8]
[0.9 0.9]

(Variational) Auto-encoder

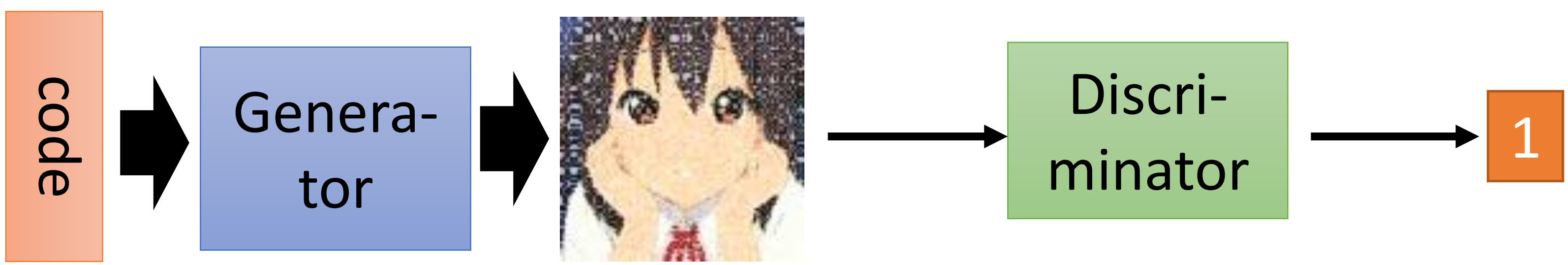


Auto-encoder v.s. GAN

Auto-encoder

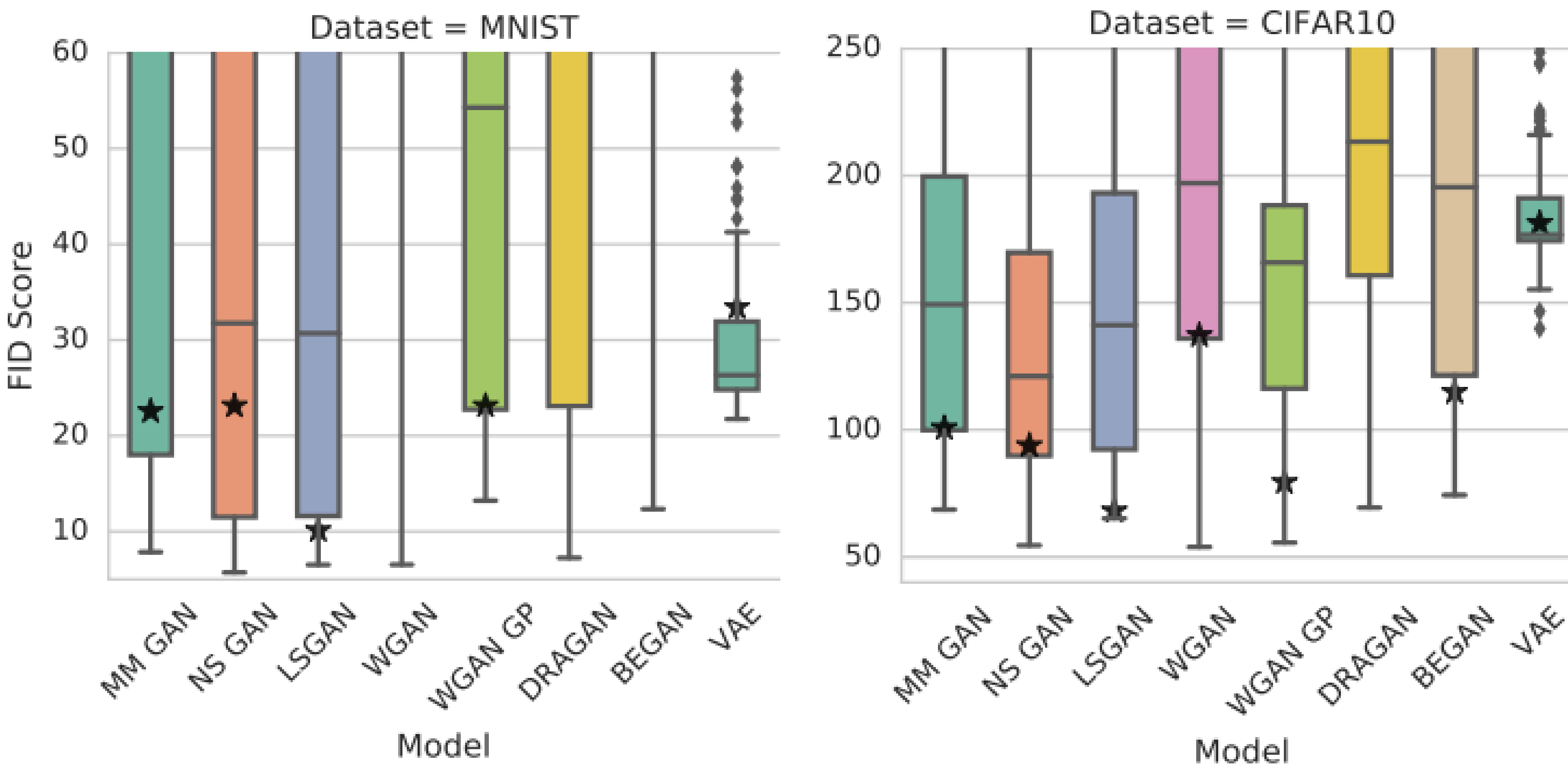


GAN



If discriminator does not simply memorize the images,
Generator learns the patterns of faces.

[Mario Lucic, et al. arXiv, 2017]



FID[Martin Heusel, et al., NIPS, 2017]: Smaller is better

Outline of Part 1

Generation

- Image Generation as Example
- Theory behind GAN
- Issues and Possible Solutions

Conditional Generation

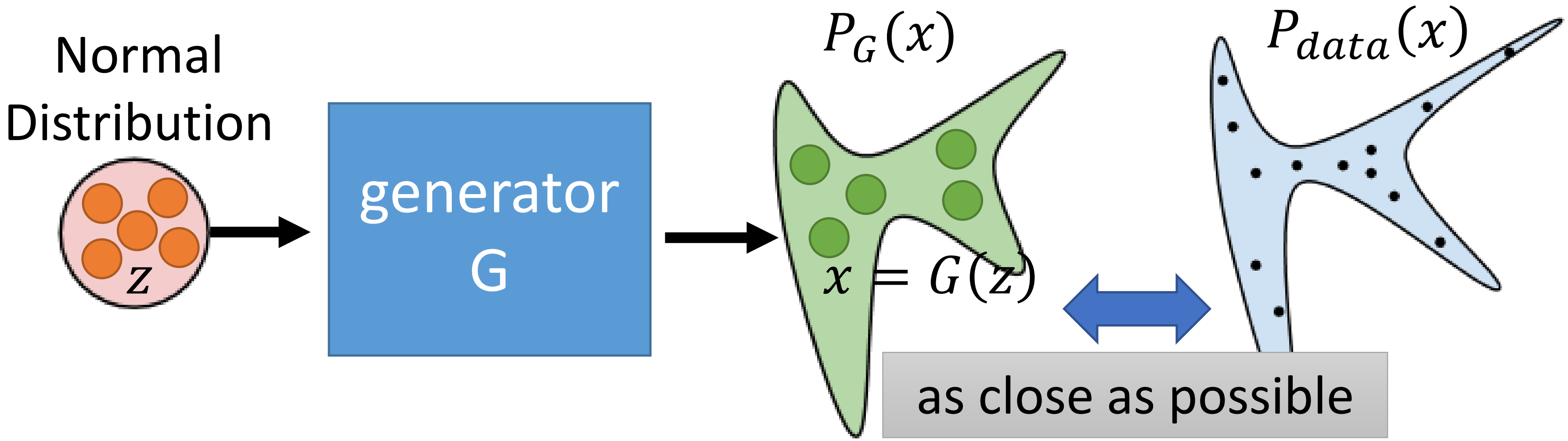
Unsupervised Conditional Generation

Relation to Reinforcement Learning

Generator

x: an image (a high-dimensional vector)

- A generator G is a network. The network defines a probability distribution P_G



$$G^* = arg \min_G \underline{Div(P_G, P_{data})}$$

Divergence between distributions P_G and P_{data}

How to compute the divergence?

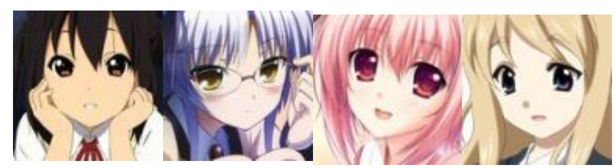
Discriminator

$$G^* = arg \min_G Div(P_G, P_{data})$$

Although we do not know the distributions of P_G and P_{data}, we can sample from them.

Database

sample →



Sampling from P_{data}

sample from normal

vector vector vector vector → G →

Sampling from P_G

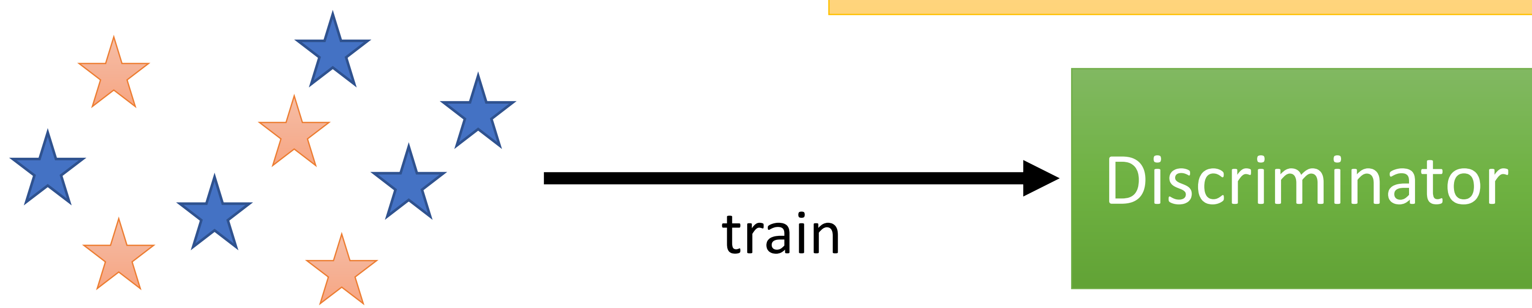
Discriminator

$$G^* = arg\min_G Div(P_G, P_{data})$$

★ : data sampled from P_{data}

☆ : data sampled from P_G

Using the example objective function is exactly the same as training a binary classifier.



Example Objective Function for D

$$V(G,D) = E_{x \sim P_{data}}[logD(x)] + E_{x \sim P_G}[log(1 - D(x))]$$

(G is fixed)

Training: $D^* = arg\boxed{\max_D V(D,G)}$

The maximum objective value is related to JS divergence.

[Goodfellow, et al., NIPS, 2014]

Discriminator

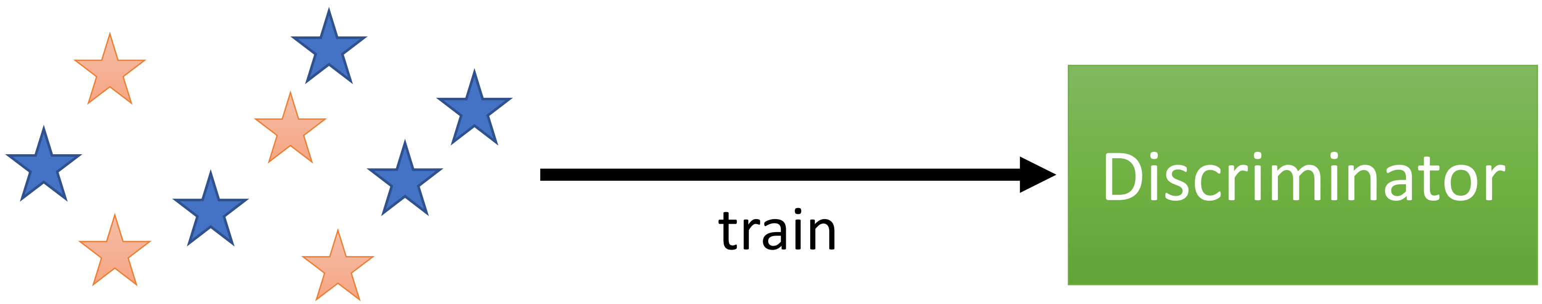
$$G^* = arg \min_G Div(P_G, P_{data})$$

★ : data sampled from P_{data}

☆ : data sampled from P_G

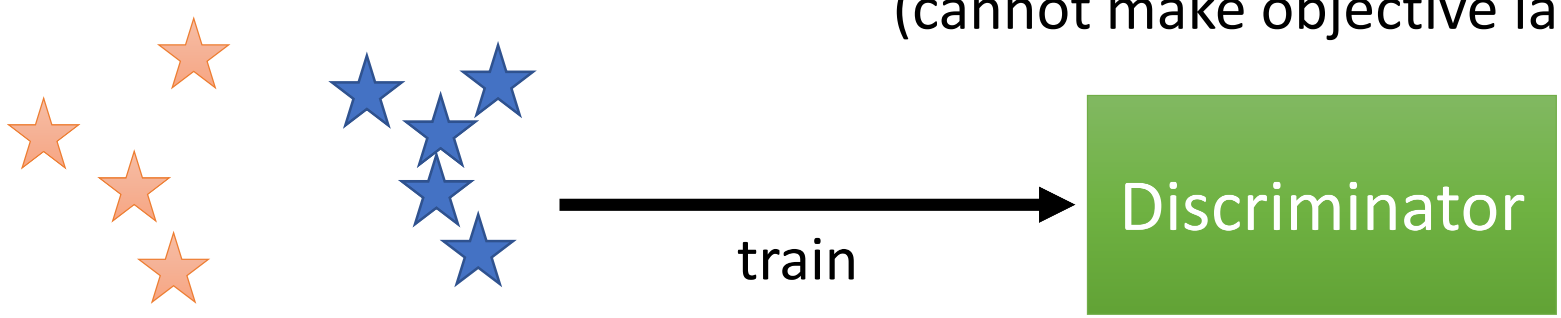
Training:

$$D^* = arg \boxed{\max_D V(D, G)}$$



small divergence

hard to discriminate
(cannot make objective large)



large divergence

easy to discriminate

[Goodfellow, et al., NIPS, 2014]

$$G^* = arg \min_G \boxed{\max_D V(G,D)}$$

$$D^* = arg \boxed{\max_D V(D,G)}$$

The maximum objective value is related to the divergence.

- Initialize generator and discriminator
- In each training iteration:

  Step 1: Fix generator G, and update discriminator D

  Step 2: Fix discriminator D, and update generator G

Can we use other divergence?

Name	$D_f(P\|Q)$	Generator $f(u)$
Total variation	$\frac{1}{2}\int \lvert p(x)-q(x)\rvert \,\mathrm{d}x$	$\frac{1}{2}\lvert u-1\rvert$
Kullback-Leibler	$\int p(x)\log\frac{p(x)}{q(x)}\,\mathrm{d}x$	$u\log u$
Reverse Kullback-Leibler	$\int q(x)\log\frac{q(x)}{p(x)}\,\mathrm{d}x$	$-\log u$
Pearson χ^2	$\int\frac{(q(x)-p(x))^2}{p(x)}\,\mathrm{d}x$	$(u-1)^2$
Neyman χ^2	$\int\frac{(p(x)-q(x))^2}{q(x)}\,\mathrm{d}x$	$\frac{(1-u)^2}{u}$
Squared Hellinger	$\int\left(\sqrt{p(x)}-\sqrt{q(x)}\right)^2\mathrm{d}x$	$\left(\sqrt{u}-1\right)^2$
Jeffrey	$\int(p(x)-q(x))\log\left(\frac{p(x)}{q(x)}\right)\mathrm{d}x$	$(u-1)\log u$
Jensen-Shannon	$\frac{1}{2}\int p(x)\log\frac{2p(x)}{p(x)+q(x)}+q(x)\log\frac{2q(x)}{p(x)+q(x)}\,\mathrm{d}x$	$-(u+1)\log\frac{1+u}{2}+u\log u$
Jensen-Shannon-weighted	$\int p(x)\pi\log\frac{p(x)}{\pi p(x)+(1-\pi)q(x)}+(1-\pi)q(x)\log\frac{q(x)}{\pi p(x)+(1-\pi)q(x)}\,\mathrm{d}x$	$\pi u\log u-(1-\pi+\pi u)\log(1-\pi+\pi u)$
GAN	$\int p(x)\log\frac{2p(x)}{p(x)+q(x)}+q(x)\log\frac{2q(x)}{p(x)+q(x)}\,\mathrm{d}x-\log(4)$	$u\log u-(u+1)\log(u+1)$

Using the divergence you like ☺

[Sebastian Nowozin, et al., NIPS, 2016]

Name	Conjugate $f^*(t)$
Total variation	t
Kullback-Leibler (KL)	$\exp(t-1)$
Reverse KL	$-1-\log(-t)$
Pearson χ^2	$\frac{1}{4}t^2+t$
Neyman χ^2	$2-2\sqrt{1-t}$
Squared Hellinger	$\frac{t}{1-t}$
Jeffrey	$W(e^{1-t})+\frac{1}{W(e^{1-t})}+t-2$
Jensen-Shannon	$-\log(2-\exp(t))$
Jensen-Shannon-weighted	$(1-\pi)\log\frac{1-\pi}{1-\pi e^{t/\pi}}$
GAN	$-\log(1-\exp(t))$

Outline of Part 1

https://github.com/soumith/ganhacks

Generation

- Image Generation as Example
- Theory behind GAN
- Issues and Possible Solutions

Conditional Generation

Unsupervised Conditional Generation

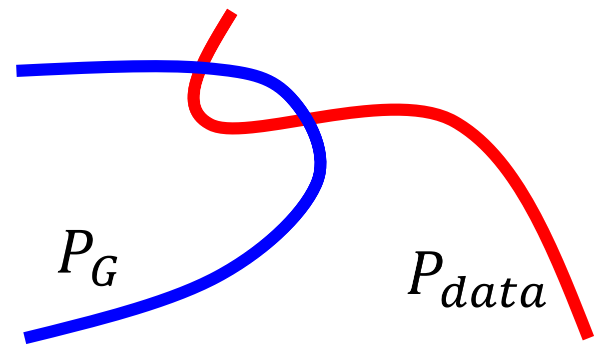
Relation to Reinforcement Learning

JS divergence is not suitable

- In most cases, P_G and P_{data} are not overlapped.
- 1. The nature of data

Both P_{data} and P_G are low-dim manifold in high-dim space.

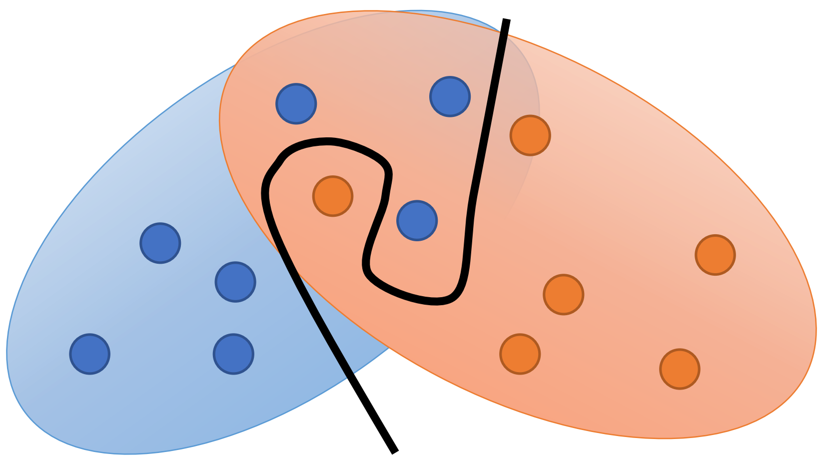
The overlap can be ignored.



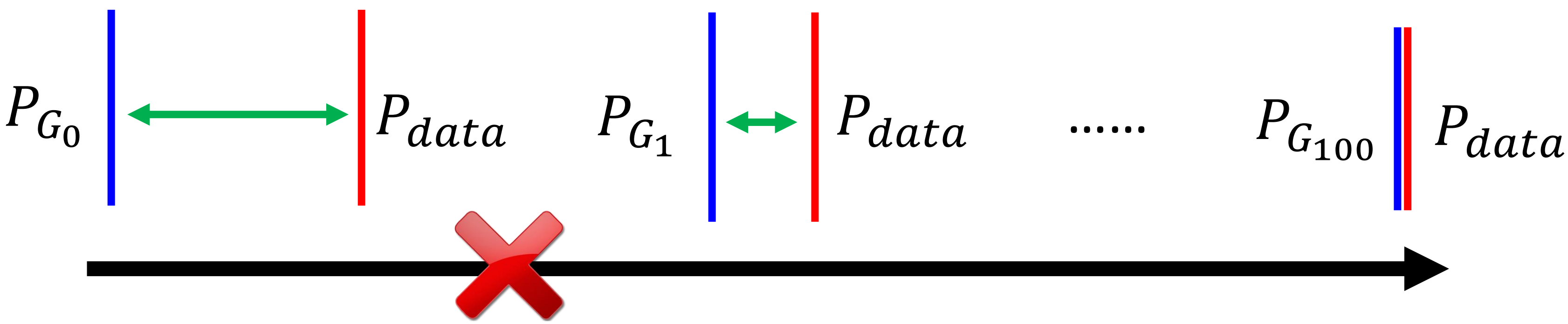
- 2. Sampling

Even though P_{data} and P_G have overlap.

If you do not have enough sampling ……



What is the problem of JS divergence?



Equally bad

$JS(P_{G_0}, P_{data}) = log2$ $\quad$ $JS(P_{G_1}, P_{data}) = log2$ $\quad$ …… $\quad$ $JS(P_{G_{100}}, P_{data}) = 0$

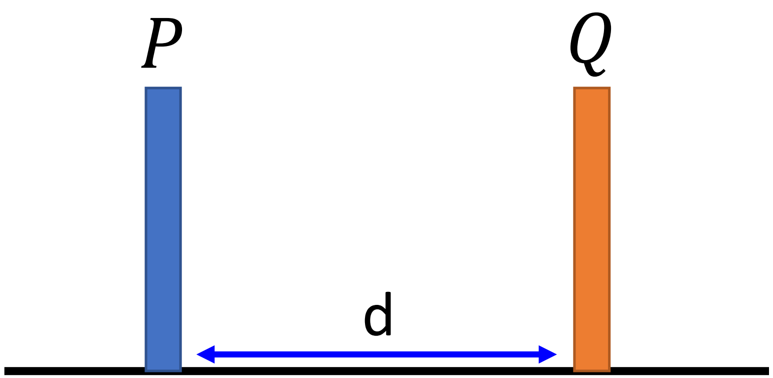
JS divergence is log2 if two distributions do not overlap.

Intuition: If two distributions do not overlap, binary classifier achieves 100% accuracy

➡ Same objective value is obtained. ➡ Same divergence

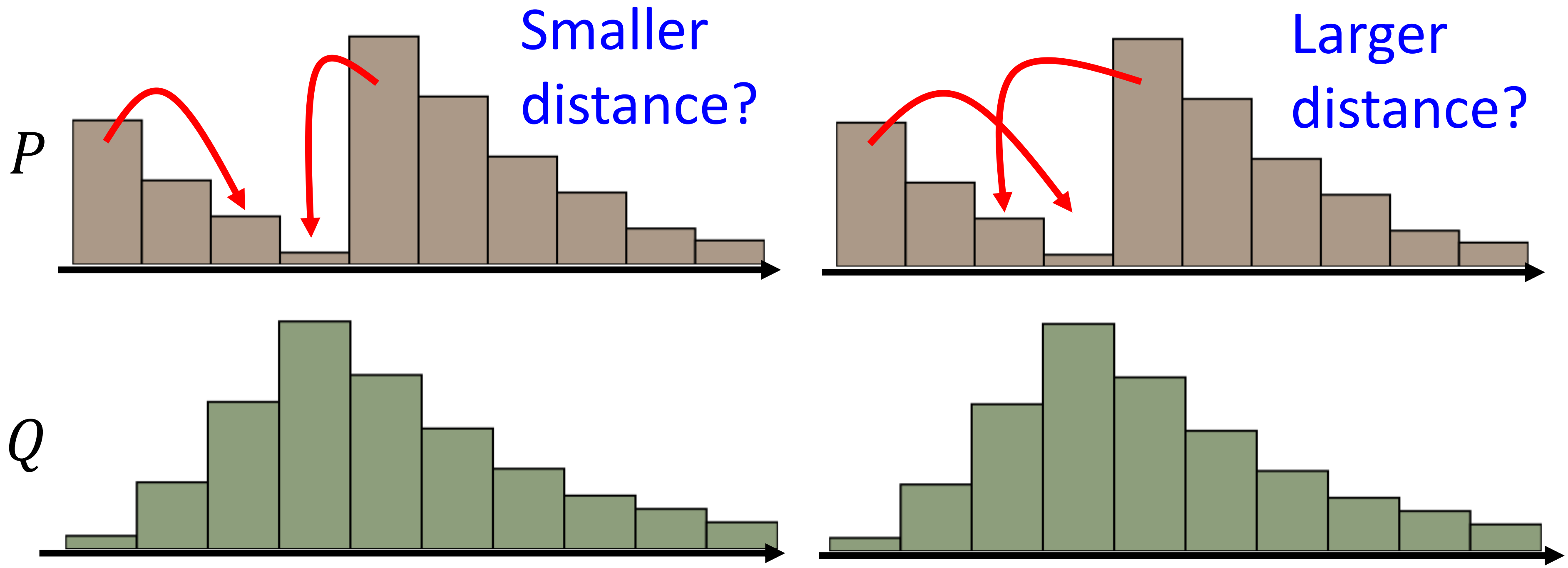
Wasserstein distance

- Considering one distribution P as a pile of earth, and another distribution Q as the target
- The average distance the earth mover has to move the earth.



$$W(P,Q) = d$$

Wasserstein distance

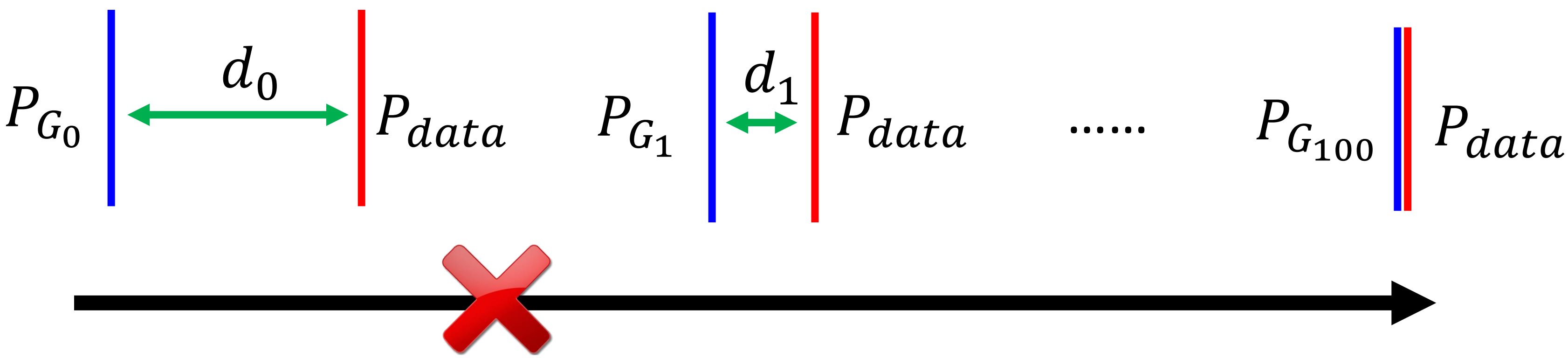


There are many possible "moving plans".

Using the "moving plan" with the smallest average distance to define the Wasserstein distance.

Source of image: https://vincentherrmann.github.io/blog/wasserstein/

What is the problem of JS divergence?



$JS(P_{G_0}, P_{data})$ $= log2$

$JS(P_{G_1}, P_{data})$ $= log2$

......

$JS(P_{G_{100}}, P_{data})$ $= 0$

$W(P_{G_0}, P_{data})$ $= d_0$

$W(P_{G_1}, P_{data})$ $= d_1$

......

$W(P_{G_{100}}, P_{data})$ $= 0$

Better!

[Martin Arjovsky, et al., arXiv, 2017]

WGAN

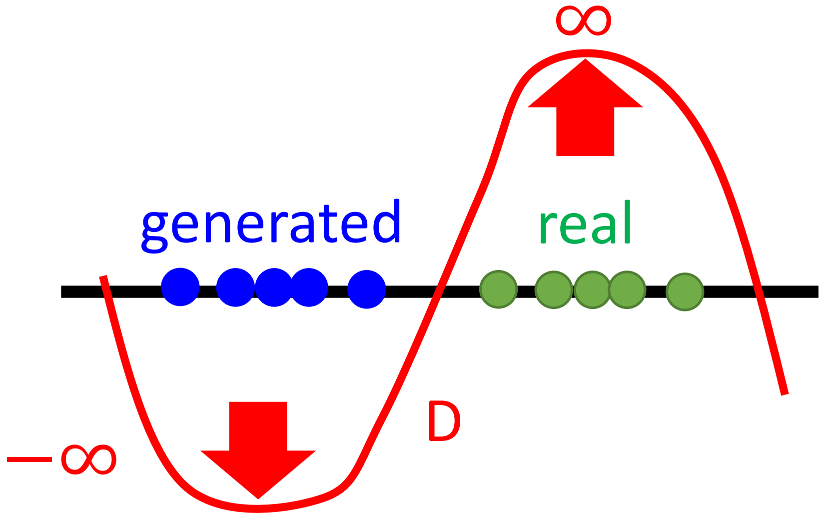
Evaluate wasserstein distance between P_{data} and P_G

$$V(G,D) = \max_{D \in 1-Lipschitz} \left\{ E_{x \sim P_{data}}[D(x)] - E_{x \sim P_G}[D(x)] \right\}$$

D has to be smooth enough. How to fulfill this constraint?

Without the constraint, the training of D will not converge.

Keeping the D smooth forces D(x) become ∞ and $-\infty$



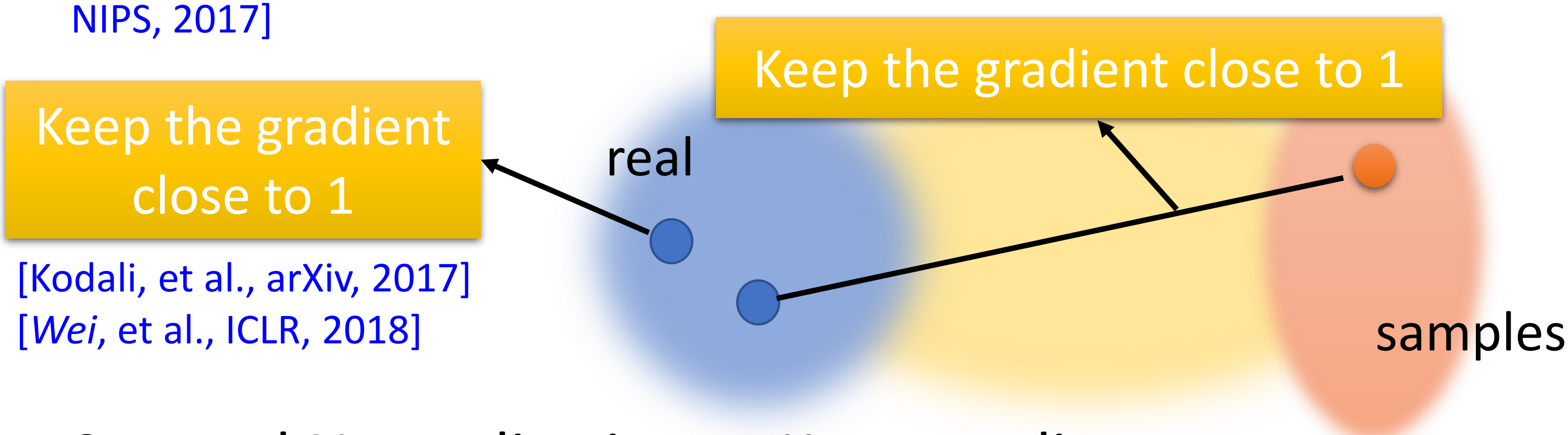
$$V(G,D) = \max_{D \in 1-Lipschitz} \{E_{x\sim P_{data}}[D(x)] - E_{x\sim P_G}[D(x)]\}$$

- Original WGAN → Weight Clipping [Martin Arjovsky, et al., arXiv, 2017]

  Force the parameters w between c and -c

  After parameter update, if w > c, w = c; if w < -c, w = -c

- Improved WGAN → Gradient Penalty [Ishaan Gulrajani, NIPS, 2017]



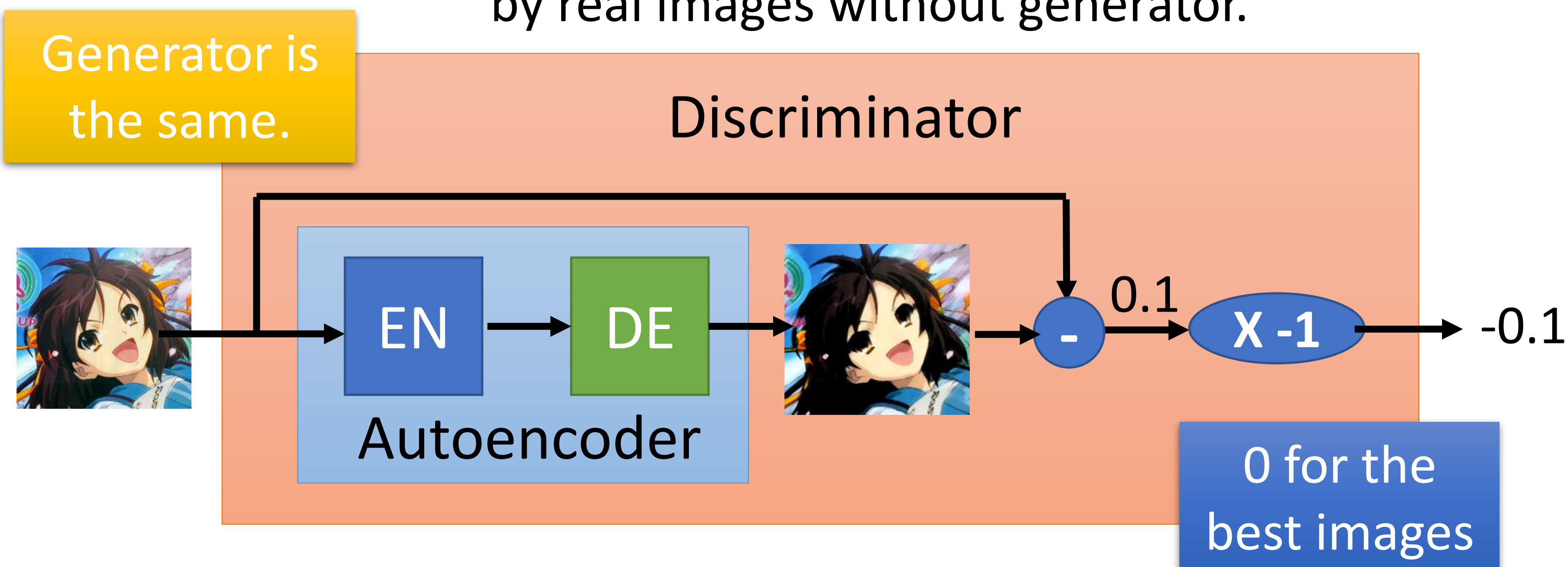
[Kodali, et al., arXiv, 2017]
[*Wei*, et al., ICLR, 2018]

- Spectral Normalization → Keep gradient norm smaller than 1 everywhere [Miyato, et al., ICLR, 2018]

[Junbo Zhao, et al., arXiv, 2016]

Energy-based GAN (EBGAN)

- Using an autoencoder as discriminator D
  - ➢ Using the negative reconstruction error of auto-encoder to determine the goodness
  - ➢ **Benefit**: The auto-encoder can be pre-train by real images without generator.

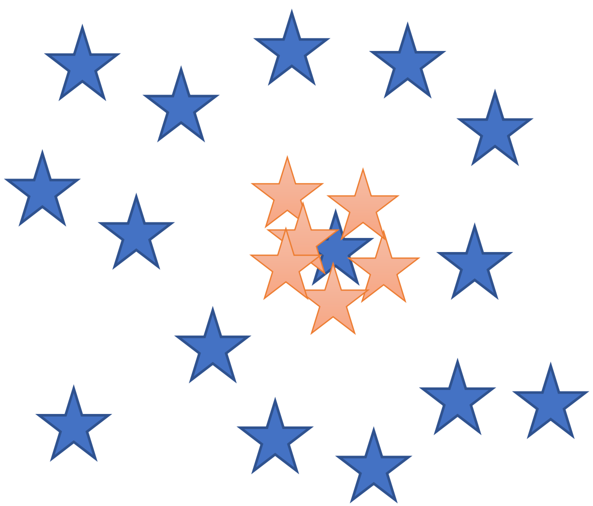


Mode Collapse

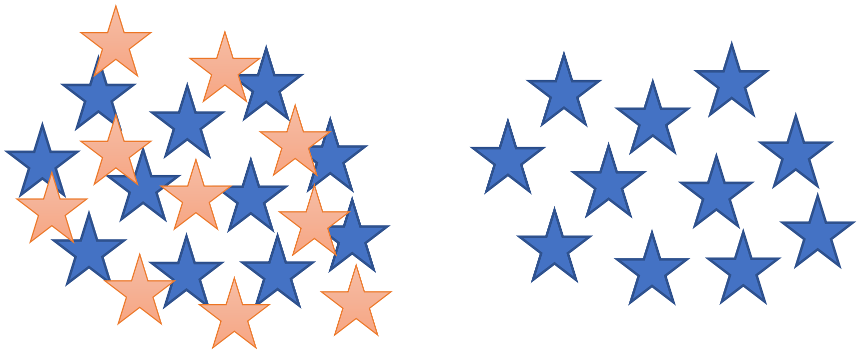
Training with too many iterations ……

★ : real data

★ : generated data



Missing Mode?



Generator switches mode during training

Generator
at iteration t

Generator
at iteration t+1

Generator
at iteration t+2

BEGAN on CelebA

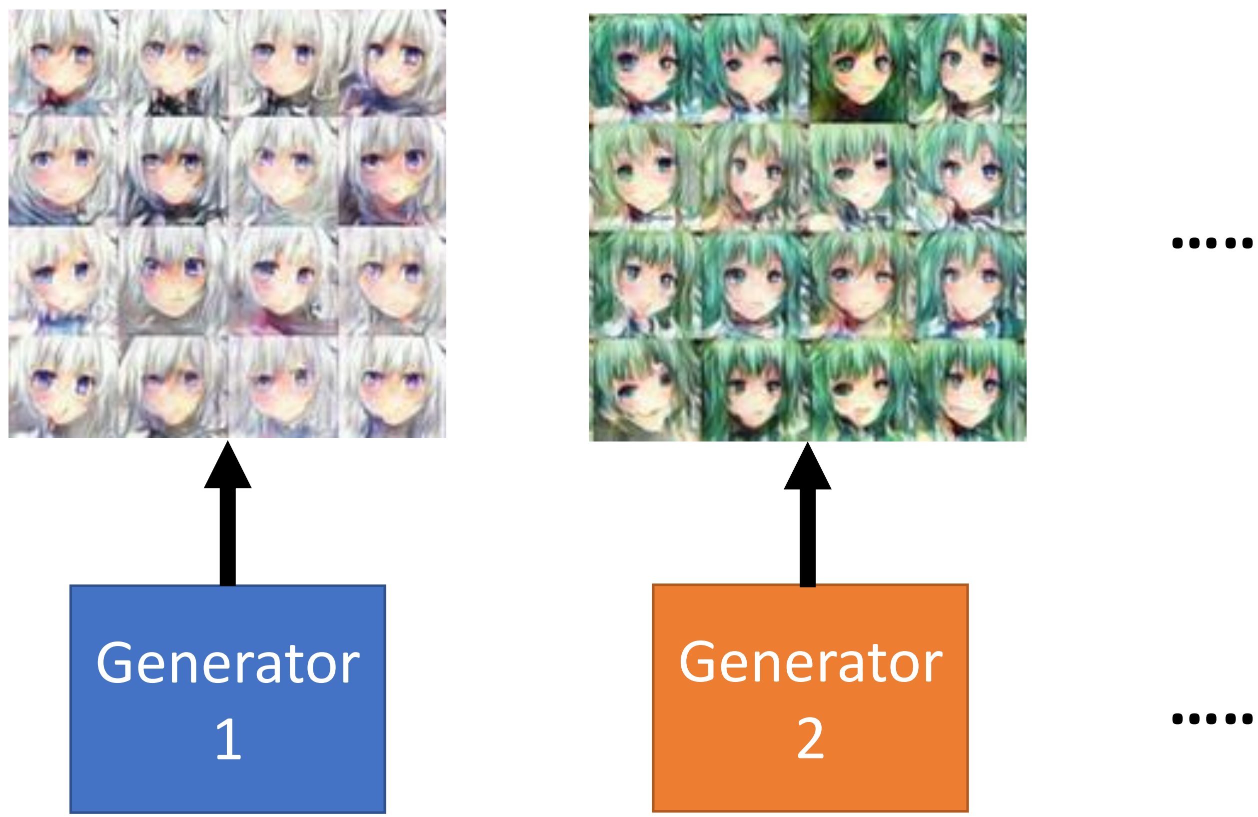
Ensemble

Train a set of generators: $\{G_1, G_2, \cdots, G_N\}$

To generate an image

Random pick a generator G_i

Use G_i to generate the image

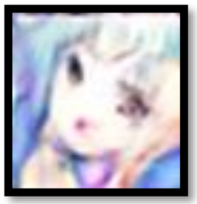


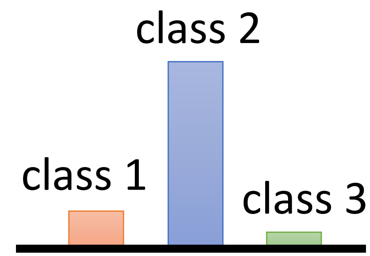
[Tim Salimans, et al., NIPS, 2016]

Objective Evaluation

x: image

y: class (output of CNN)

x  → **Off-the-shelf Image Classifier** → $P(y|x)$



e.g. Inception net, VGG, etc.

Concentrated distribution means higher visual quality

---

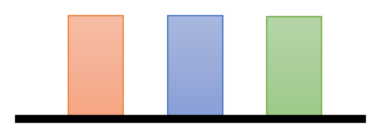
x^1 → CNN → $P(y^1|x^1)$

x^2 → CNN → $P(y^2|x^2)$

x^3 → CNN → $P(y^3|x^3)$

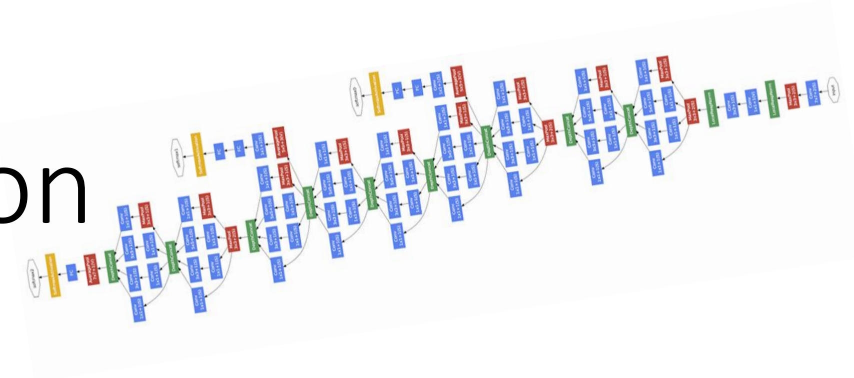
⋮

$$P(y) = \frac{1}{N}\sum_n P(y^n|x^n)$$

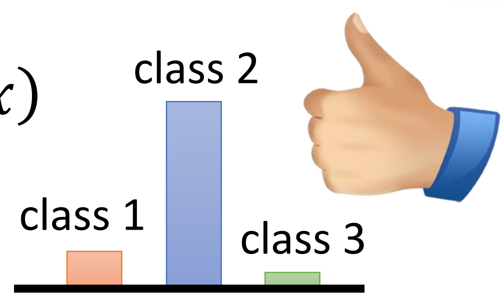


Uniform distribution means higher variety

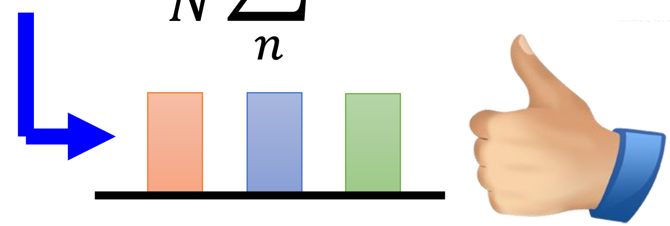
Objective Evaluation



$P(y|x)$



$$P(y) = \frac{1}{N}\sum_n P(y^n|x^n)$$



Inception Score

$$= \sum_x \sum_y P(y|x)logP(y|x)$$

Negative entropy of P(y|x)

$$- \sum_y P(y)logP(y)$$

Entropy of P(y)

Outline of Part 1

Generation

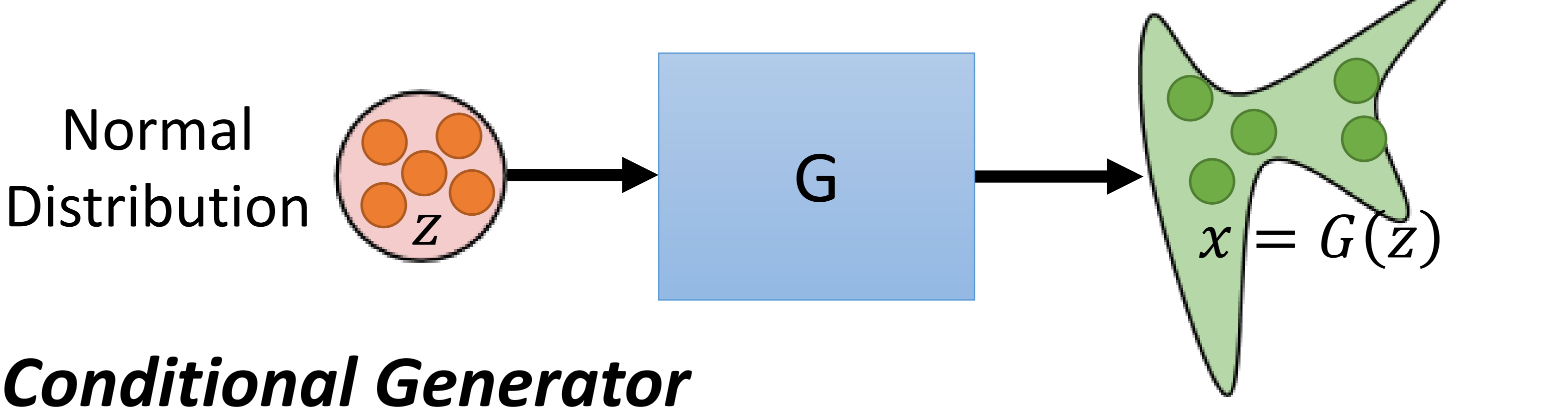
Conditional Generation

Unsupervised Conditional Generation

Relation to Reinforcement Learning

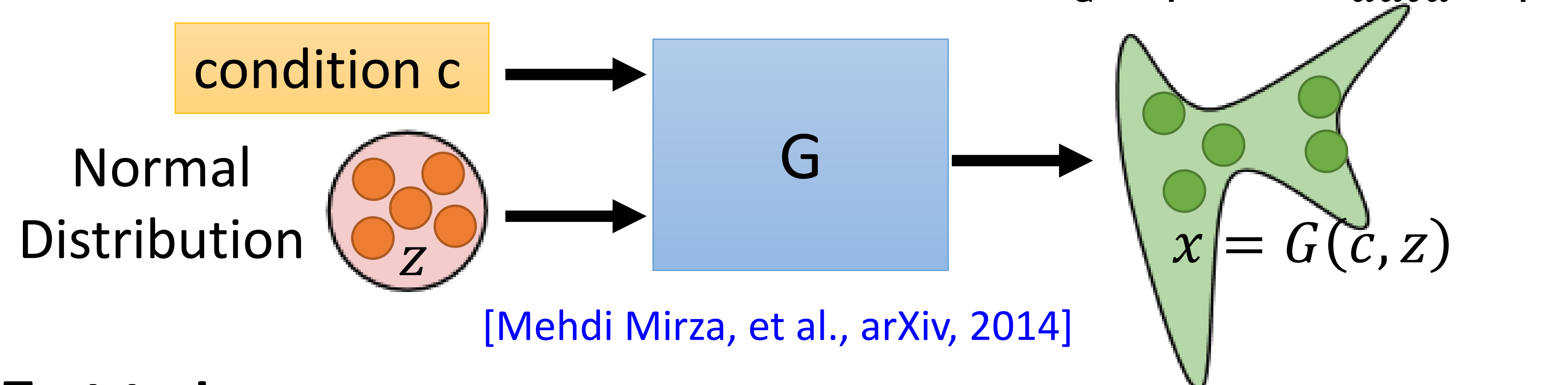
- ***Original Generator***

$$P_G(x) \rightarrow P_{data}(x)$$



- ***Conditional Generator***

$$P_G(x|c) \rightarrow P_{data}(x|c)$$



e.g. Text-to-Image

"Girl with red hair and red eyes"

"Girl with yellow ribbon"

Text-to-Image

a dog is running 

a bird is flying 

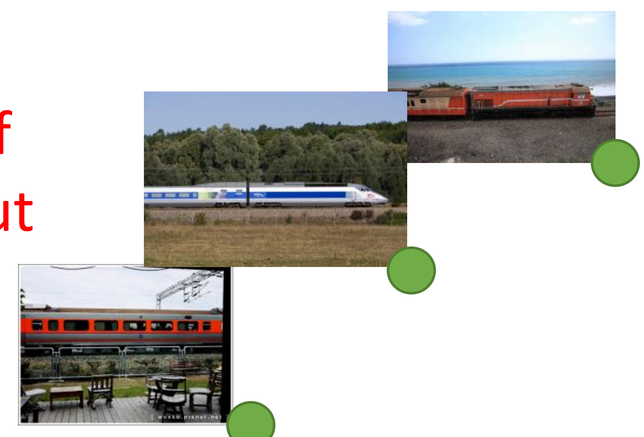
- **Traditional supervised approach**

c^1: a dog is running → NN → Image ↔ 

as close as possible

Text: "train"

Target of NN output



A blurry image!

[Scott Reed, et al, ICML, 2016]

Conditional GAN

c: train

Normal distribution z

G

Image

x = G(c,z)

x is real image or not

x

D (original)

scalar

Real images:  1

Generated images: Image 0

Generator will learn to generate realistic images ....

But completely ignore the input conditions.

[Scott Reed, et al, ICML, 2016]

Conditional GAN

c: train → G ← Normal distribution z

G → Image $x = G(c,z)$

c → D (better) ← x

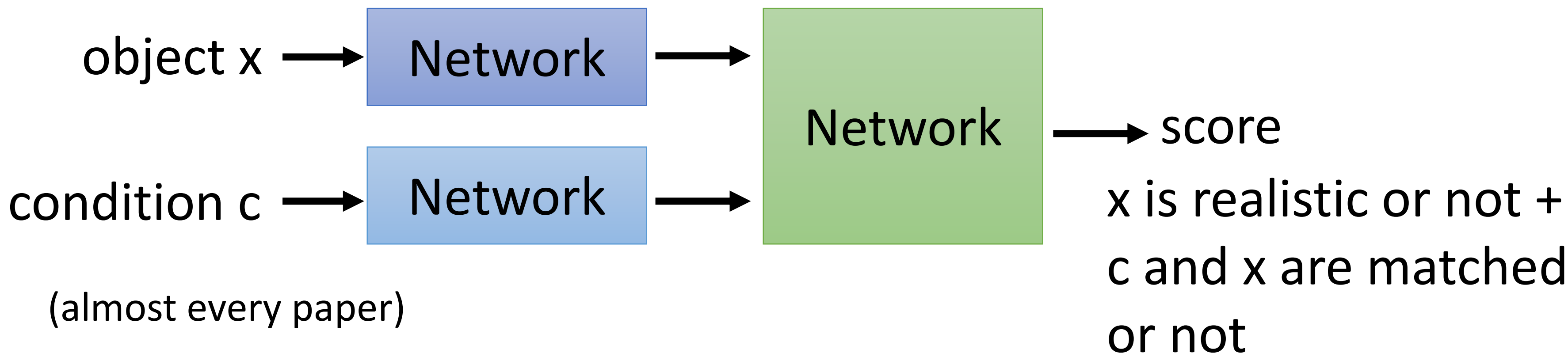
D (better) → scalar

x is realistic or not +
c and x are matched or not

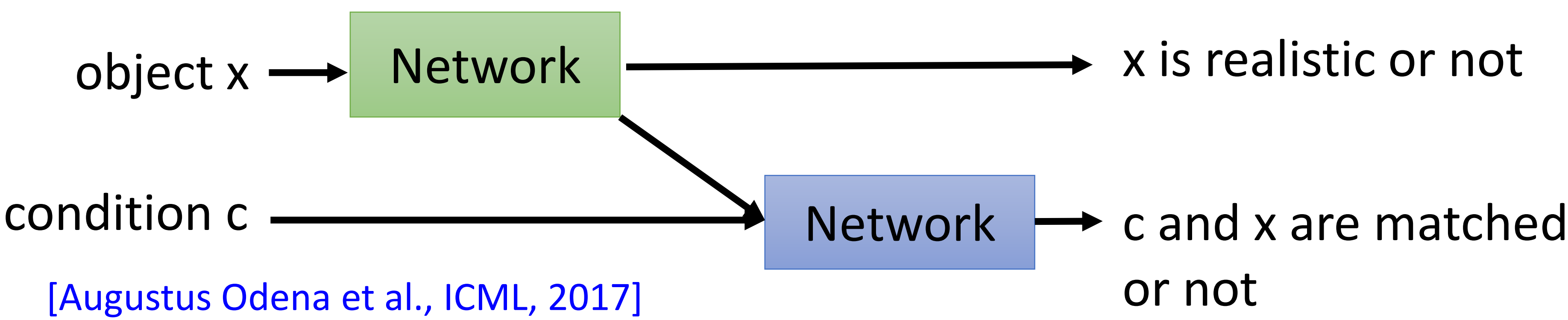
True text-image pairs: (train , ) 1

(cat , ) 0   (train , Image) 0

Conditional GAN - Discriminator



(almost every paper)



[Augustus Odena et al., ICML, 2017]

[Takeru Miyato, et al., ICLR, 2018]

[Han Zhang, et al., arXiv, 2017]

Conditional GAN

The images are generated by Yen-Hao Chen, Po-Chun Chien, Jun-Chen Xie, Tsung-Han Wu.

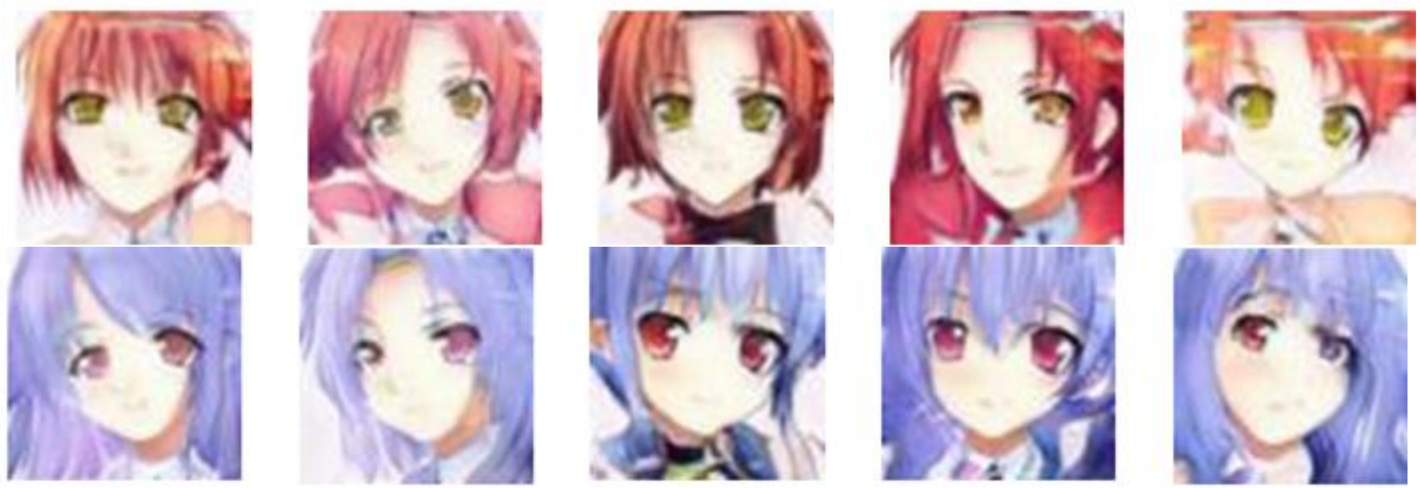
paired data

blue eyes
red hair
short hair

Collecting anime faces and the description of its characteristics

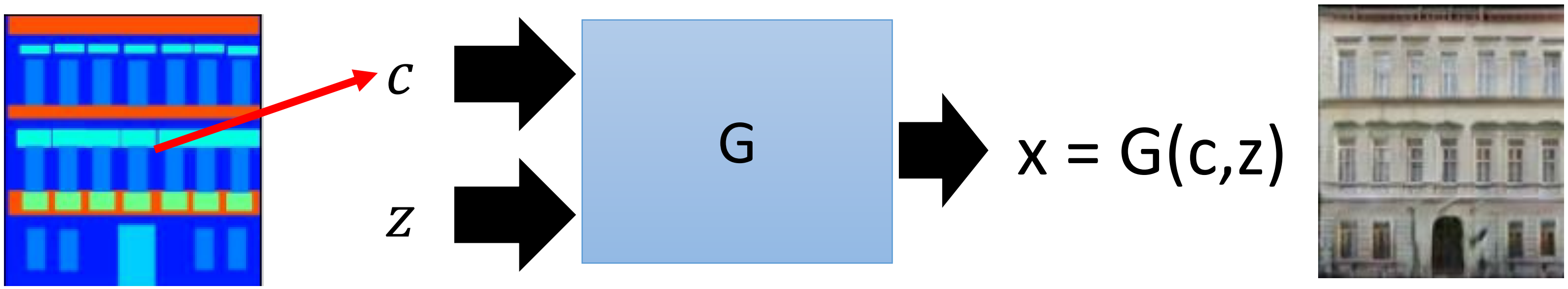
red hair, green eyes

blue hair, red eyes



[Phillip Isola, et al., CVPR, 2017]

Conditional GAN - Image-to-image



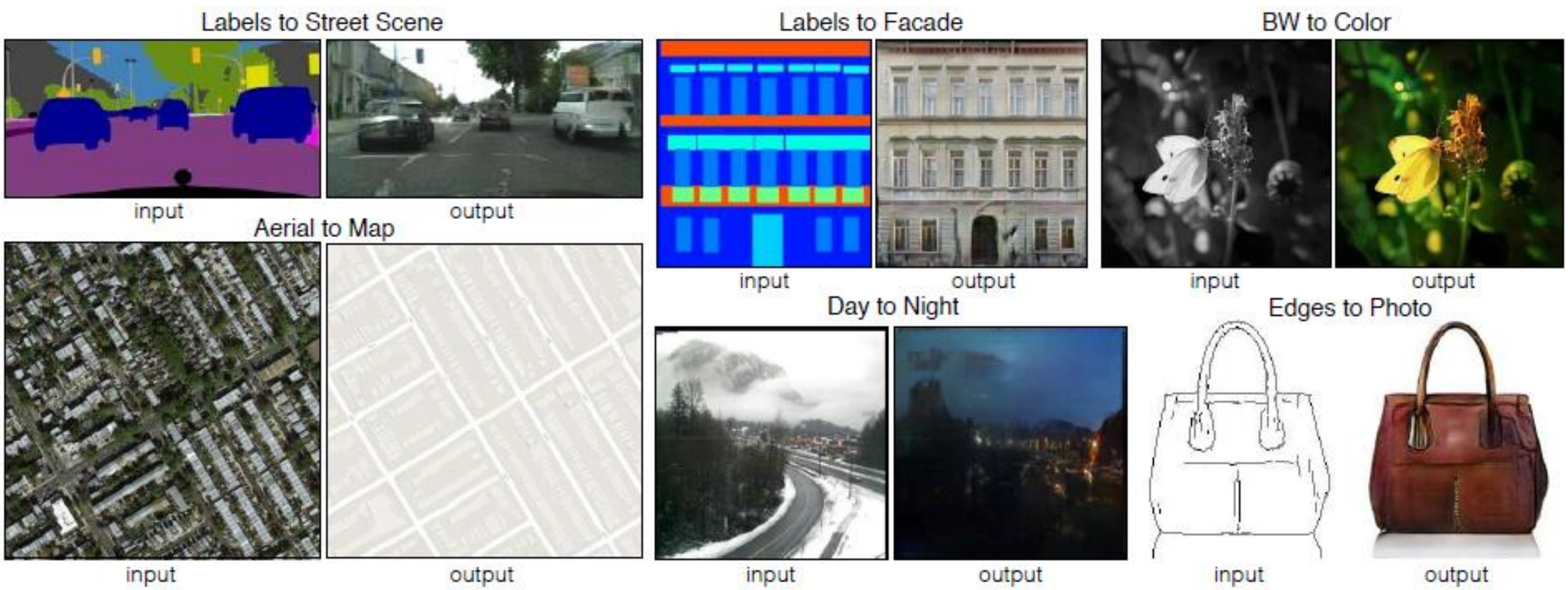
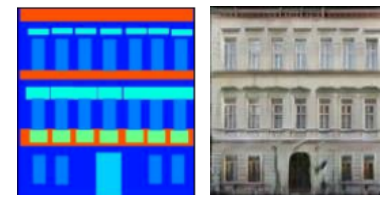
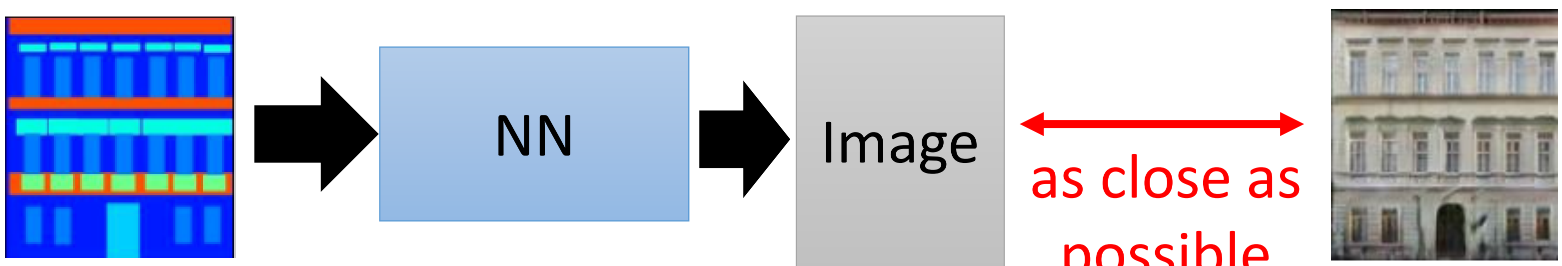


Image translation, or pix2pix

Conditional GAN - Image-to-image

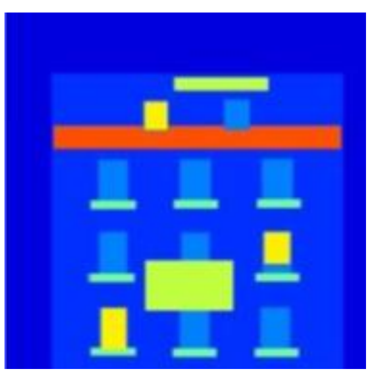
- **Traditional supervised approach**



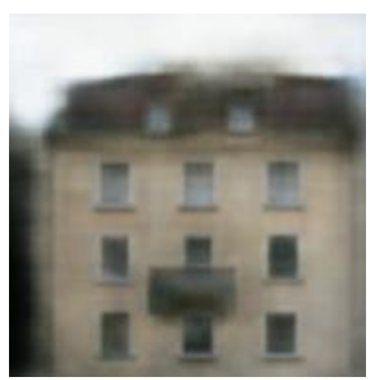


e.g. L1

Testing:



input

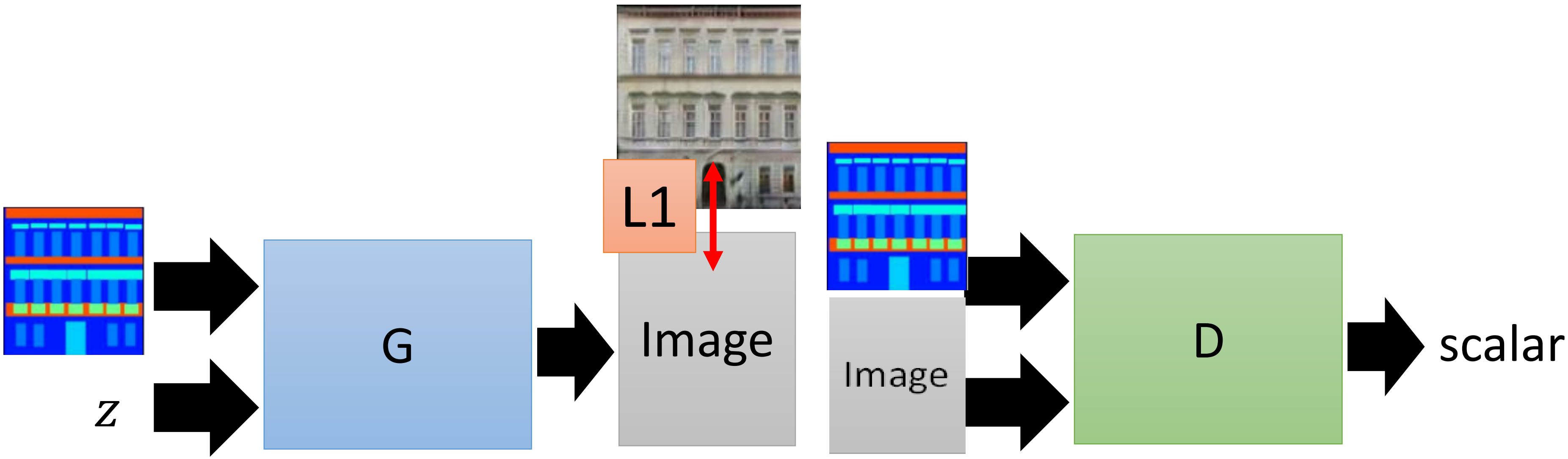


L1

It is blurry.

[Phillip Isola, et al., CVPR, 2017]

Conditional GAN - Image-to-image

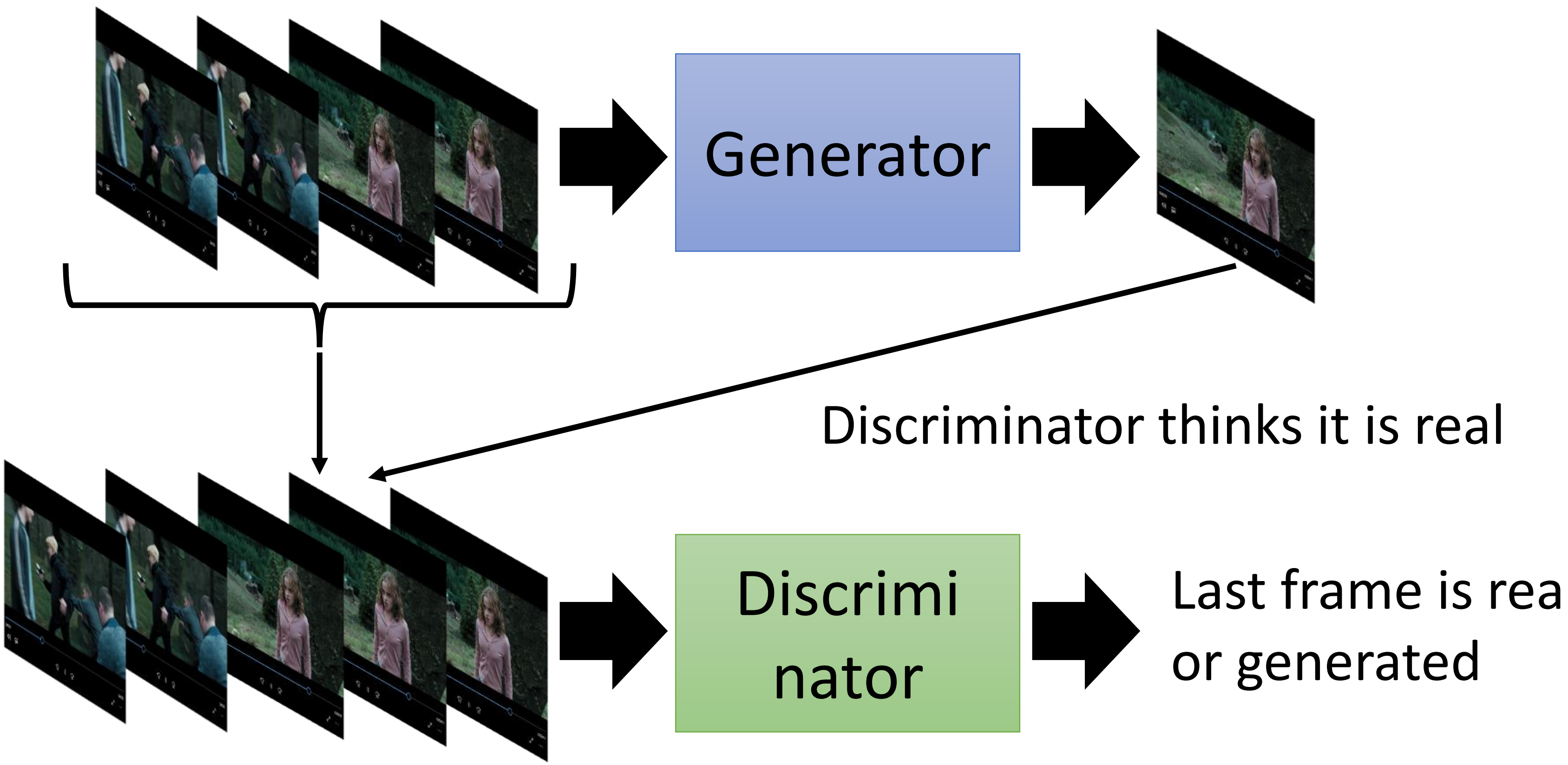


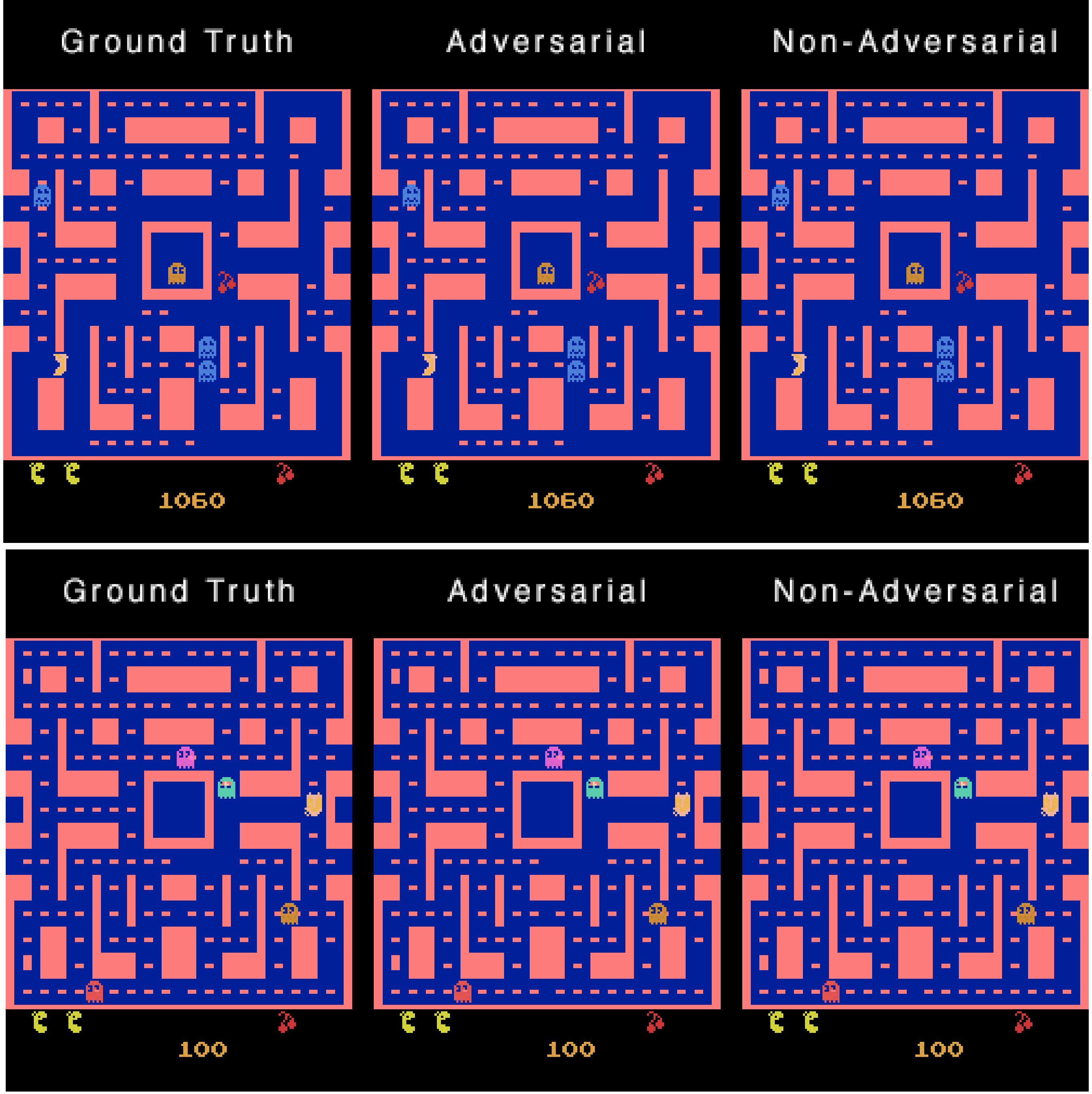
Testing:

input L1 GAN GAN + L1

Conditional GAN
- Video Generation

[Michael Mathieu, et al., arXiv, 2015]

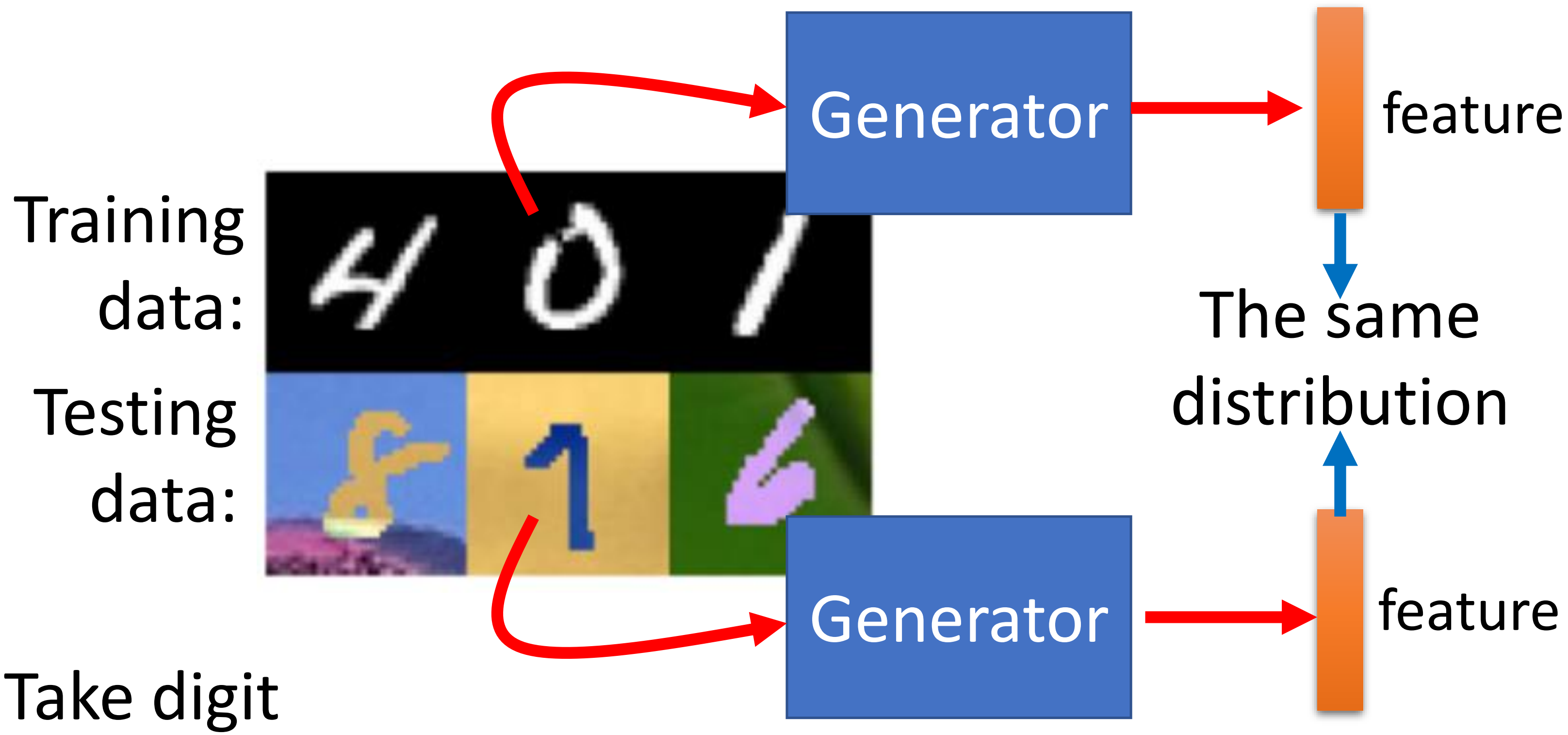




https://github.com/dyelax/Adversarial_Video_Generation

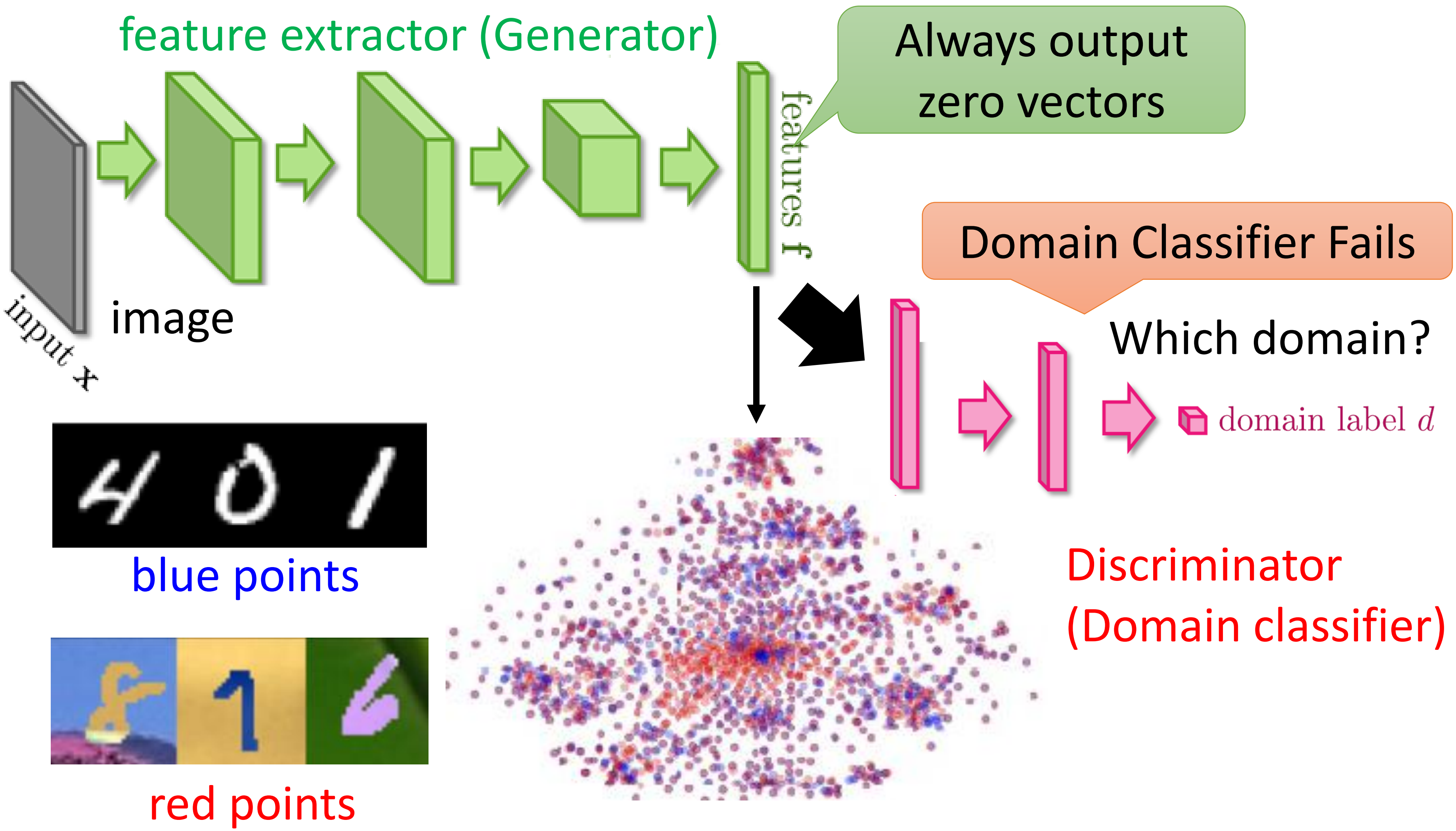
Domain Adversarial Training

- Training and testing data are in different domains

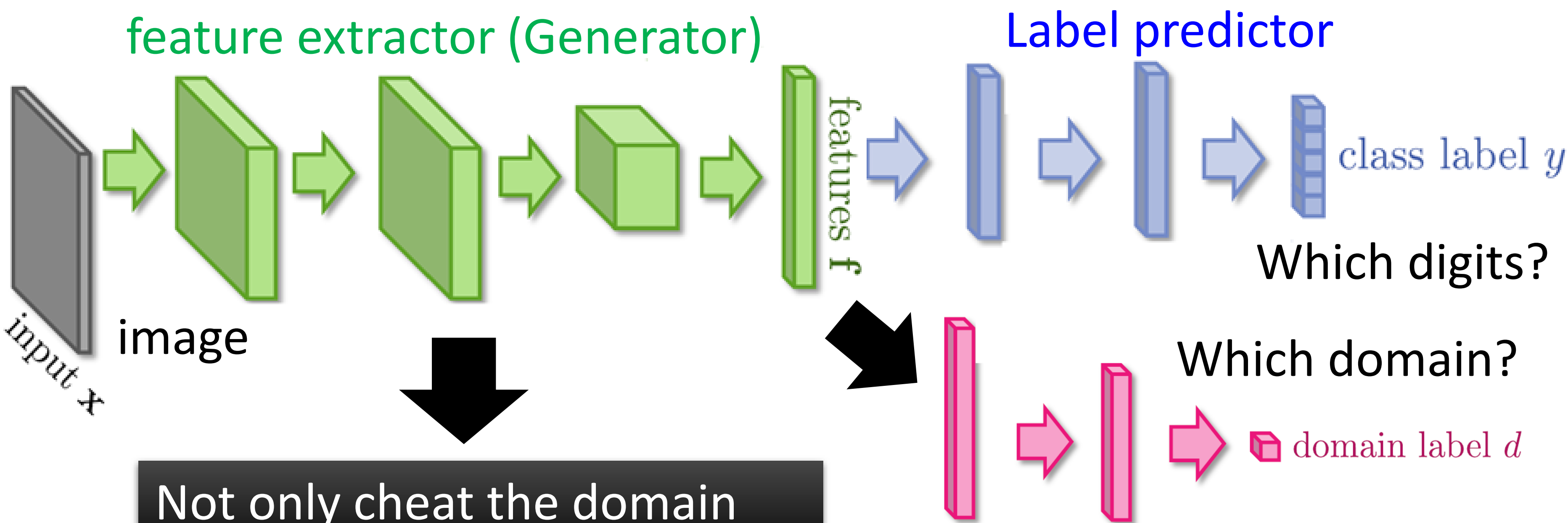


Take digit classification as example

Domain Adversarial Training



Domain Adversarial Training



Not only cheat the domain classifier, but satisfying label predictor at the same time

Successfully applied on image classification
[Ganin et al, ICML, 2015][Ajakan et al. JMLR, 2016]

More speech-related applications in Part II.

Outline of Part 1

Generation

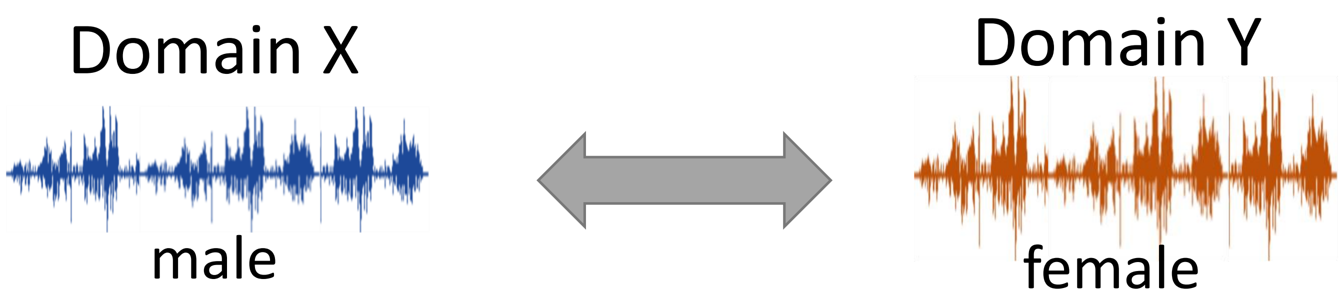
Conditional Generation

Unsupervised Conditional Generation

Relation to Reinforcement Learning

Unsupervised Conditional Generation

Transform an object from one domain to another ***without paired data*** (e.g. style transfer)



Part II

Domain X	Domain Y
It is good.	It is bad.
It's a good day.	It's a bad day.
I love you.	I don't love you.

Part III

Unsupervised Conditional Generation

- Approach 1: Direct Transformation

Domain X

$G_{X \to Y}$

Domain Y

For texture or color change

- Approach 2: Projection to Common Space

Domain X

EN_X

Encoder of domain X

Face Attribute

DE_Y

Decoder of domain Y

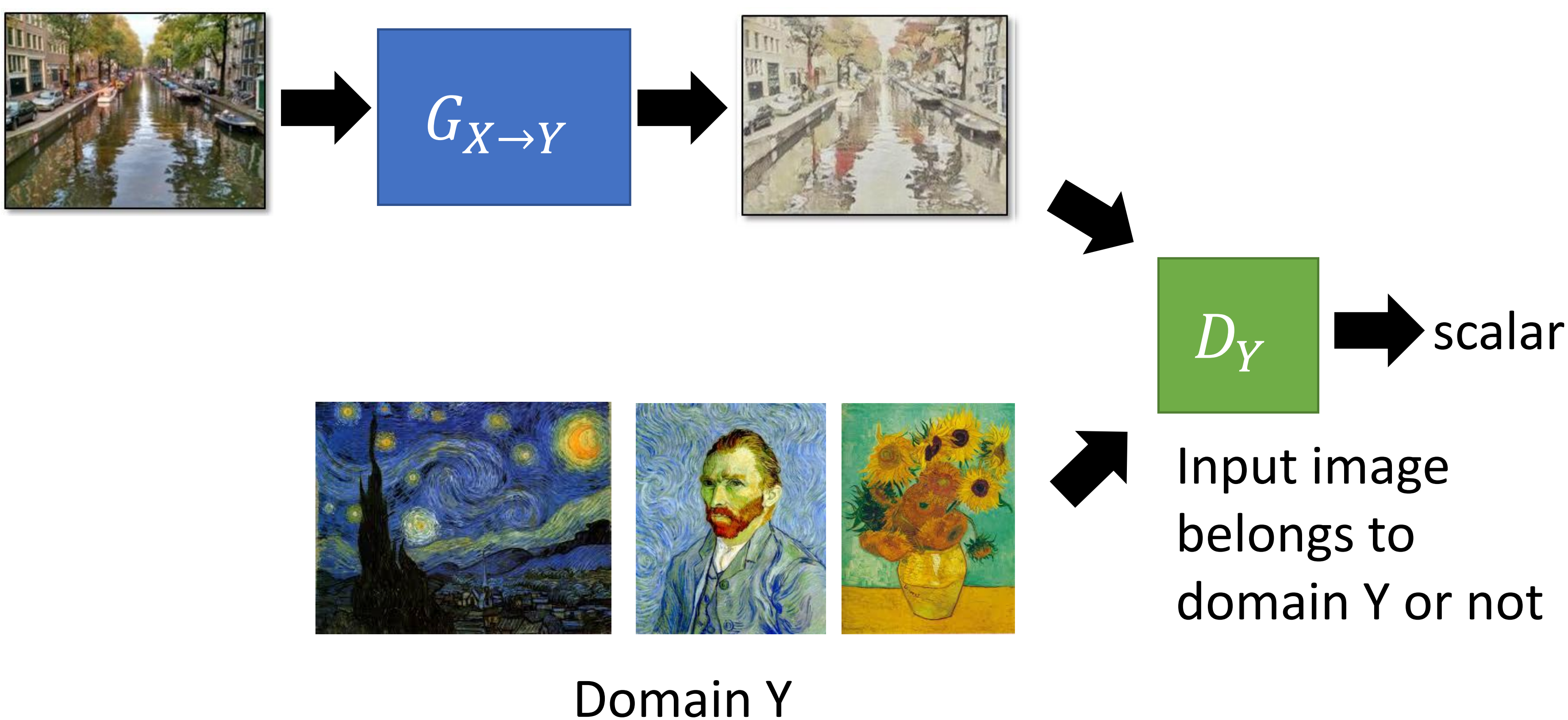
Domain Y

Larger change, only keep the semantics

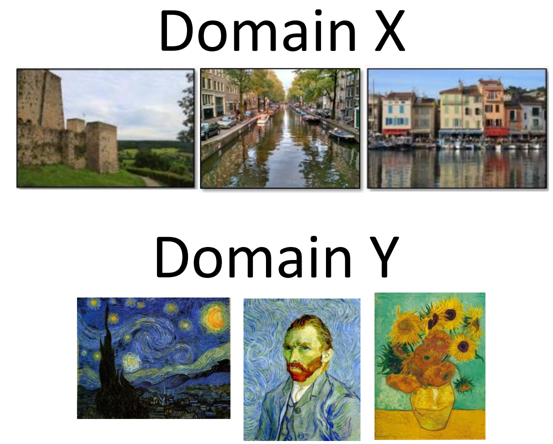
Direct Transformation

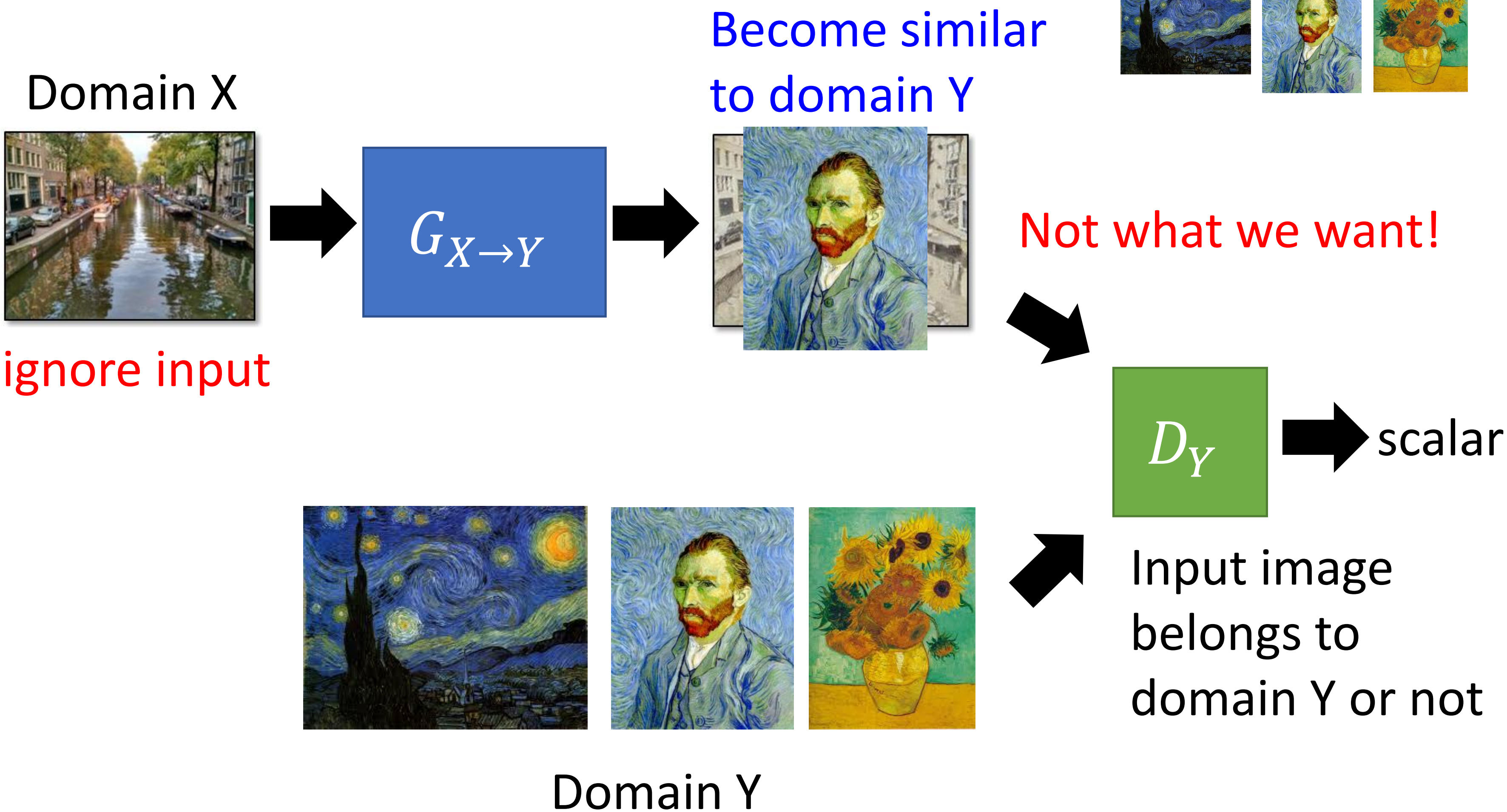
Domain X

Domain Y



Direct Transformation





Direct Transformation

Domain X

Domain Y

Become similar to domain Y

Domain X

$G_{X \to Y}$

ignore input

Not what we want!

D_Y → scalar

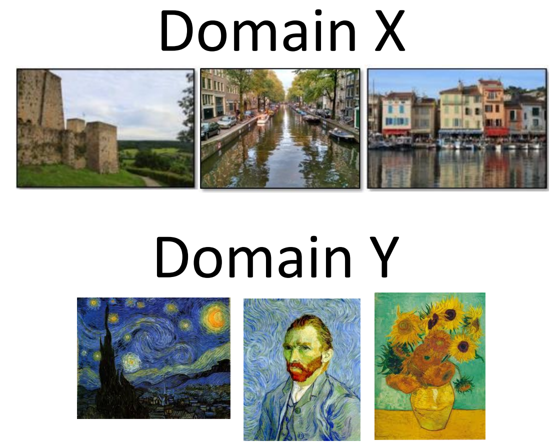
Input image belongs to domain Y or not

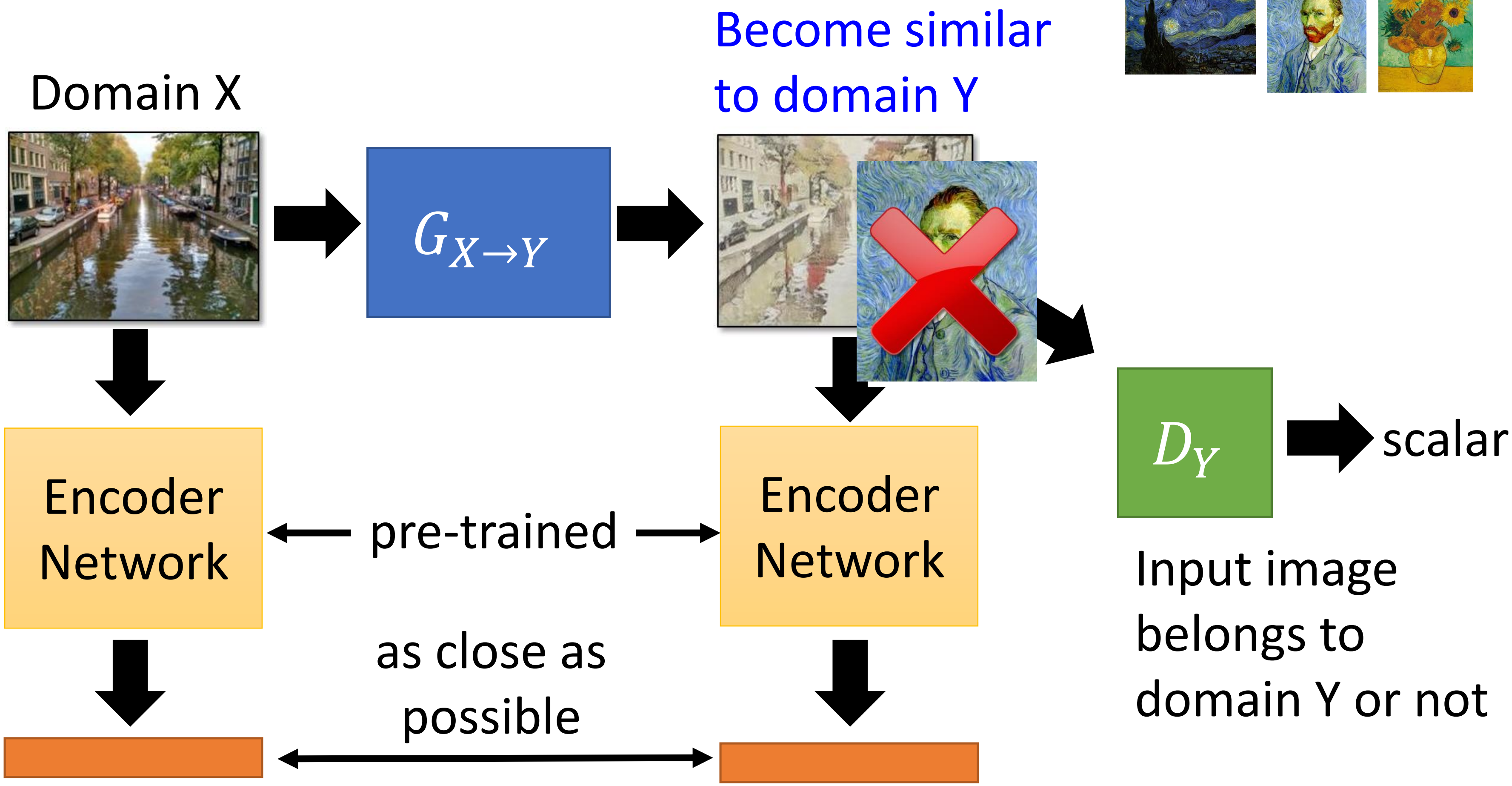
The issue can be avoided by network design.

Simpler generator makes the input and output more closely related.

[Tomer Galanti, et al. ICLR, 2018]

Direct Transformation

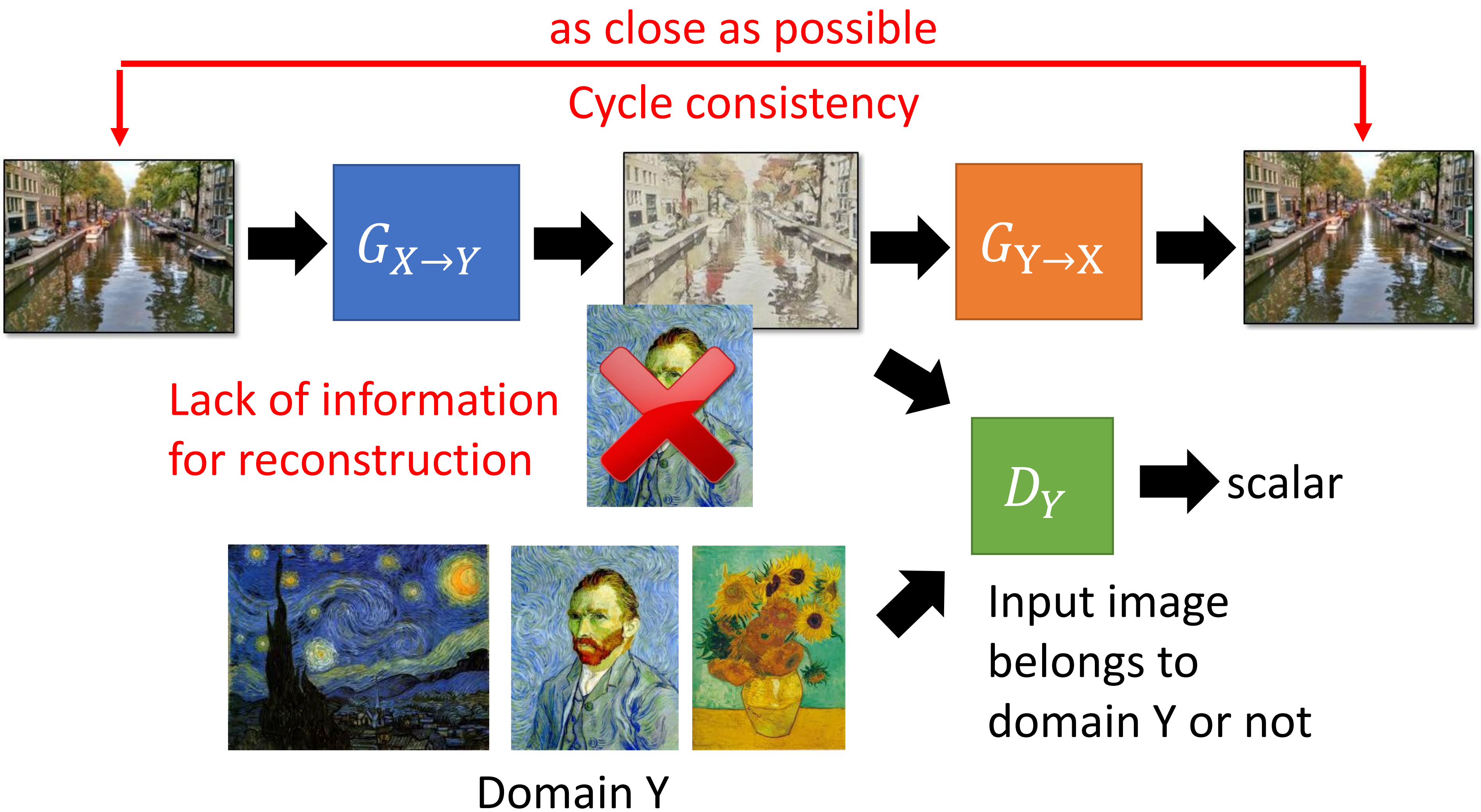




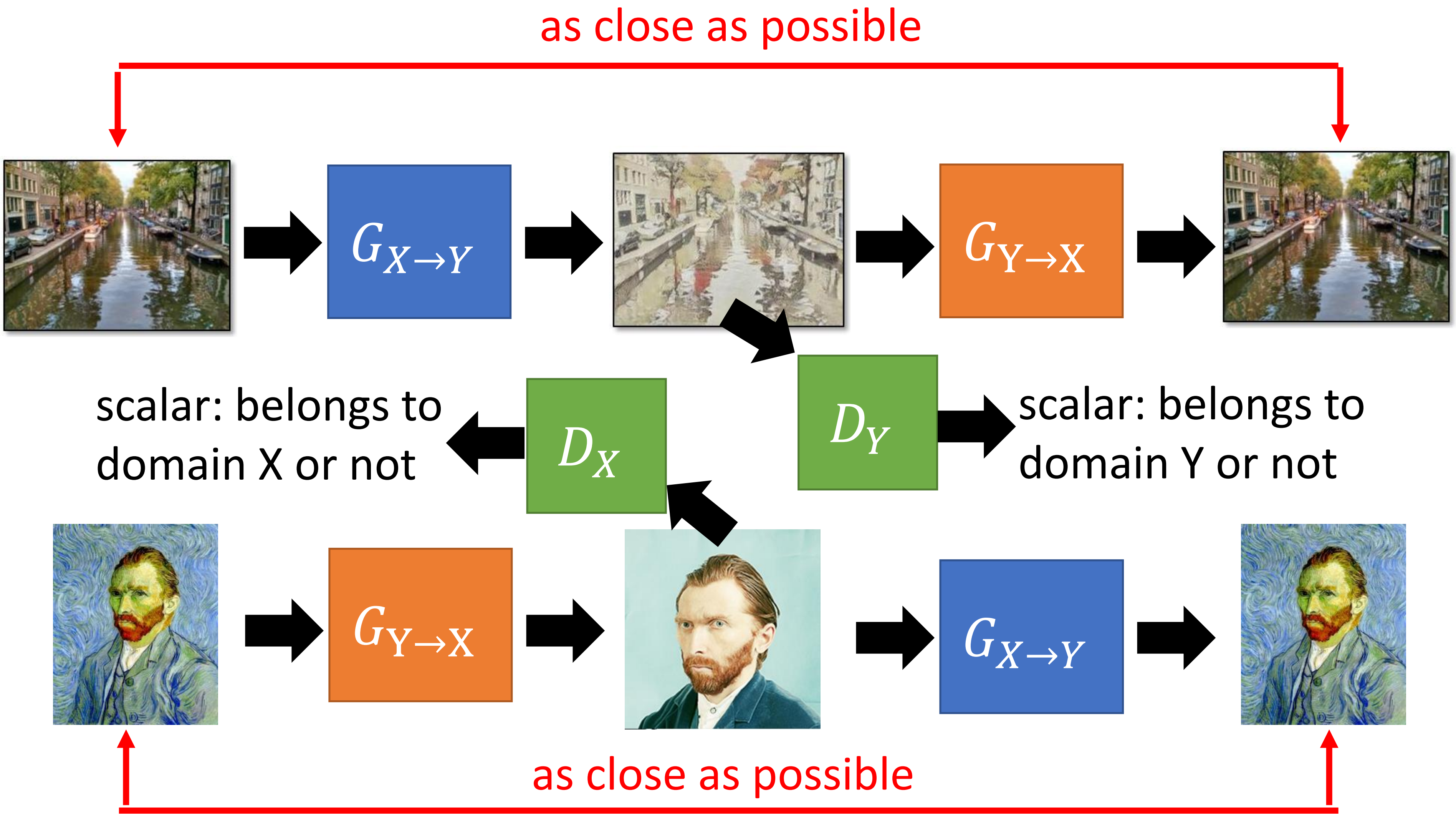
Baseline of DTN [Yaniv Taigman, et al., ICLR, 2017]

[Jun-Yan Zhu, et al., ICCV, 2017]

Direct Transformation



Direct Transformation

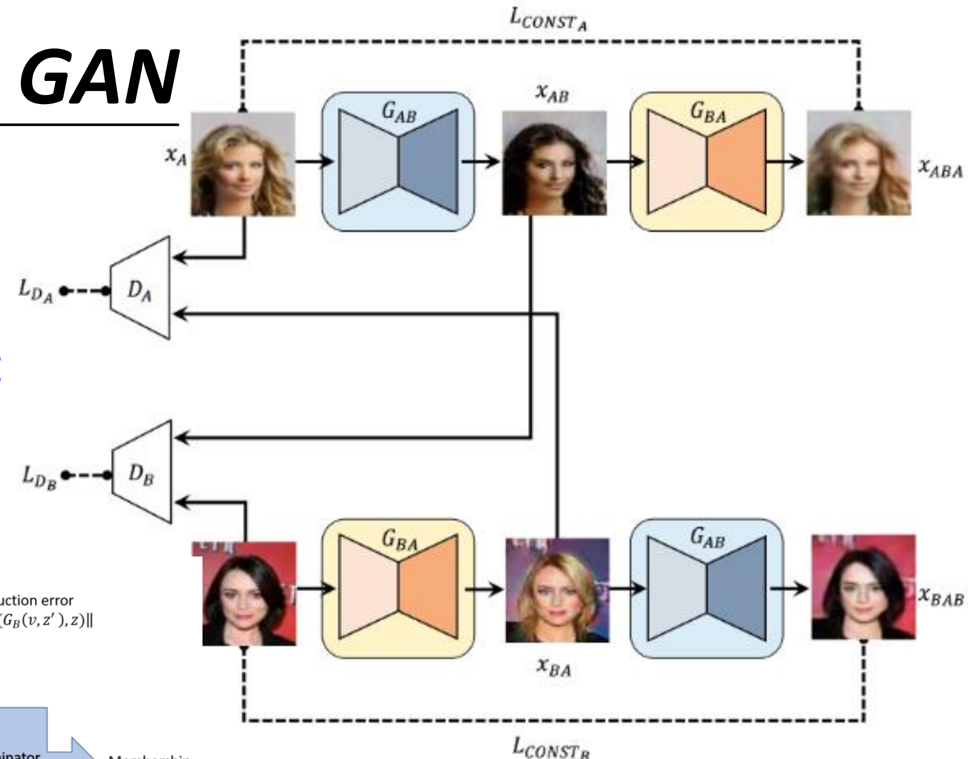


For multiple domains, considering starGAN

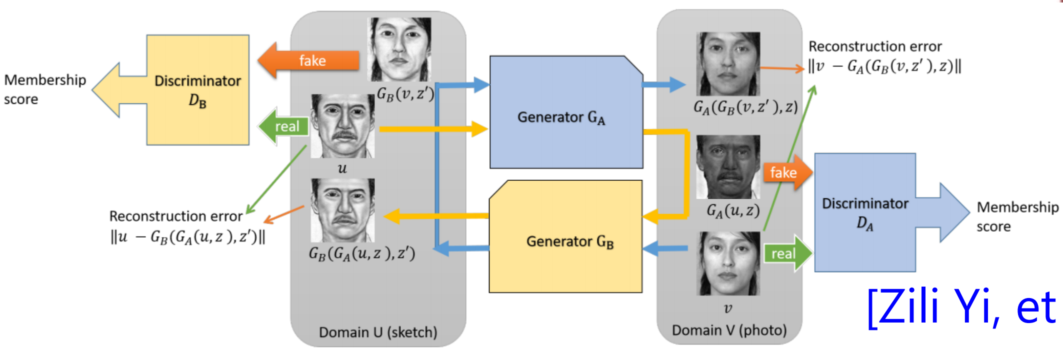
[Yunjey Choi, arXiv, 2017]

Disco GAN

[Taeksoo Kim, et al., ICML, 2017]

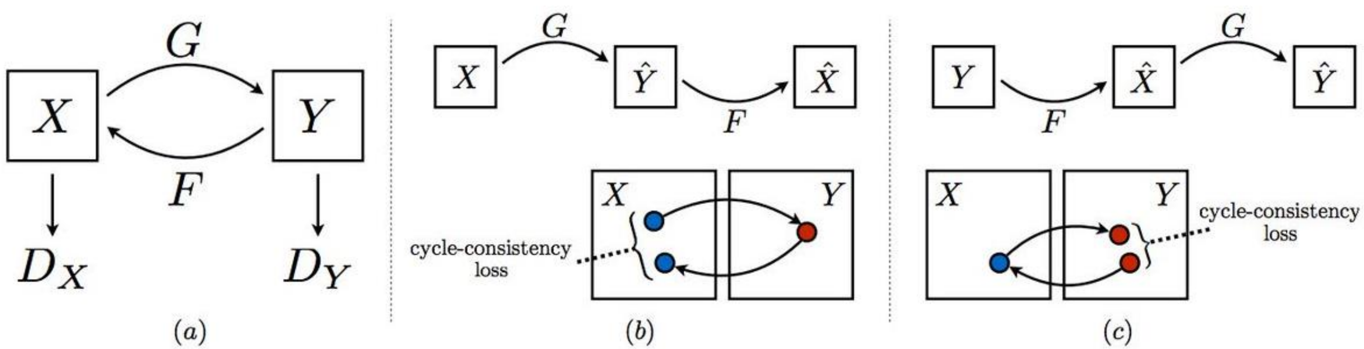


Dual GAN



[Zili Yi, et al., ICCV, 2017]

Cycle GAN

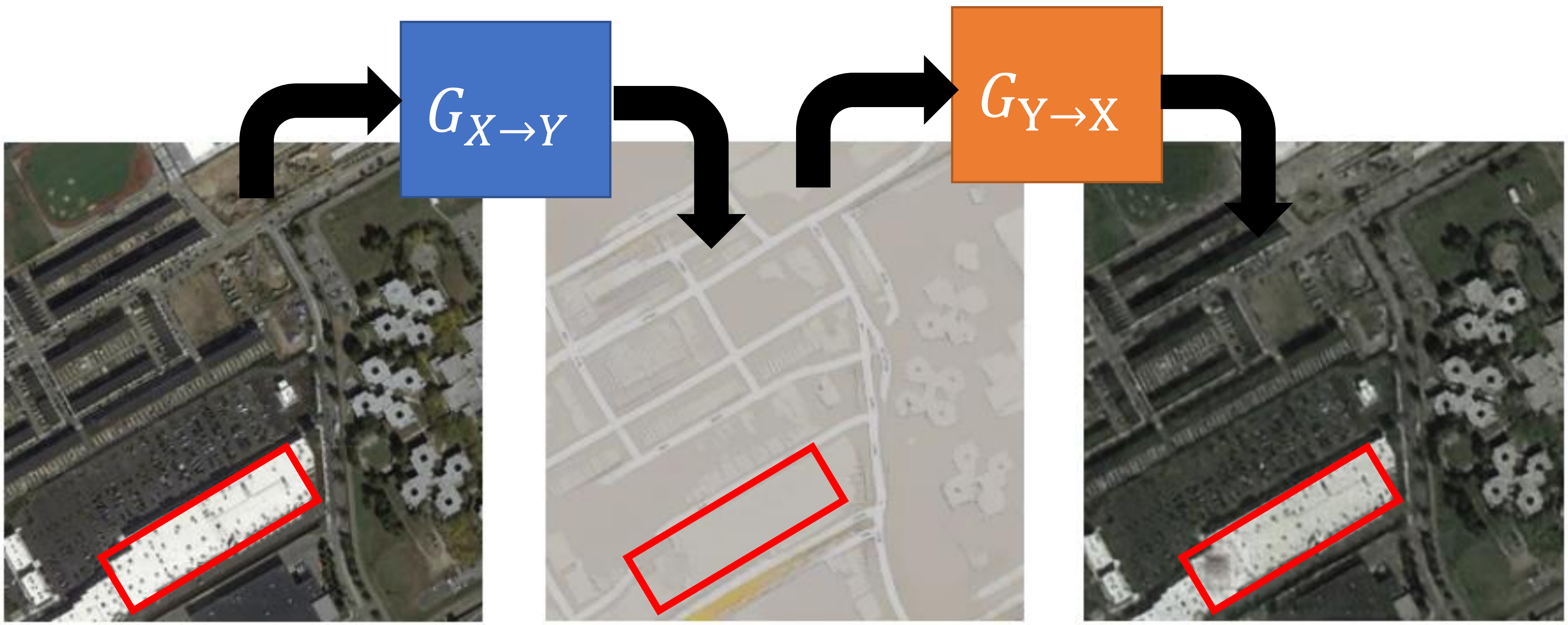


[Jun-Yan Zhu, et al., ICCV, 2017]

Issue of Cycle Consistency

- **CycleGAN: a Master of Steganography**

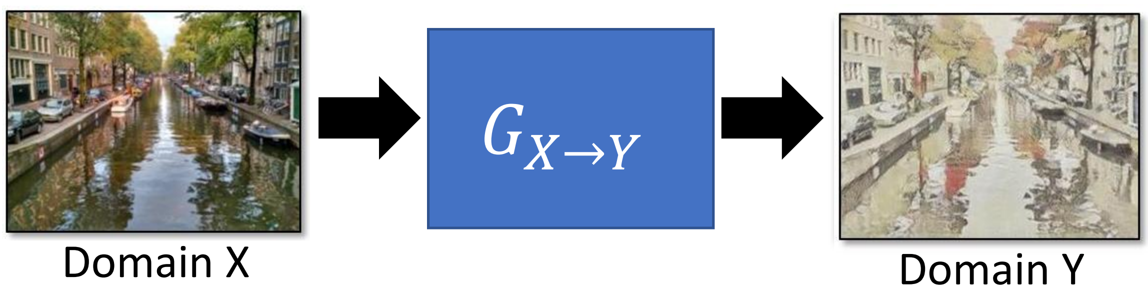
[Casey Chu, et al., NIPS workshop, 2017]



The information is hidden.

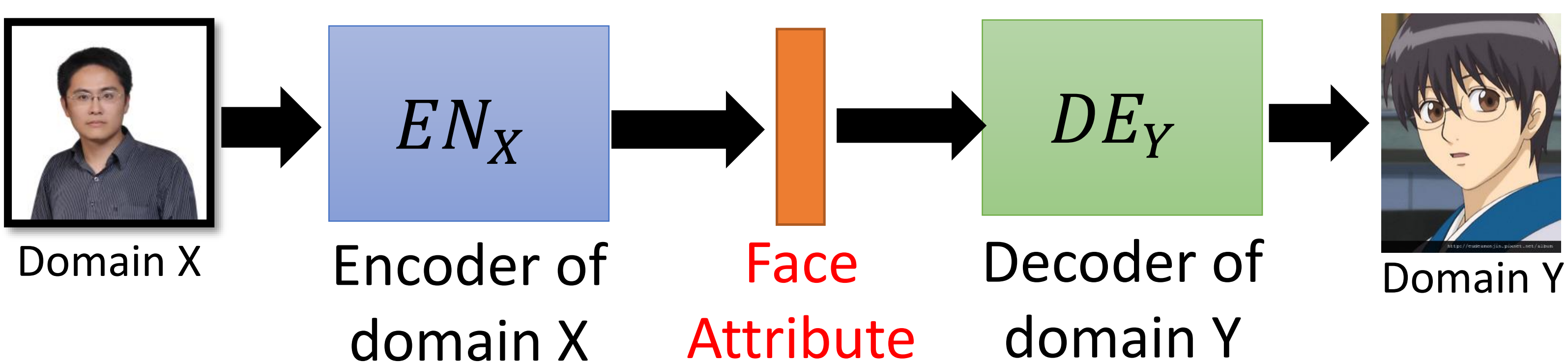
Unsupervised Conditional Generation

- Approach 1: Direct Transformation



For texture or color change

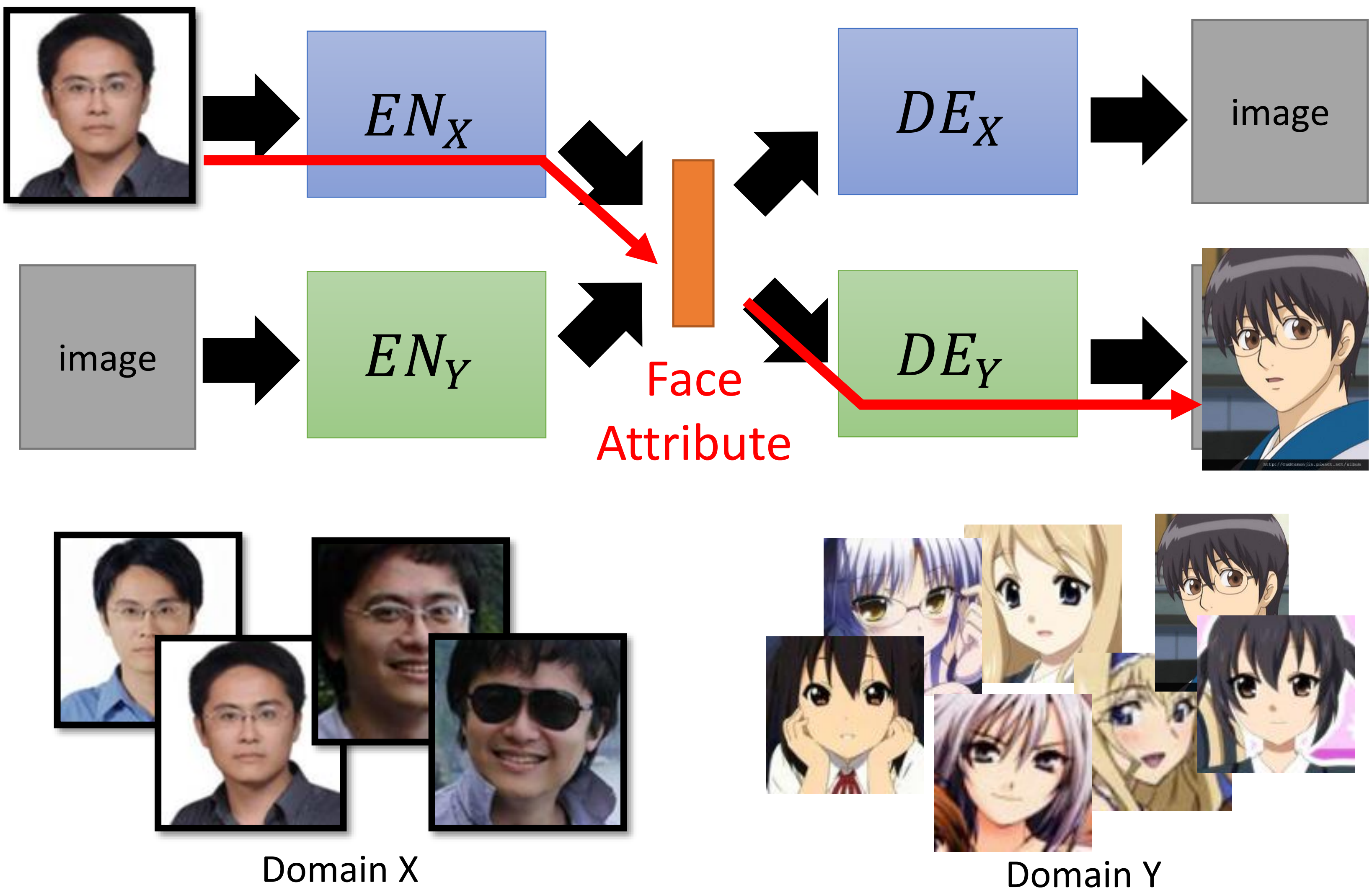
- Approach 2: Projection to Common Space



Larger change, only keep the semantics

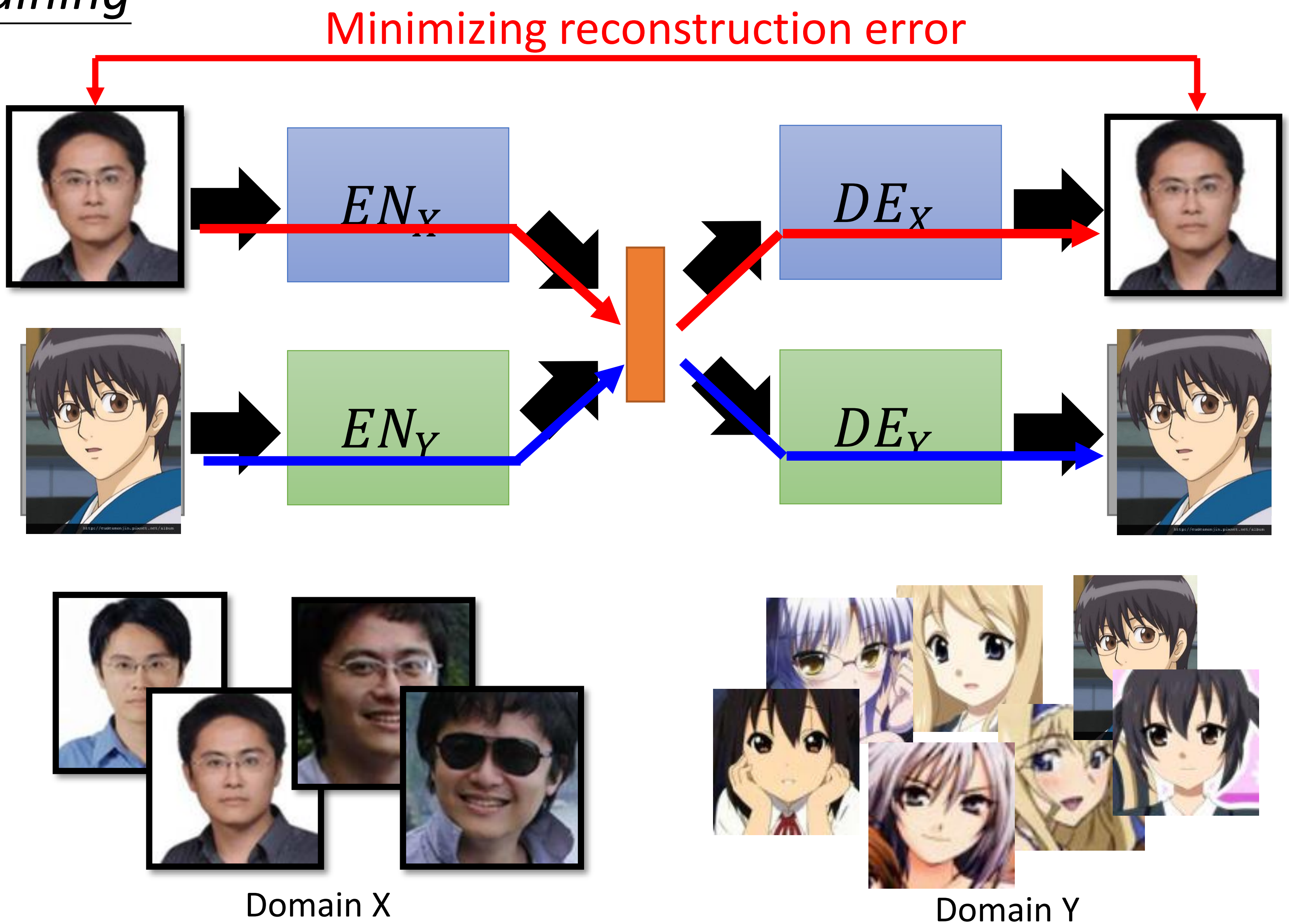
Projection to Common Space

Target



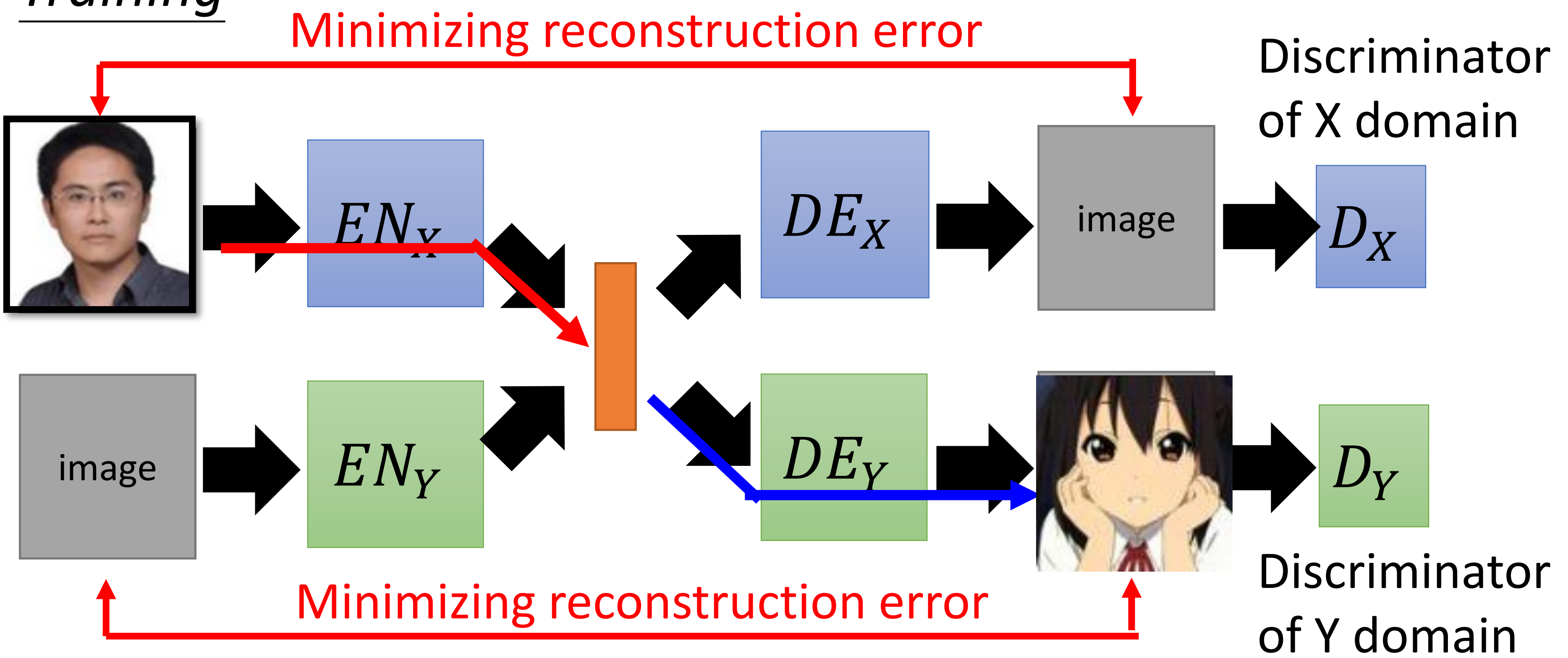
Projection to Common Space

Training



Projection to Common Space

Training

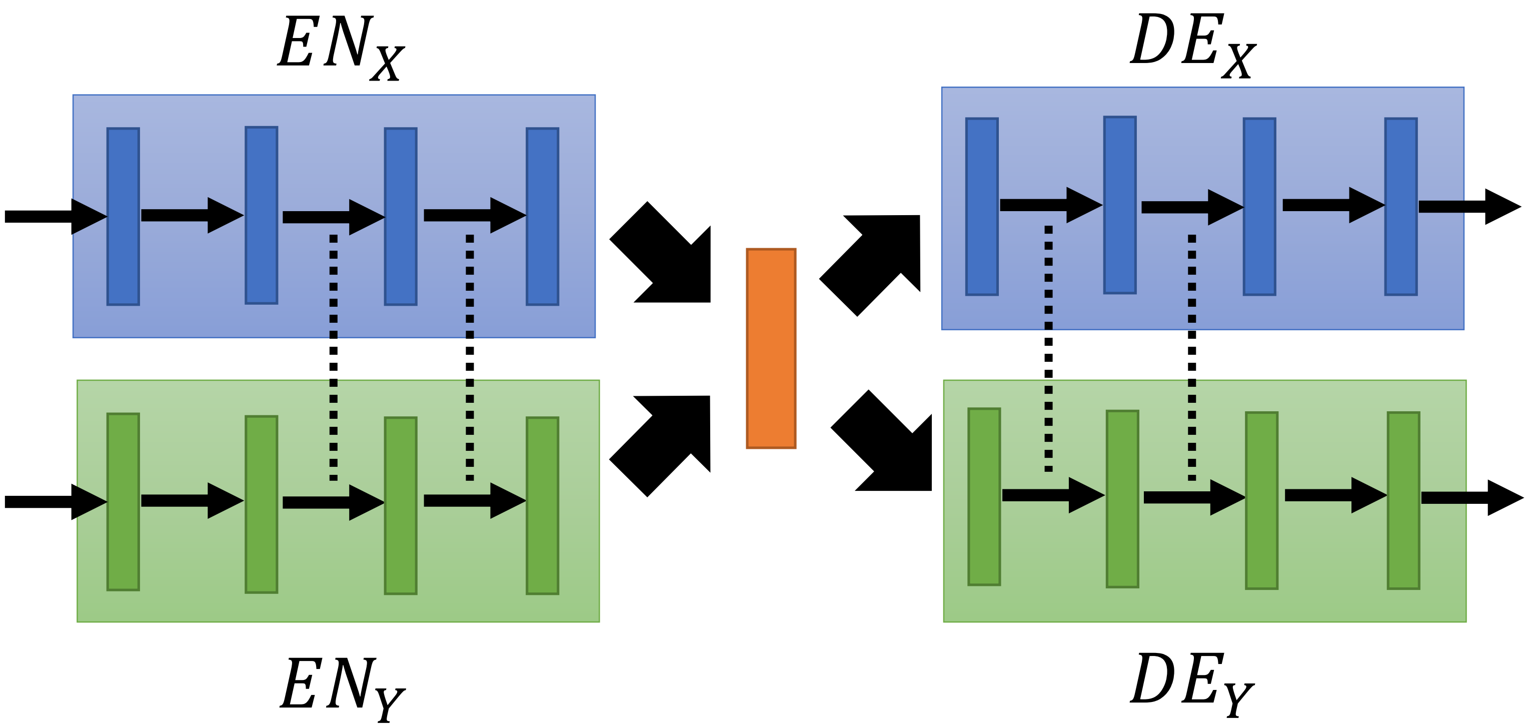


Because we train two auto-encoders separately …

The images with the same attribute may not project to the same position in the latent space.

Projection to Common Space

Training



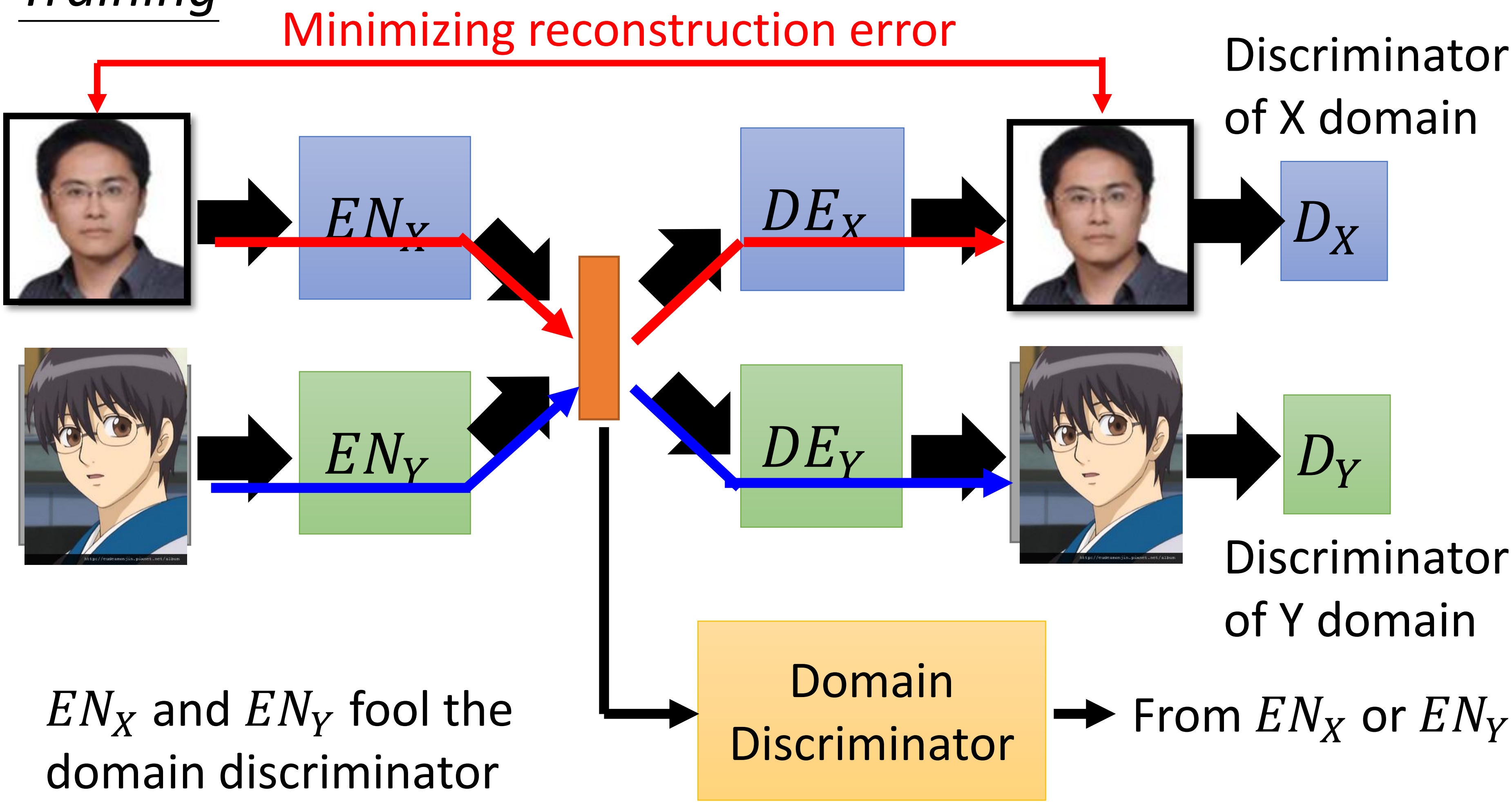
Sharing the parameters of encoders and decoders

Couple GAN[Ming-Yu Liu, et al., NIPS, 2016]

UNIT[Ming-Yu Liu, et al., NIPS, 2017]

Projection to Common Space

Training

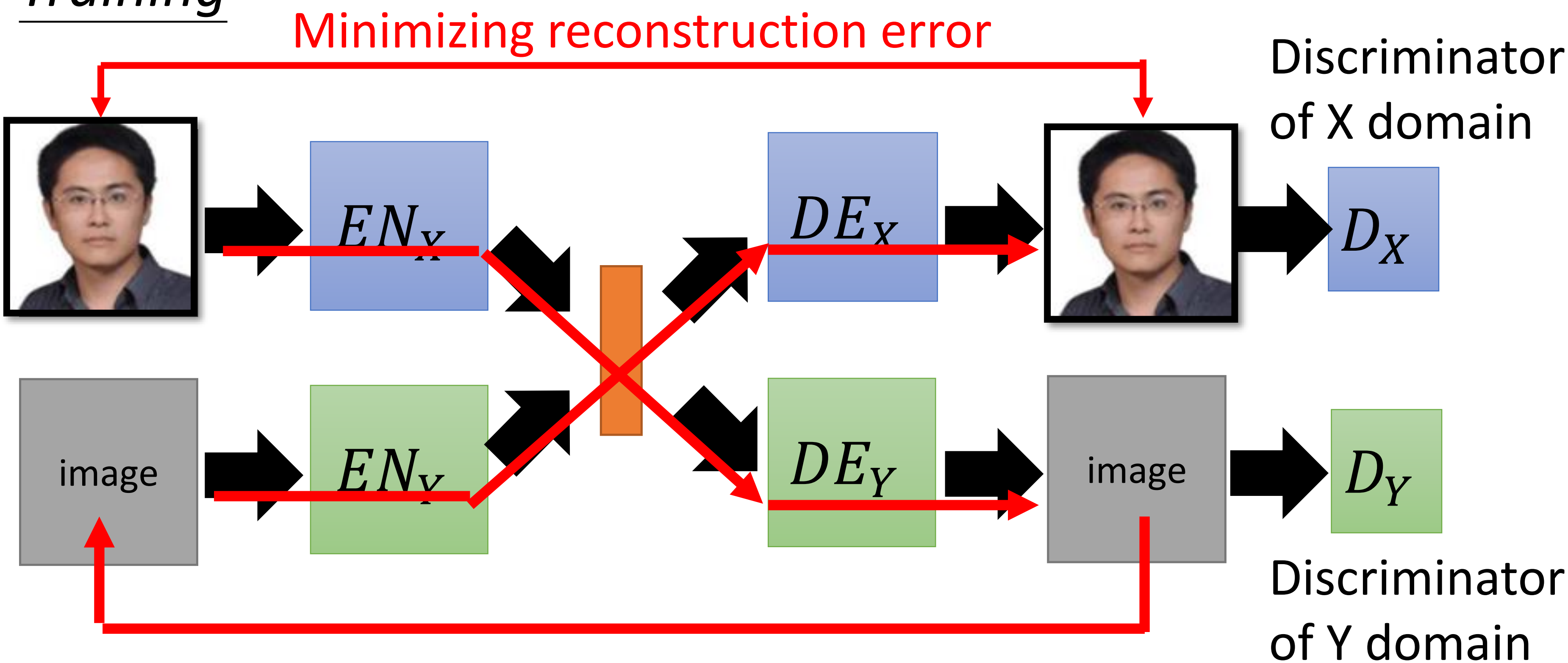


The domain discriminator forces the output of EN_X and EN_Y have the same distribution.

[Guillaume Lample, et al., NIPS, 2017]

Projection to Common Space

Training

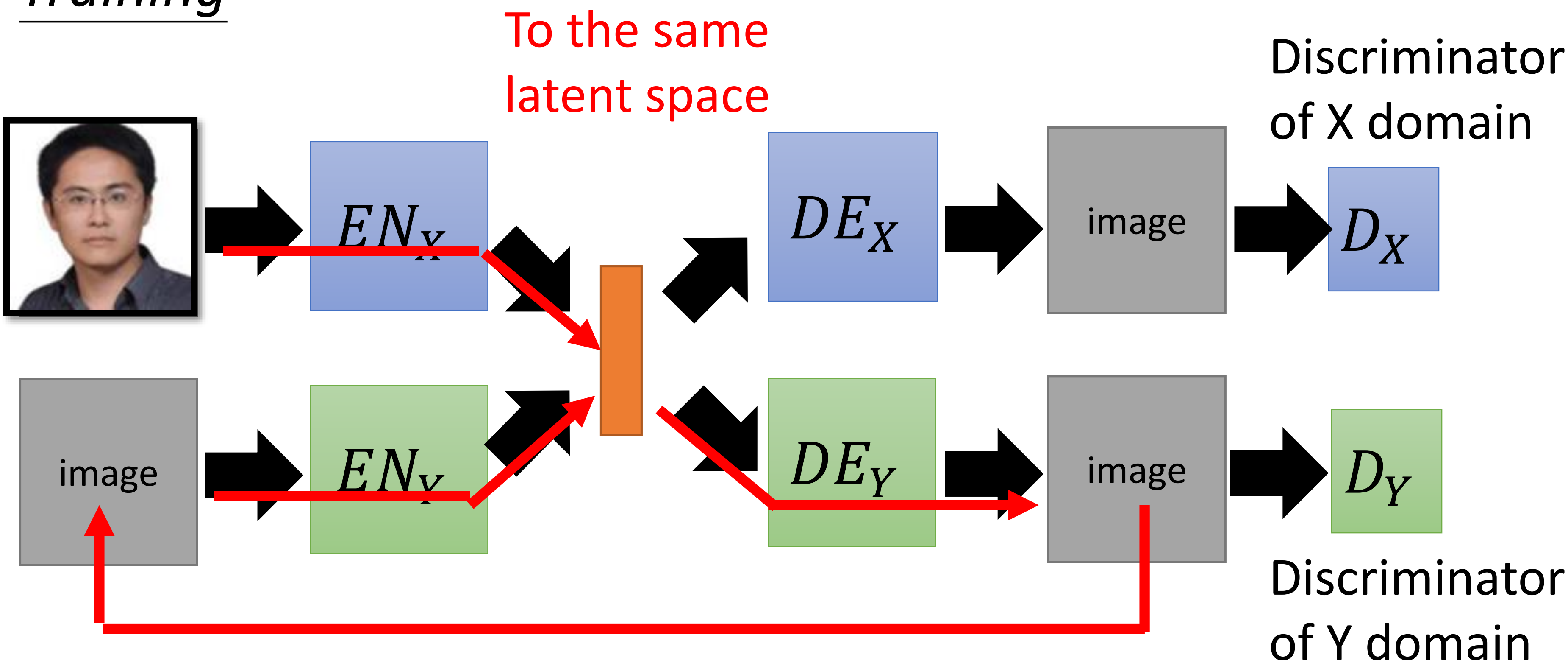


Cycle Consistency:

Used in ComboGAN [Asha Anoosheh, et al., arXiv, 017]

Projection to Common Space

Training



Semantic Consistency:

Used in DTN [Yaniv Taigman, et al., ICLR, 2017] and XGAN [Amélie Royer, et al., arXiv, 2017]

Outline of Part 1

Generation

Conditional Generation

Unsupervised Conditional Generation

Relation to Reinforcement Learning

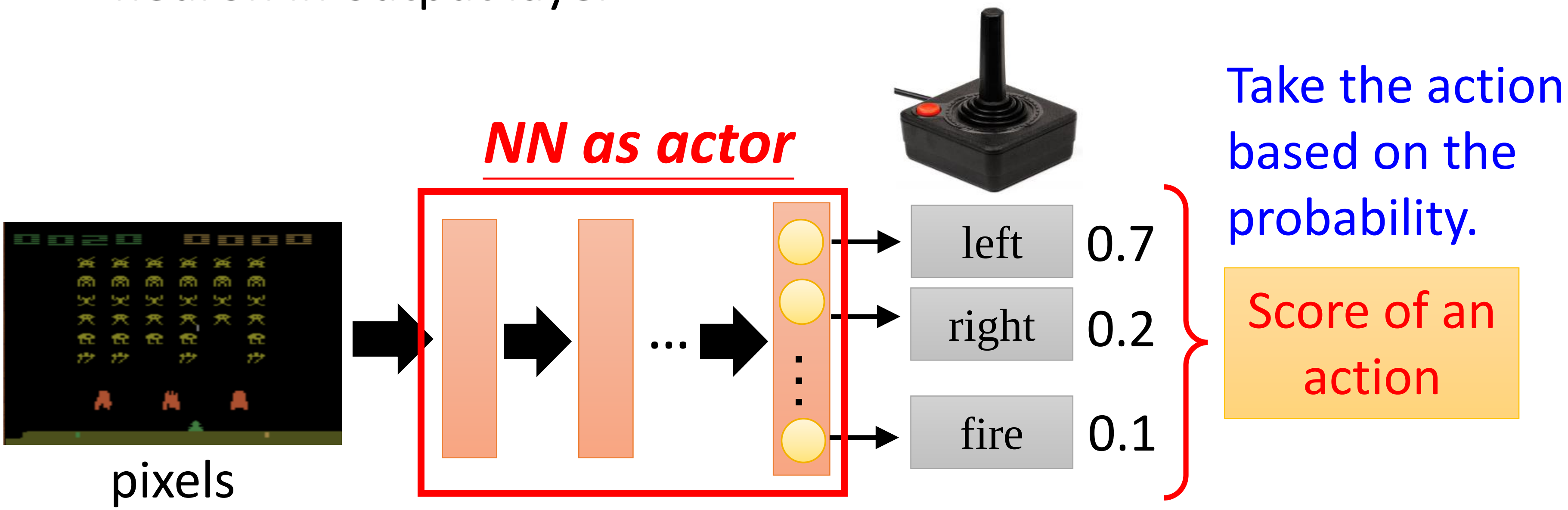
Basic Components

You cannot control

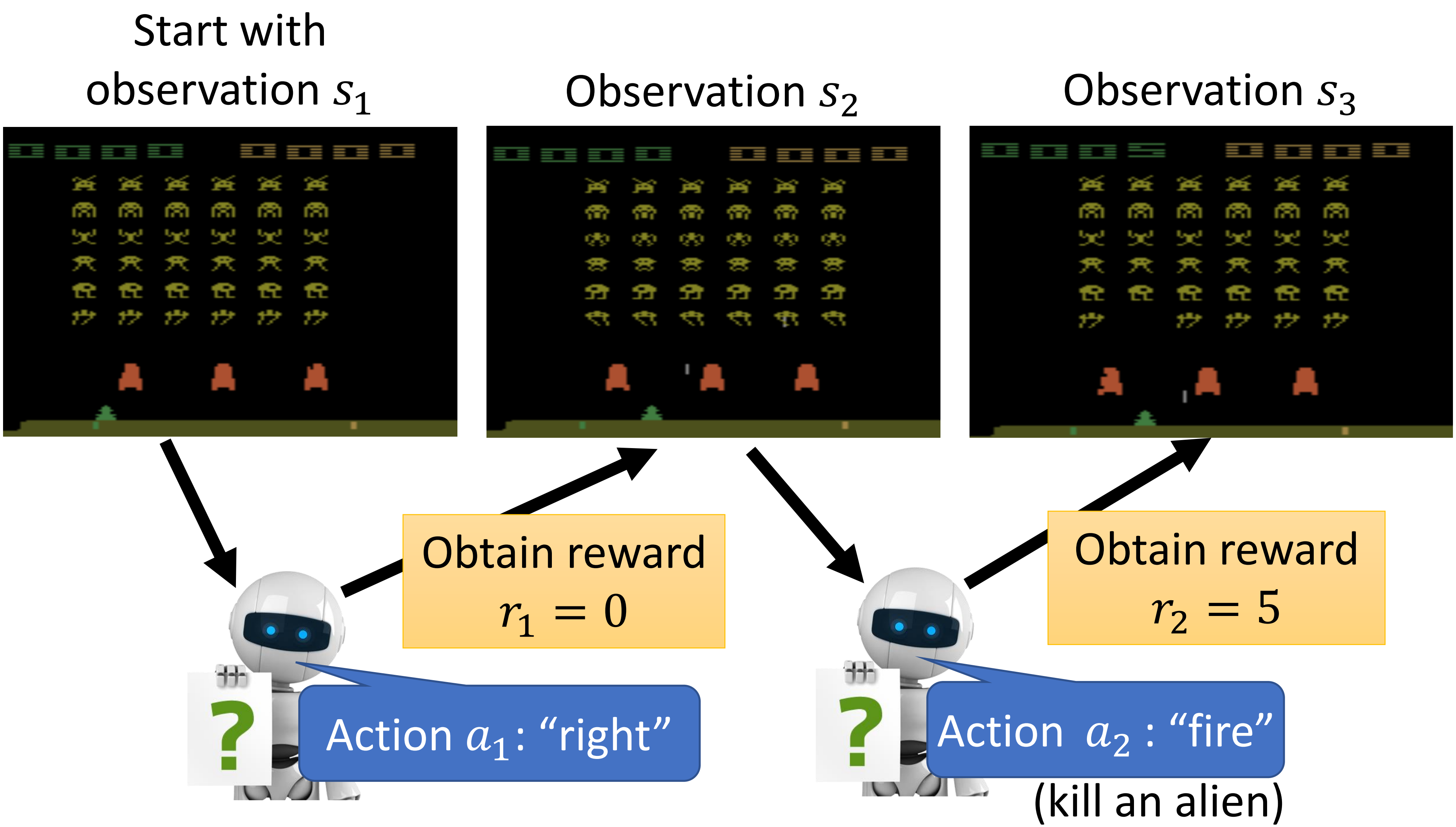
	Actor	Env	Reward Function
Video Game			Get 20 scores when killing a monster
Go	AlphaGo Google DeepMind		The rule of GO

Neural network as Actor

- Input of neural network: the observation of machine represented as a vector or a matrix
- Output neural network : each action corresponds to a neuron in output layer



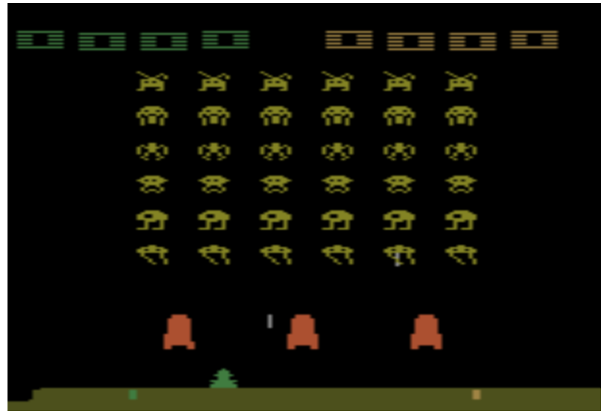
Example: Playing Video Game



Example: Playing Video Game

Start with observation s_1

Observation s_2



Observation s_3

After many turns

Game Over (spaceship destroyed)

Obtain reward r_T

Action a_T

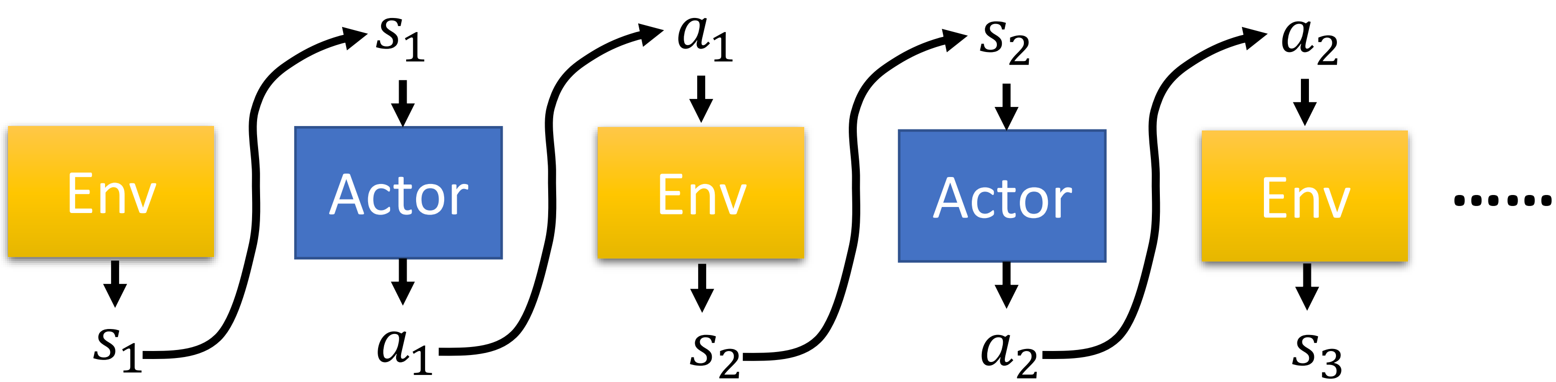
This is an ***episode***.

Total reward:

$$R = \sum_{t=1}^{T} r_t$$

We want the total reward be maximized.

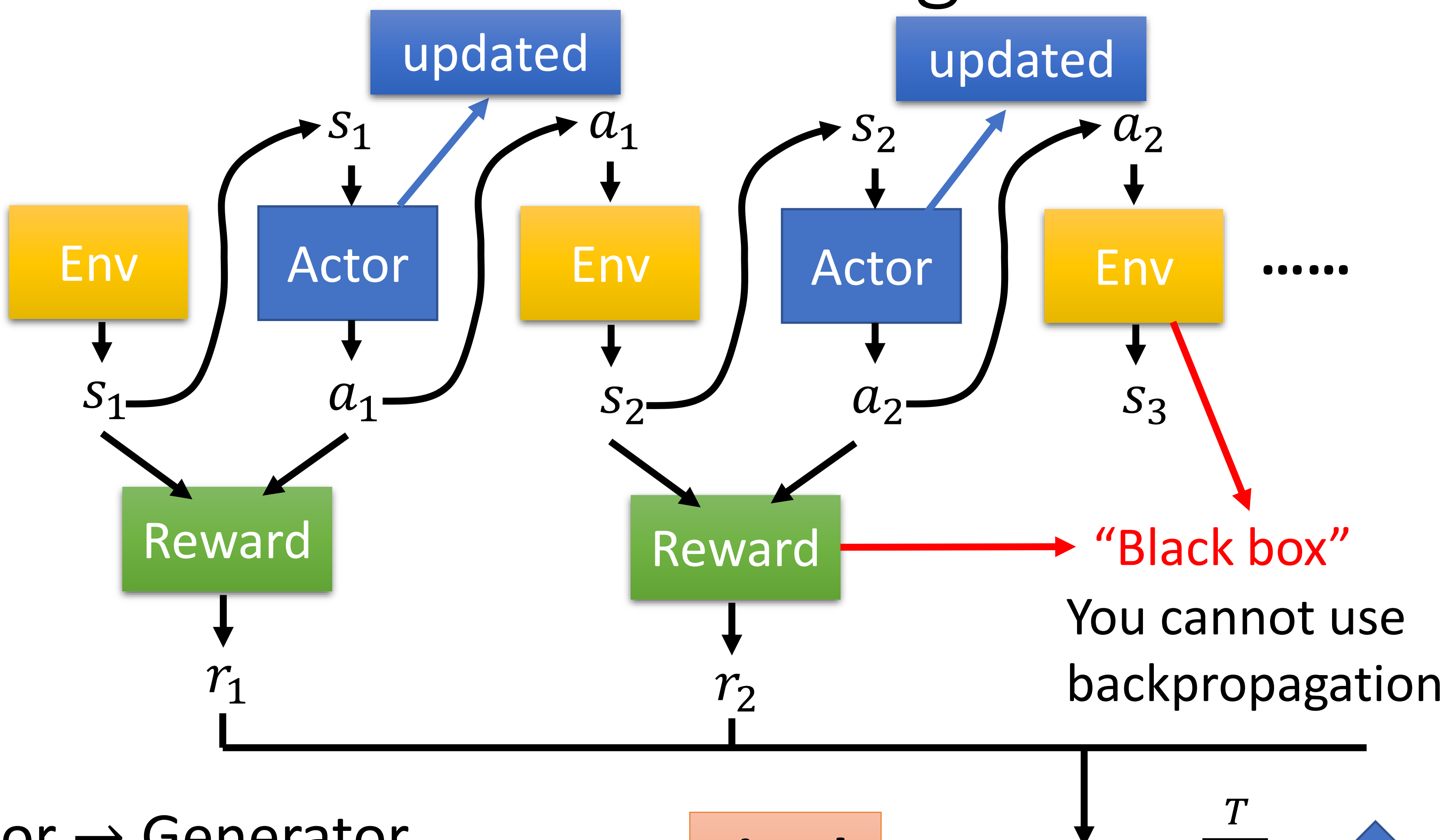
Actor, Environment, Reward



Trajectory

$$\tau = \{s_1, a_1, s_2, a_2, \cdots, s_T, a_T\}$$

Reinforcement Learning v.s. GAN



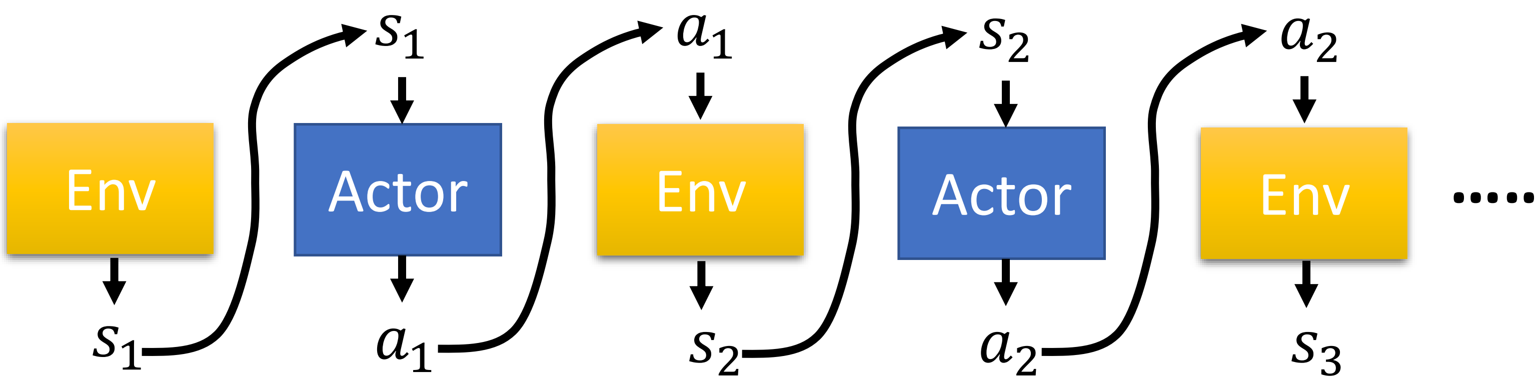
Actor → Generator

Reward Function → Discriminator

Fixed

$$R(\tau) = \sum_{t=1}^{T} r_t$$

Imitation Learning

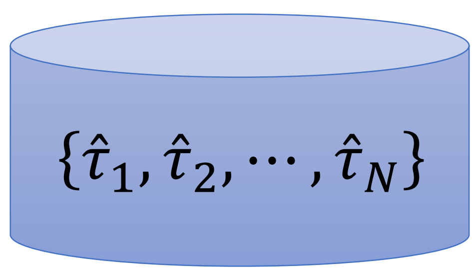


reward function is not available

We have demonstration of the expert.

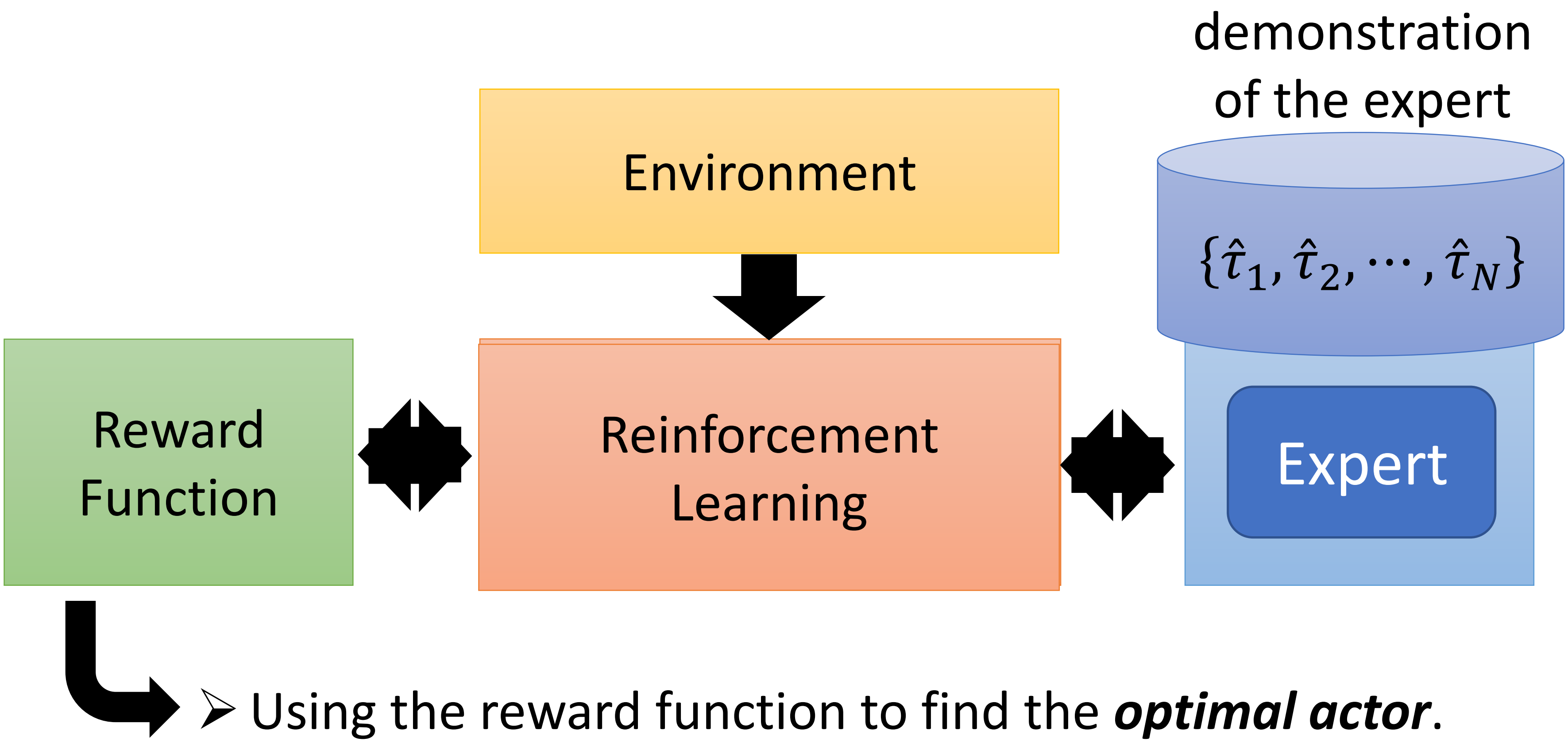
Self driving: record human drivers

Robot: grab the arm of robot



Each $\hat{\tau}$ is a trajectory of the expert.

Inverse Reinforcement Learning

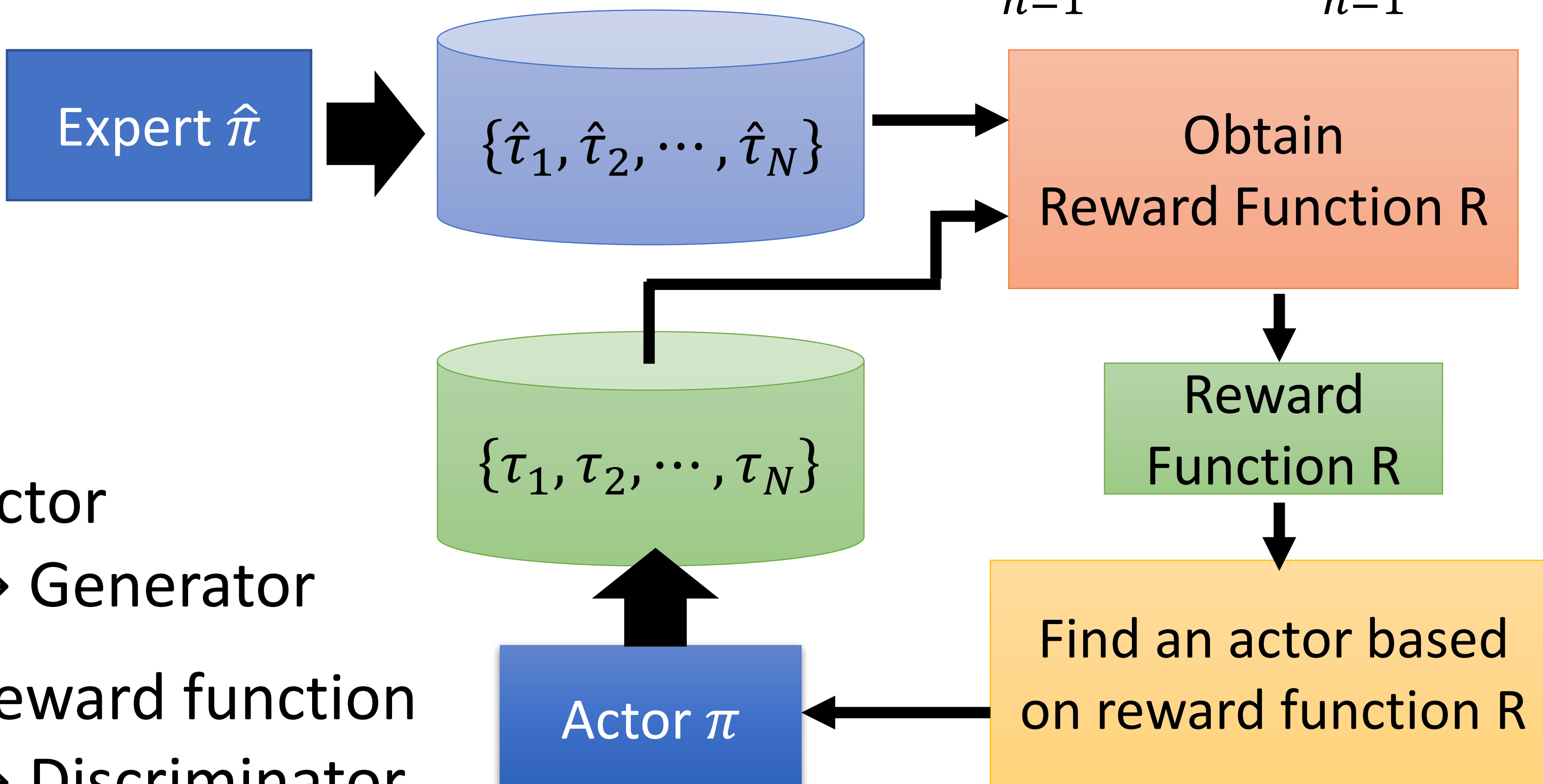


- Using the reward function to find the ***optimal actor***.
- Modeling reward can be easier. Simple reward function can lead to complex policy.

Framework of IRL

The expert is always the best.

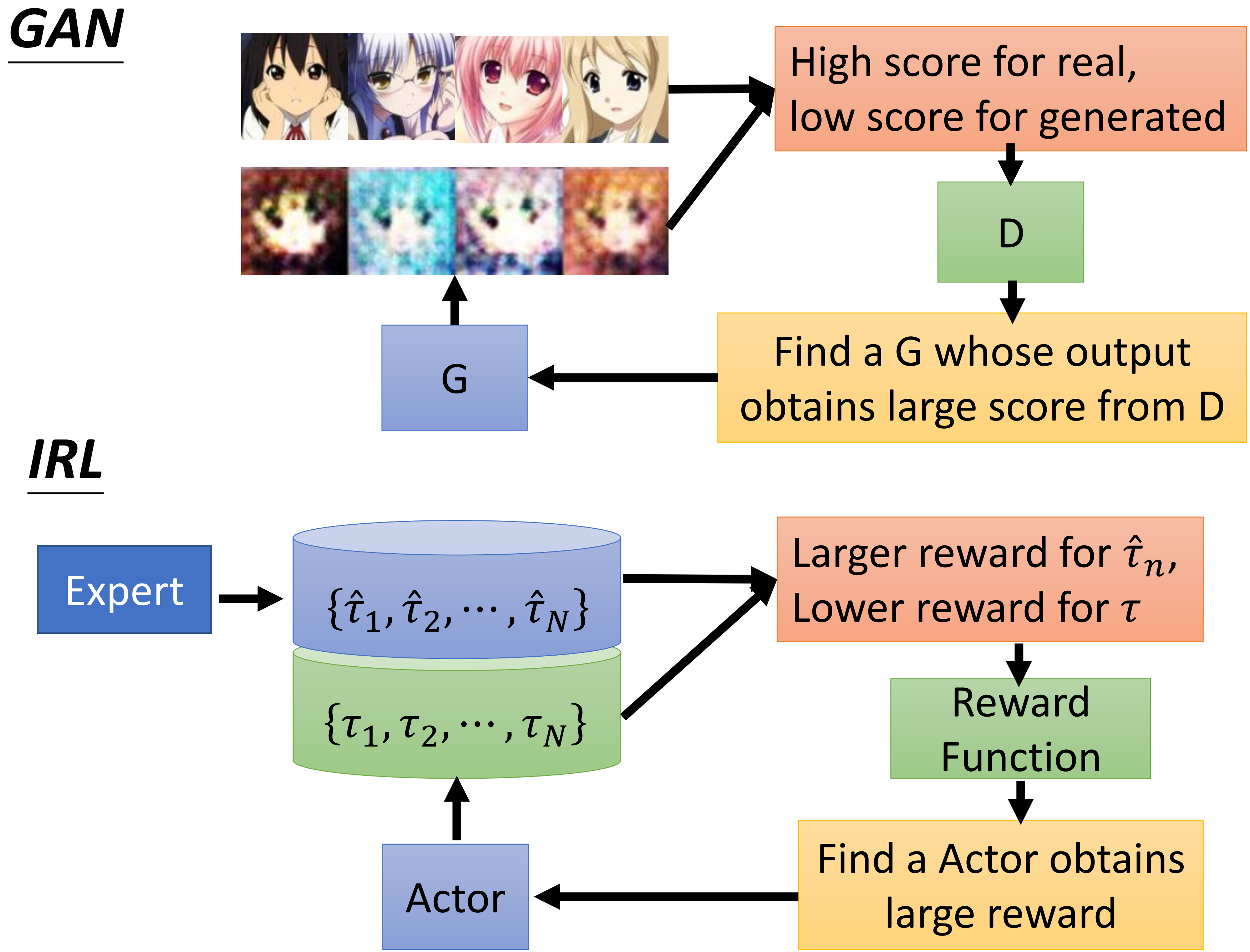
$$\sum_{n=1}^{N} R(\hat{\tau}_n) > \sum_{n=1}^{N} R(\tau)$$



By Reinforcement learning

Actor → Generator

Reward function → Discriminator


GAN
High score for real,
low score for generated
D
Find a G whose output
obtains large score from D
G
IRL
Expert
$\{\hat{\tau}_1, \hat{\tau}_2, \cdots, \hat{\tau}_N\}$
$\{\tau_1, \tau_2, \cdots, \tau_N\}$
Larger reward for $\hat{\tau}_n$,
Lower reward for τ
Reward Function
Find a Actor obtains
large reward
Actor

Concluding Remarks

- Generation
- Conditional Generation
- Unsupervised Conditional Generation
- Relation to Reinforcement Learning

Reference

- **Generation**

  - Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio, Generative Adversarial Networks, NIPS, 2014
  - Sebastian Nowozin, Botond Cseke, Ryota Tomioka, “f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization”, NIPS, 2016
  - Martin Arjovsky, Soumith Chintala, Léon Bottou, Wasserstein GAN, arXiv, 2017
  - Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, Aaron Courville, Improved Training of Wasserstein GANs, NIPS, 2017
  - Junbo Zhao, Michael Mathieu, Yann LeCun, Energy-based Generative Adversarial Network, arXiv, 2016
  - Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, Olivier Bousquet, “Are GANs Created Equal? A Large-Scale Study”, arXiv, 2017
  - Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen Improved Techniques for Training GANs, NIPS, 2016
  - Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, Sepp Hochreiter, GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium, NIPS, 2017

Reference

- **Generation**
  - Naveen Kodali, Jacob Abernethy, James Hays, Zsolt Kira, “On Convergence and Stability of GANs”, arXiv, 2017
  - Xiang Wei, Boqing Gong, Zixia Liu, Wei Lu, Liqiang Wang, Improving the Improved Training of Wasserstein GANs: A Consistency Term and Its Dual Effect, ICLR, 2018
  - Takeru Miyato, Toshiki Kataoka, Masanori Koyama, Yuichi Yoshida, Spectral Normalization for Generative Adversarial Networks, ICLR, 2018

Reference

- **Generational Generation**

  - Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, Honglak Lee, Generative Adversarial Text to Image Synthesis, ICML, 2016
  - Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros, Image-to-Image Translation with Conditional Adversarial Networks, CVPR, 2017
  - Michael Mathieu, Camille Couprie, Yann LeCun, Deep multi-scale video prediction beyond mean square error, arXiv, 2015
  - Mehdi Mirza, Simon Osindero, Conditional Generative Adversarial Nets, arXiv, 2014
  - Takeru Miyato, Masanori Koyama, cGANs with Projection Discriminator, ICLR, 2018
  - Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, Dimitris Metaxas, StackGAN++: Realistic Image Synthesis with Stacked Generative Adversarial Networks, arXiv, 2017
  - Augustus Odena, Christopher Olah, Jonathon Shlens, Conditional Image Synthesis With Auxiliary Classifier GANs, ICML, 2017

Reference

- **Generational Generation**
  - Yaroslav Ganin, Victor Lempitsky, Unsupervised Domain Adaptation by Backpropagation, ICML, 2015
  - Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, Domain-Adversarial Training of Neural Networks, JMLR, 2016

Reference

- **Unsupervised Conditional Generation**

  - Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A. Efros, Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, ICCV, 2017
  - Zili Yi, Hao Zhang, Ping Tan, Minglun Gong, DualGAN: Unsupervised Dual Learning for Image-to-Image Translation, ICCV, 2017
  - Tomer Galanti, Lior Wolf, Sagie Benaim, The Role of Minimal Complexity Functions in Unsupervised Learning of Semantic Mappings, ICLR, 2018
  - Yaniv Taigman, Adam Polyak, Lior Wolf, Unsupervised Cross-Domain Image Generation, ICLR, 2017
  - Asha Anoosheh, Eirikur Agustsson, Radu Timofte, Luc Van Gool, ComboGAN: Unrestrained Scalability for Image Domain Translation, arXiv, 2017
  - Amélie Royer, Konstantinos Bousmalis, Stephan Gouws, Fred Bertsch, Inbar Mosseri, Forrester Cole, Kevin Murphy, XGAN: Unsupervised Image-to-Image Translation for Many-to-Many Mappings, arXiv, 2017

Reference

- **Unsupervised Conditional Generation**

  - Guillaume Lample, Neil Zeghidour, Nicolas Usunier, Antoine Bordes, Ludovic Denoyer, Marc'Aurelio Ranzato, Fader Networks: Manipulating Images by Sliding Attributes, NIPS, 2017
  - Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, Jiwon Kim, Learning to Discover Cross-Domain Relations with Generative Adversarial Networks, ICML, 2017
  - Ming-Yu Liu, Oncel Tuzel, "Coupled Generative Adversarial Networks", NIPS, 2016
  - Ming-Yu Liu, Thomas Breuel, Jan Kautz, Unsupervised Image-to-Image Translation Networks, NIPS, 2017
  - Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, Jaegul Choo, StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation, arXiv, 2017

Generative Adversarial Network

and its Applications to Signal Processing
and Natural Language Processing

Part II: Speech Signal Processing

Outline of Part II

Speech Signal Generation

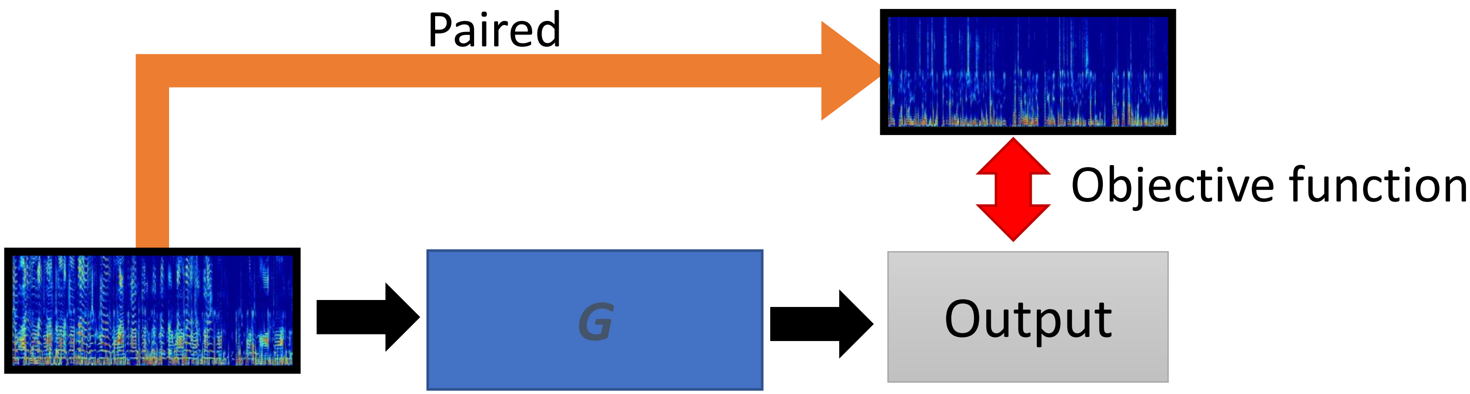
- Speech enhancement
- Postfilter, speech synthesis, voice conversion

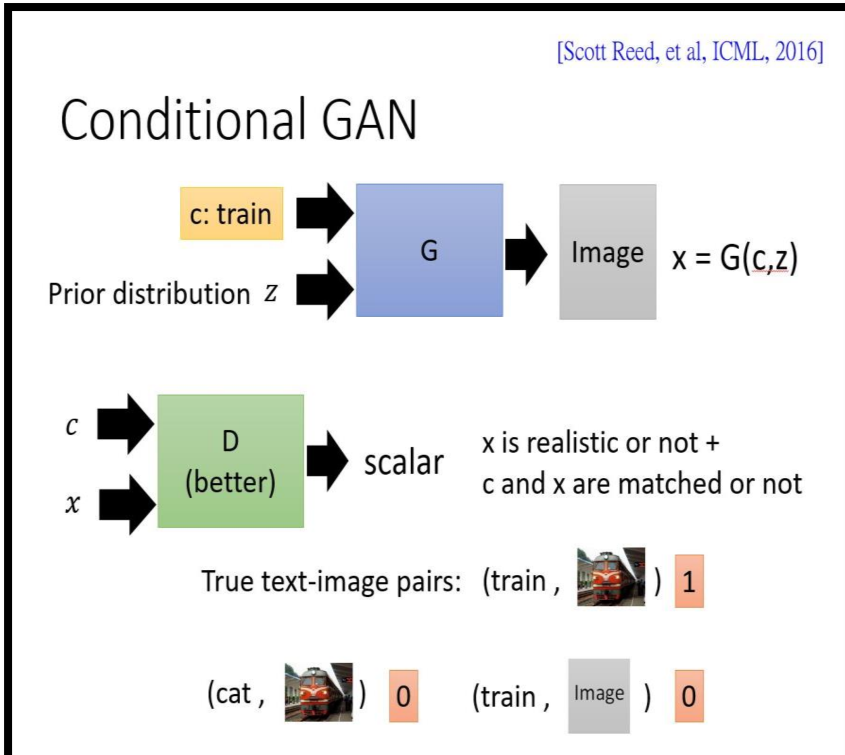
Speech Signal Recognition

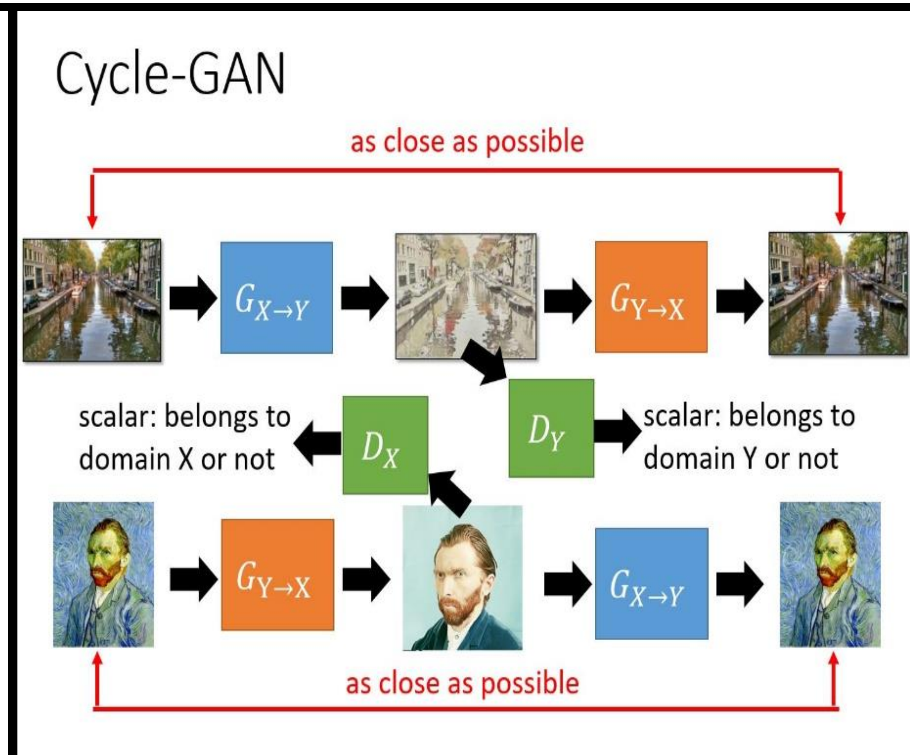
- Speech recognition
- Speaker recognition
- Speech emotion recognition
- Lip reading

Conclusion

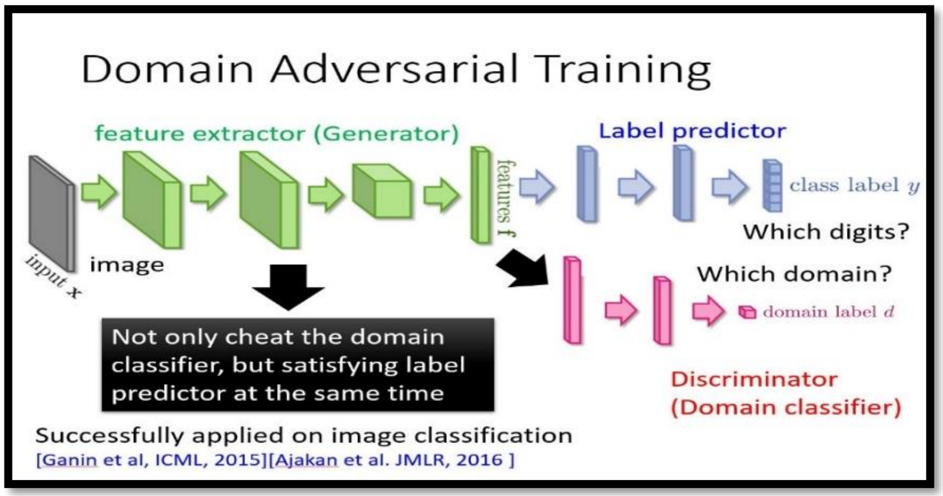
Speech Signal Generation (Regression Task)



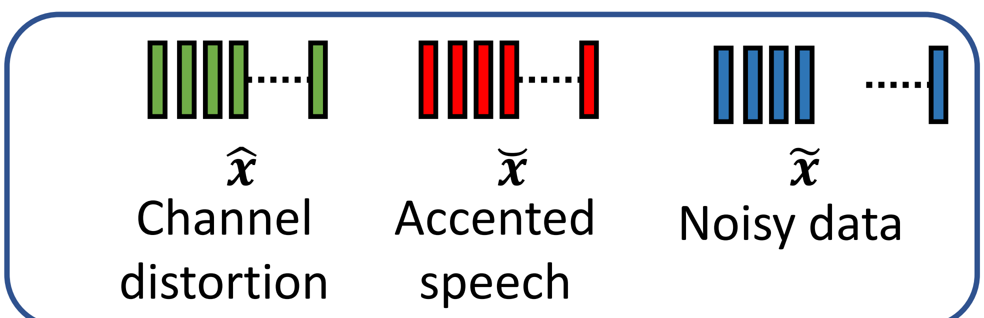


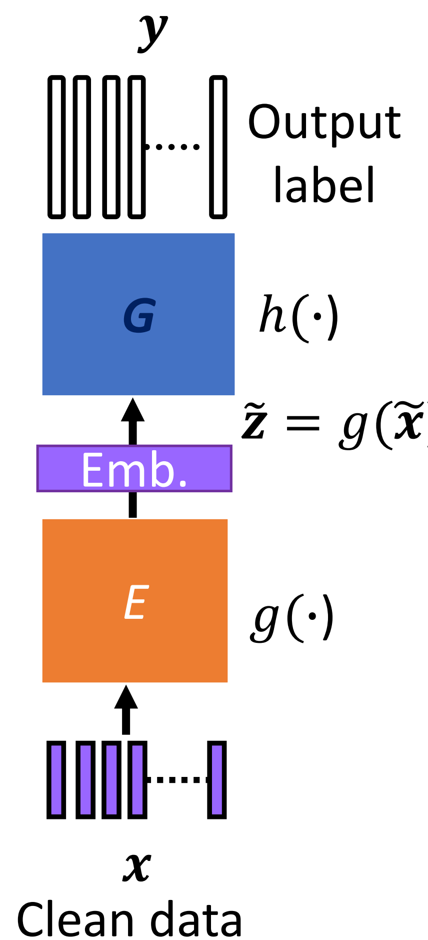


Speech, Speaker, Emotion Recognition and Lip-reading (Classification Task)



Acoustic Mismatch





Outline of Part II

Speech Signal Generation

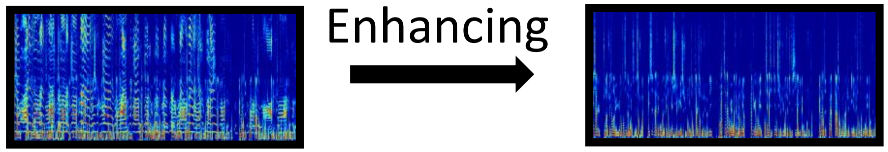
- Speech enhancement
- Postfilter, speech synthesis, voice conversion

Speech Signal Recognition

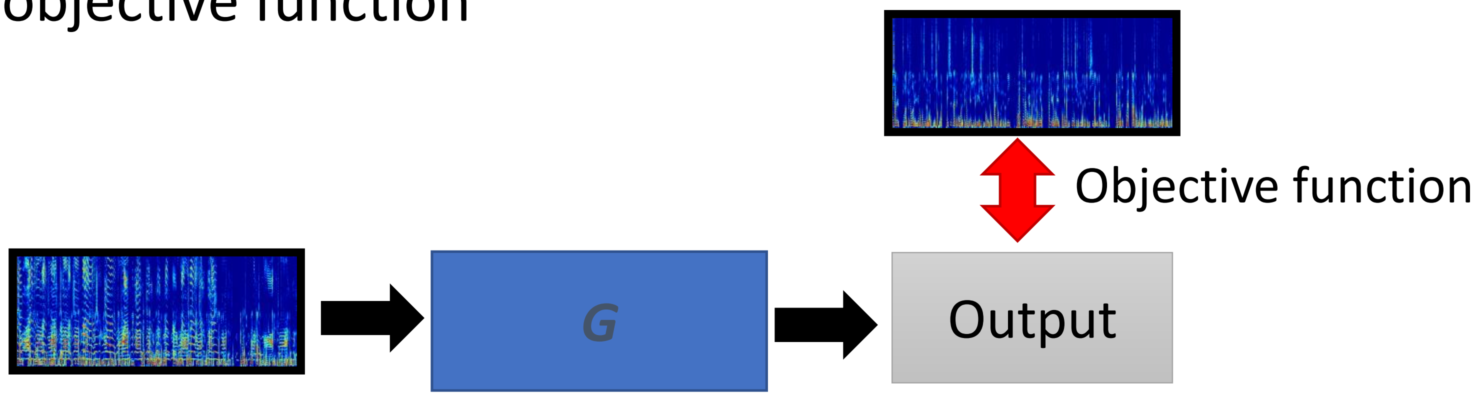
- Speech recognition
- Speaker recognition
- Speech emotion recognition
- Lip reading

Conclusion

Speech Enhancement



- Typical objective function

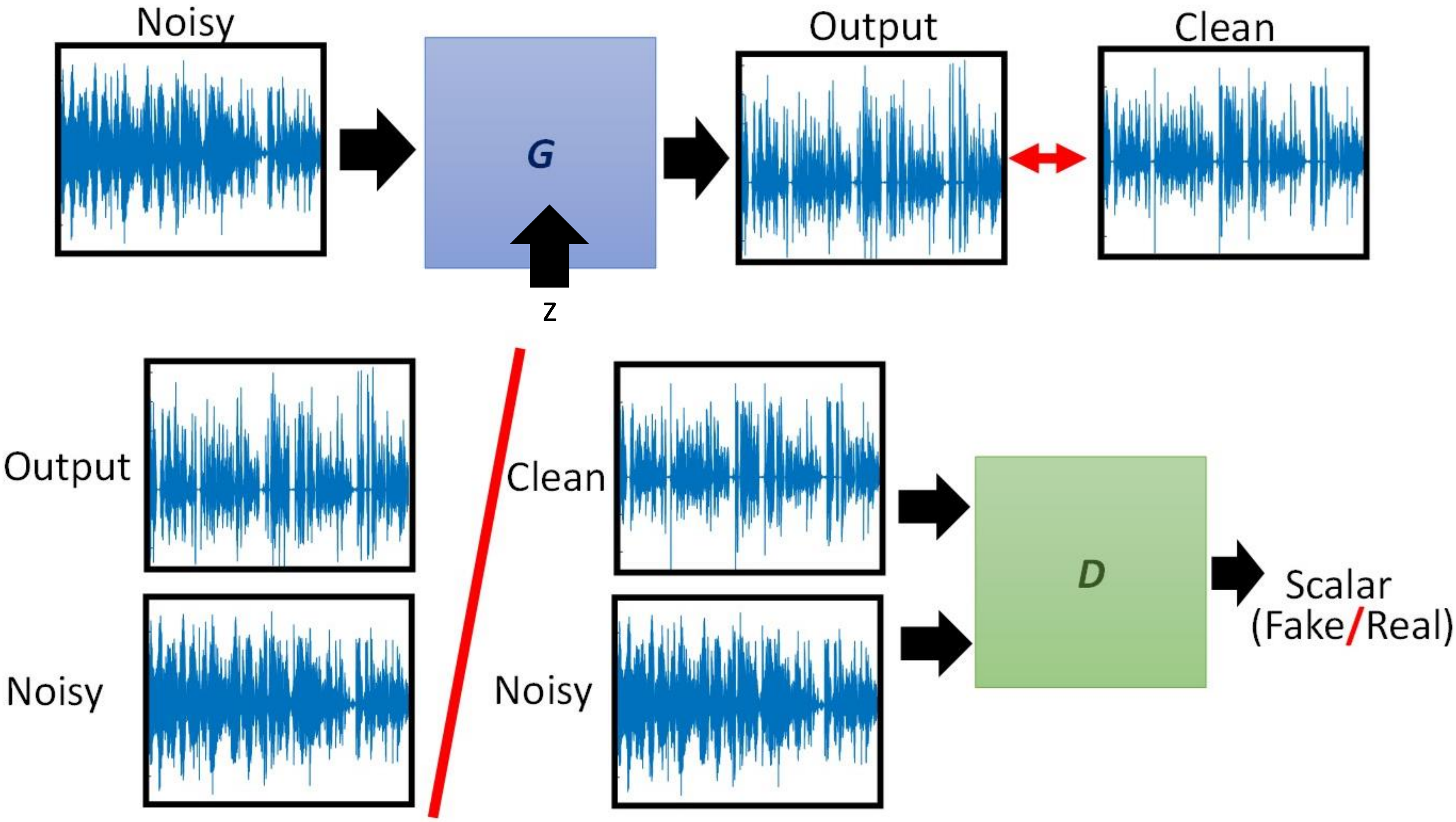


  - ➢Model structures of ***G***: DNN [Wang et al. NIPS 2012; Xu et al., SPL 2014], DDAE [Lu et al., Interspeech 2013], RNN (LSTM) [Chen et al., Interspeech 2015; Weninger et al., LVA/ICA 2015], CNN [Fu et al., Interspeech 2016].

- Typical objective function
  - ➢ Mean square error (MSE) [Xu et al., TASLP 2015], L1 [Pascual et al., Interspeech 2017], likelihood [Chai et al., MLSP 2017], STOI [Fu et al., TASLP 2018].
  - ➢ GAN is used as a new objective function to estimate the parameters in ***G***.

Speech Enhancement

- Speech enhancement GAN (SEGAN) [Pascual et al., Interspeech 2017]



Speech Enhancement (SEGAN)

- Experimental results

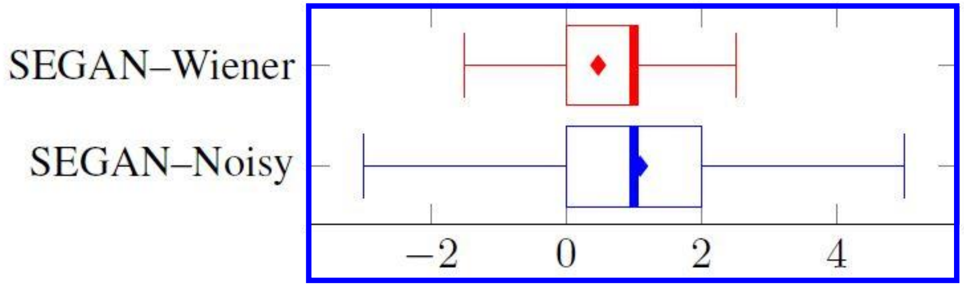
Table 1: Objective evaluation results.

Metric	Noisy	Wiener	SEGAN
PESQ	1.97	2.22	2.16
CSIG	3.35	3.23	3.48
CBAK	2.44	2.68	2.94
COVL	2.63	2.67	2.80
SSNR	1.68	5.07	7.73

Table 2: Subjective evaluation results.

Metric	Noisy	Wiener	SEGAN
MOS	2.09	2.70	3.18

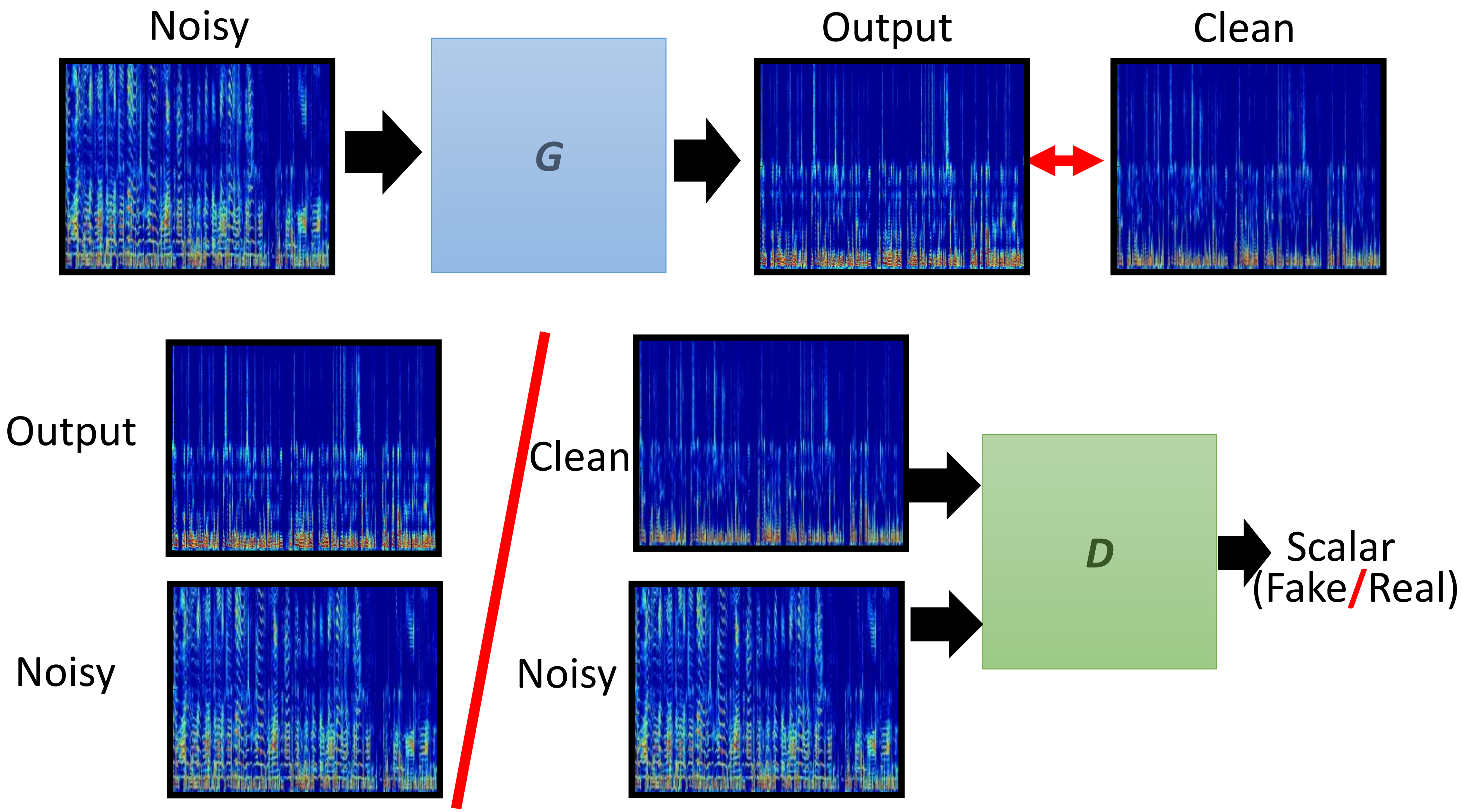
Fig. 1: Preference test results.



SEGAN yields better speech enhancement results than Noisy and Wiener.

Speech Enhancement

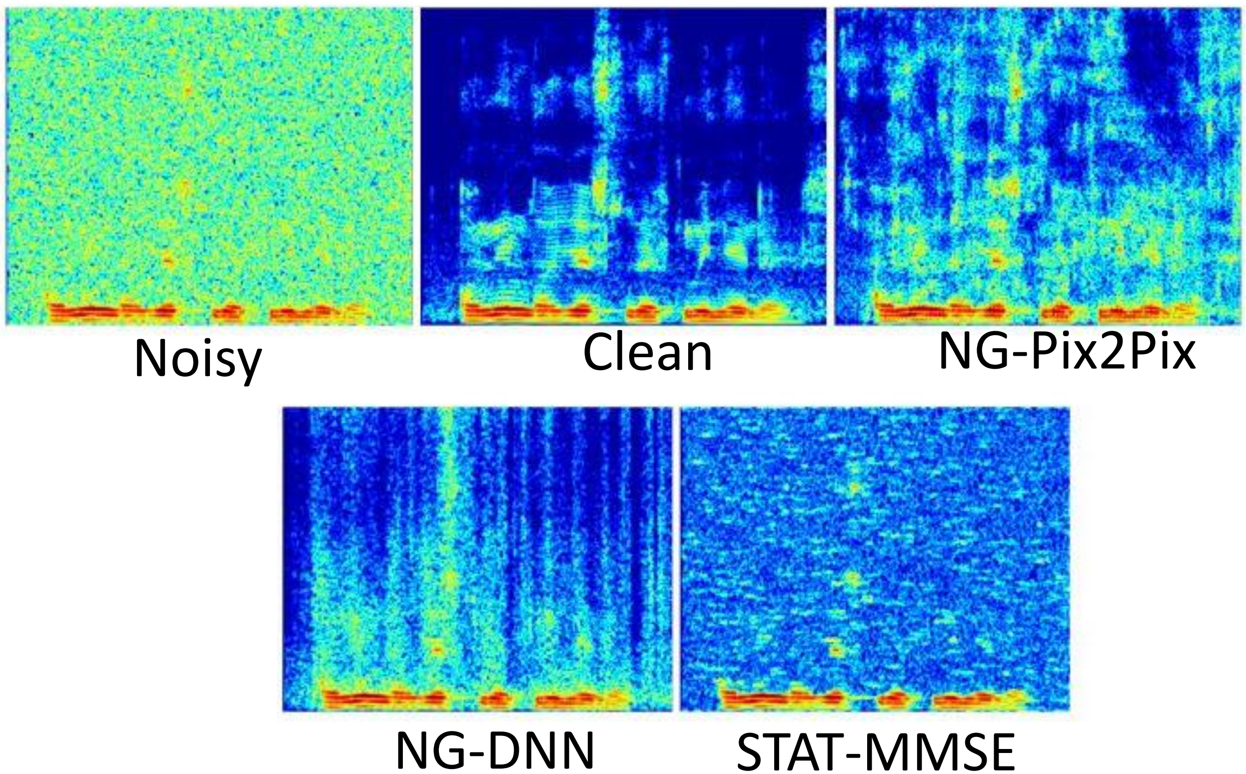
- Pix2Pix [Michelsanti et al., Interpsech 2017]



Speech Enhancement (Pix2Pix)

- Spectrogram analysis

Fig. 2: Spectrogram comparison of Pix2Pix with baseline methods.



Pix2Pix outperforms STAT-MMSE and is competitive to DNN SE.

Speech Enhancement (Pix2Pix)

- Objective evaluation and speaker verification test

Table 3: Objective evaluation results.

	SNR	PESQ 0	5	10	15	20	mean
Babble	(a)	1.20	1.42	1.79	2.40	**3.13**	1.99
	(b)	1.14	1.31	1.61	2.07	2.65	1.76
	(c)	**1.25**	1.51	1.87	2.31	2.78	1.95
	(d)	1.20	1.48	1.98	2.52	2.93	2.02
	(e)	1.24	**1.52**	1.88	2.31	2.78	1.95
	(f)	1.20	1.49	**2.00**	**2.53**	2.93	**2.03**

		STOI 0	5	10	15	20	mean
Babble	(a)	0.44	0.56	0.67	0.77	0.85	0.66
	(b)	0.43	0.56	0.66	0.74	0.81	0.64
	(c)	**0.50**	**0.63**	**0.72**	**0.79**	**0.86**	**0.70**
	(d)	0.46	0.59	0.71	0.78	0.83	0.67
	(e)	0.49	0.62	**0.72**	**0.79**	0.85	**0.70**
	(f)	0.46	0.60	0.71	0.77	0.82	0.67

Table 4: Speaker verification results.

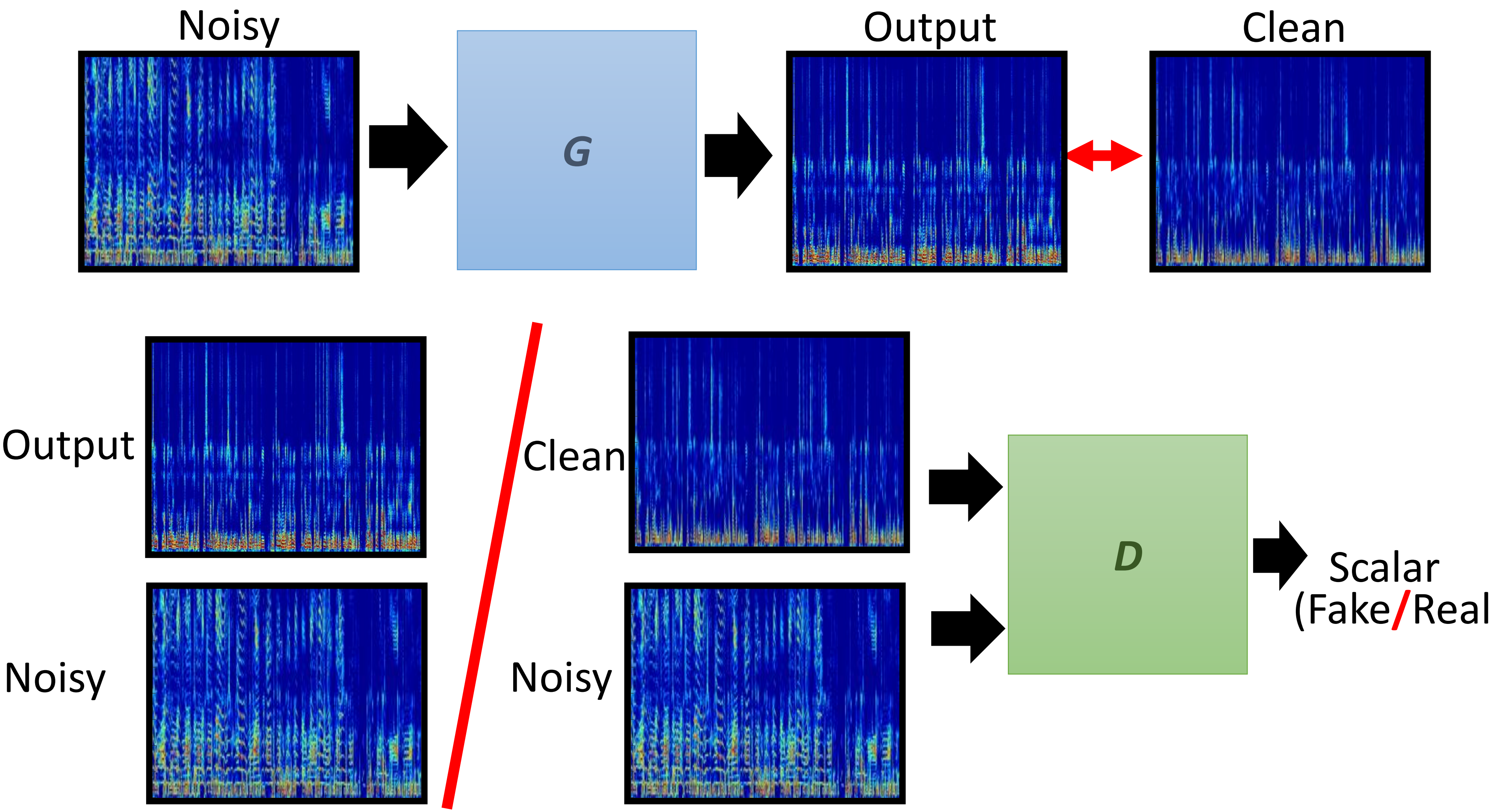
	SNR	0	5	10	15	20	clean	mean
Airplane	(a)	21.09	15.99	13.61	11.66	9.18	6.99	13.08
	(b)	17.69	12.58	8.17	6.53	6.27	5.80	9.51
	(c)	16.99	10.55	7.48	6.99	6.15	6.12	9.05
	(d)	17.19	8.84	**5.44**	5.05	**4.63**	**3.74**	7.48
	(e)	15.99	8.99	6.12	6.12	5.58	5.67	8.08
	(f)	**15.31**	**7.89**	5.58	**4.77**	4.76	5.44	**7.29**

(a)	No enhancement
(b)	STSA-MMSE
(c)	NS-DNN
(d)	**NS-Pix2Pix**
(e)	NG-DNN
(f)	**NG-Pix2Pix**

1. From the objective evaluations, Pix2Pix outperforms Noisy and MMSE and is competitive to DNN SE.
2. From the speaker verification results, Pix2Pix outperforms the baseline models when the clean training data is used.

Speech Enhancement

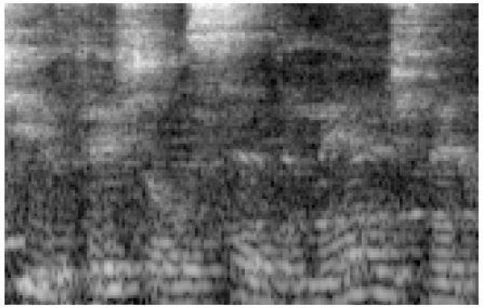
- Frequency-domain SEGAN (FSEGAN) [Donahue et al., ICASSP 2018]



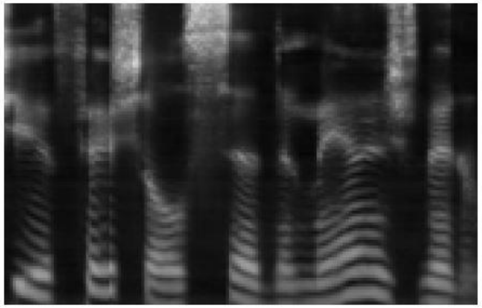
Speech Enhancement (FSEGAN)

- Spectrogram analysis

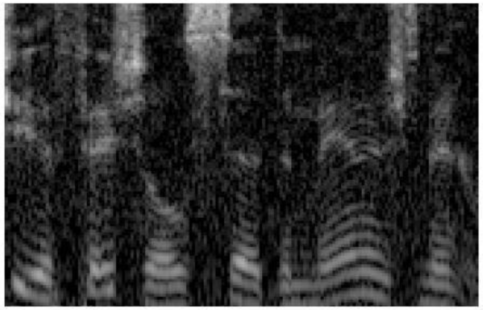
Fig. 2: Spectrogram comparison of FSEGAN with L1-trained method.



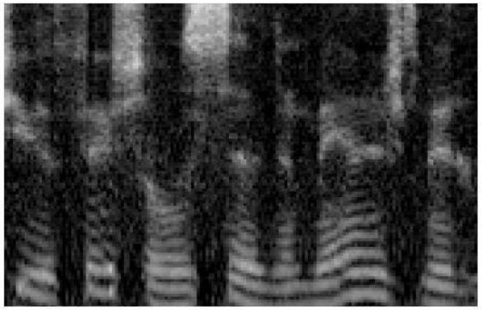
(a) Noisy speech input $\boldsymbol{x}$



(b) L1-trained output $G(\boldsymbol{x})$



(c) Clean speech target $\boldsymbol{y}$



(d) FSEGAN output $G(\boldsymbol{x})$

FSEGAN reduces both additive noise and reverberant smearing.

Speech Enhancement (FSEGAN)

- ASR results

Table 5: WER (%) of SEGAN and FSEGAN.

Test Set	Enhancer	ASR-Clean WER	ASR-MTR WER
Clean	None	11.9	14.3
MTR	None	72.2	20.3
	SEGAN	80.7	52.8
	FSEGAN	33.3	25.4

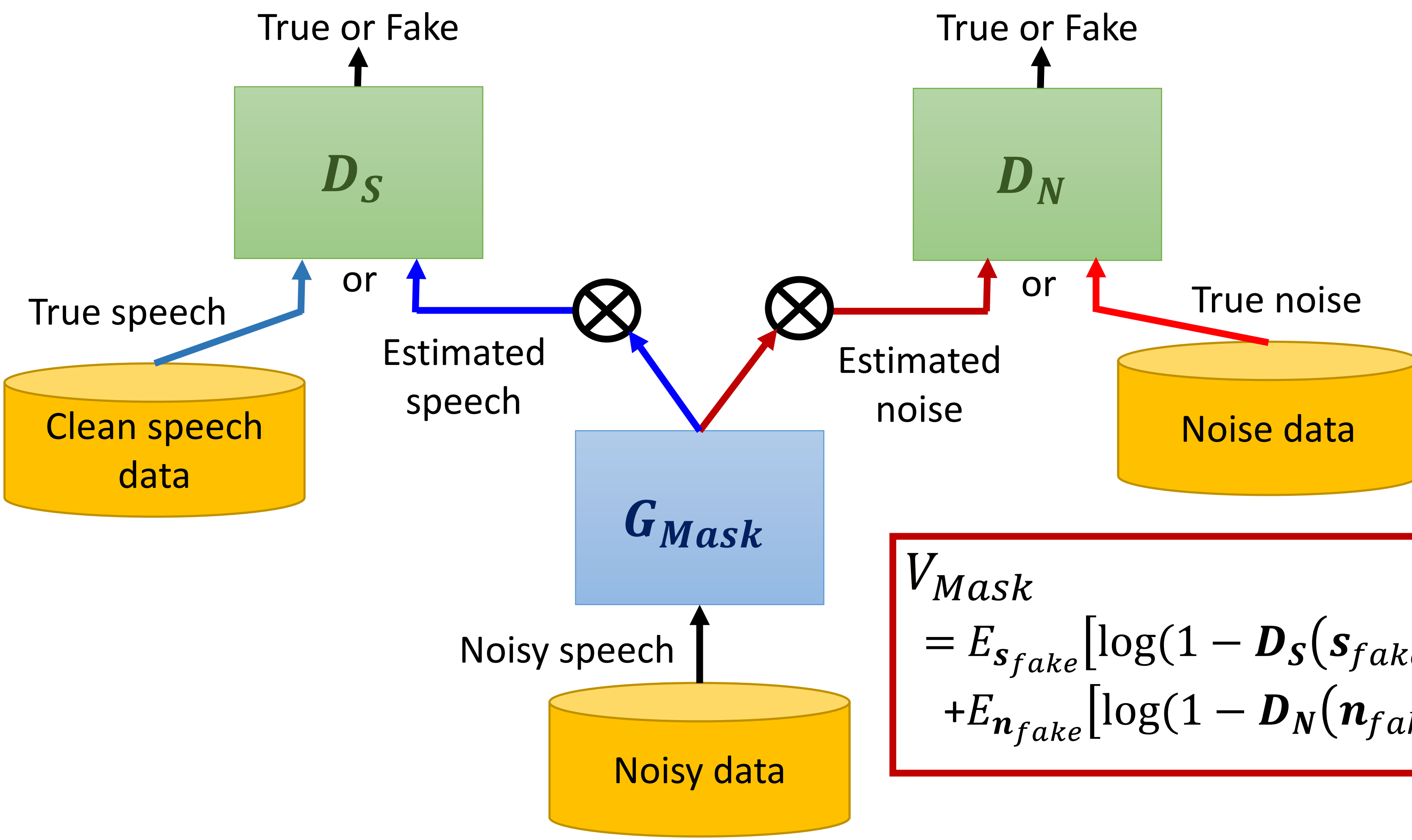
Table 6: WER (%) of FSEGAN with retrain.

Model	WER (%)
MTR Baseline *	20.3
+ Stereo	19.0
MTR + FSEGAN Enhancer *	25.4
+ Retraining	21.0
+ Hybrid Retraining	17.6
MTR + L1-trained Enhancer *	21.4
+ Retraining	18.0
+ Hybrid Retraining	17.1

1. From Table 5, (1) FSEGAN improves recognition results for ASR-Clean.
   (2) FSEGAN outperforms SEGAN as front-ends.
2. From Table 6, (1) Hybrid Retraining with FSEGAN outperforms Baseline;
   (2) FSEGAN retraining slightly underperforms L1–based retraining.

Speech Enhancement

- Adversarial training based mask estimation (ATME)
  [Higuchi et al., ASRU 2017]

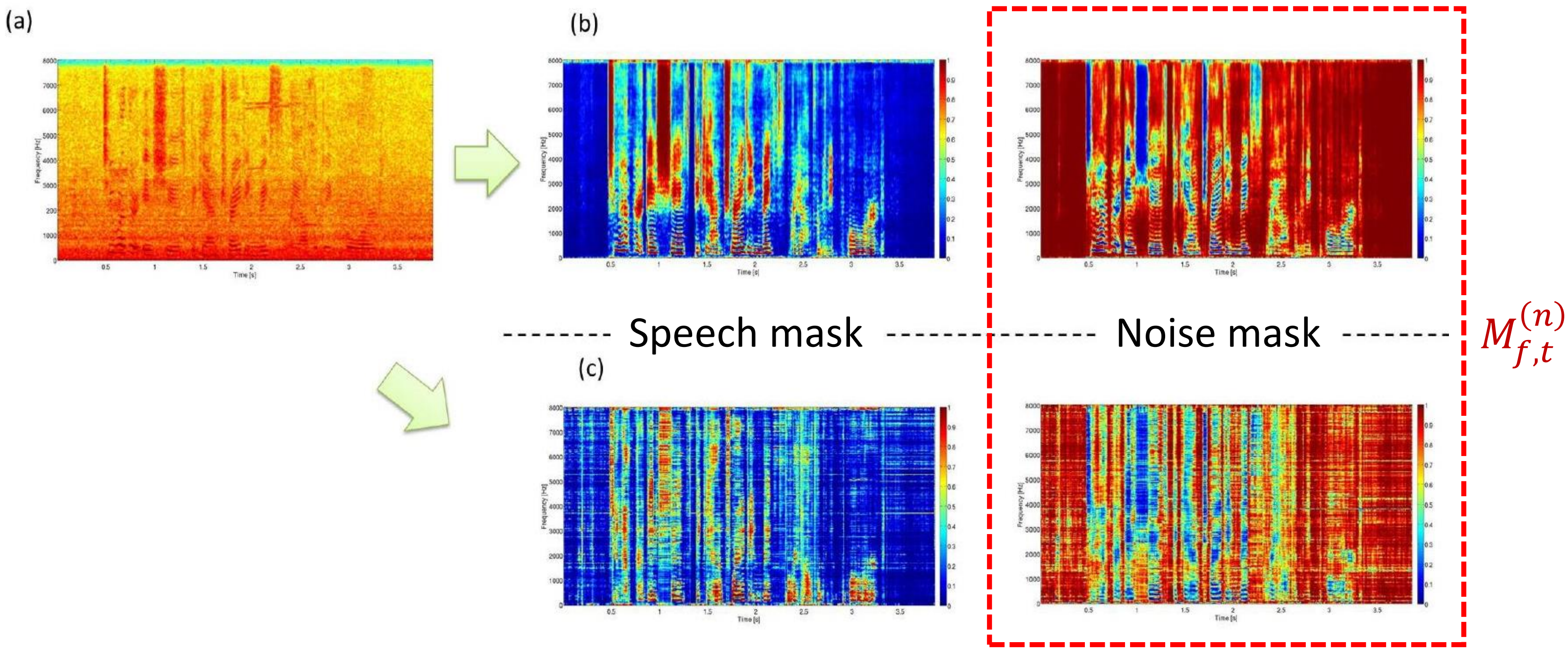


$$V_{Mask} = E_{\boldsymbol{s}_{fake}}\left[\log\left(1 - \boldsymbol{D_S}\left(\boldsymbol{s}_{fake}, \theta\right)\right)\right] + E_{\boldsymbol{n}_{fake}}\left[\log\left(1 - \boldsymbol{D_N}\left(\boldsymbol{n}_{fake}, \theta\right)\right)\right]$$

Speech Enhancement (ATME)

- Spectrogram analysis

Fig. 3: Spectrogram comparison of (a) noisy; (b) MMSE with supervision; (c) ATMB without supervision.



The proposed adversarial training mask estimation can capture speech/noise signals without supervised data.

Speech Enhancement (ATME)

- Mask-based beamformer for robust ASR

➢ The estimated mask parameters are used to compute spatial covariance matrix for MVDR beamformer.

➢ $\hat{s}_{f,t} = \boxed{\mathbf{w}_f^{\mathrm{H}}}\mathbf{y}_{f,t}$, where $\hat{s}_{f,t}$ is the enhanced signal, and $\mathbf{y}_{f,t}$ denotes the observation of M microphones, f and t are frequency and time indices; $\mathbf{w}_f$ denotes the beamformer coefficient.

➢ The MVDR solves $\mathbf{w}_f$ by: $\mathbf{w}_f = \dfrac{(R_f^{(s+n)})^{-1}\boxed{\mathbf{h}_f}}{\boxed{\mathbf{h}_f^{\mathrm{H}}}(R_f^{(s+n)})^{-1}\boxed{\mathbf{h}_f}}$

➢ To estimate $\mathbf{h}_f$, the spatial covariance matrix of the target signal, $\boxed{R_f^{(s)}}$, is computed by : $\boxed{R_f^{(s)} = R_f^{(s+n)} - R_f^{(n)}}$, where $\boxed{R_f^{(n)} = \dfrac{M_{f,t}^{(n)}\,\mathbf{y}_{f,t}\,\mathbf{y}_{f,t}^{\mathrm{H}}}{\sum_{f,t} M_{f,t}^{(n)}}, \boxed{M_{f,t}^{(n)}}}$ was computed by AT.

Speech Enhancement (ATME)

- ASR results

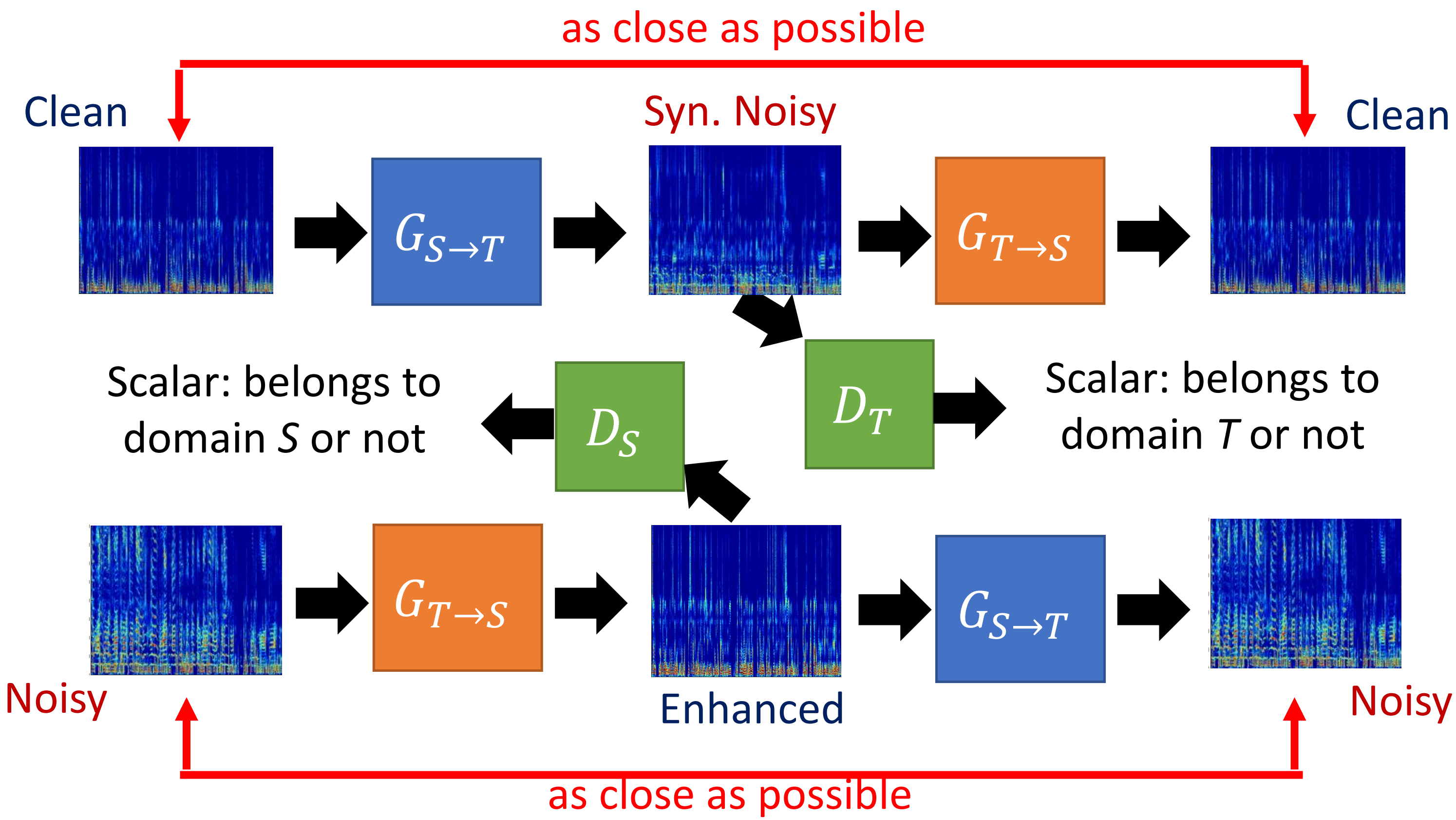
Table 7: WERs (%) for the development and evaluation sets.

systems	dev					eval				
	avg	bus	caf	ped	str	avg	bus	caf	ped	str
Unprocessed	9.01	14.00	7.94	6.03	8.05	15.60	22.55	16.21	12.89	10.74
Adversarial Training	5.00	7.60	4.09	4.03	4.29	7.58	10.24	7.51	6.20	6.39
MMSE	4.83	7.20	4.04	3.98	4.10	7.04	9.25	6.67	6.02	6.24

1. ATME provides significant improvements over Unprocessed.
2. Unsupervised ATME slightly underperforms supervised MMSE.

Speech Enhancement (AFT)

- Cycle-GAN-based acoustic feature transformation (AFT) [Mimura et al., ASRU 2017]



$$V_{Full} = V_{GAN}(G_{X\to Y}, D_Y) + V_{GAN}(G_{X\to Y}, D_Y) \\ + \lambda\, V_{Cyc}(G_{X\to Y}, G_{Y\to X})$$

Speech Enhancement (AFT)

- ASR results on noise robustness and style adaptation

Table 8: Noise robust ASR.

acoustic model	feature	cycle loss	λ and μ	WER	ID
no adapt.	no adapt.	-	-	41.08	(1)
no adapt.	adapt. with $G_{T\to S}$	no	1, 1	55.45	(2)
		yes	1, 1	37.34	(3)
		yes	trained	36.56	(4)
adapt. with $G_{S\to T}$	no adapt.	yes	1, 1	35.98	(5)
		yes	trained	34.31	(6)

S: Clean; T: Noisy

Table 9: Speaker style adaptation.

source	target	feature	WER
JNAS	CSJ-SPS	no adapt.	26.47
		adapt. with $G_{T\to S}$	25.93
CSJ-APS	CSJ-SPS	no adapt.	17.15
		adapt. with $G_{T\to S}$	16.60

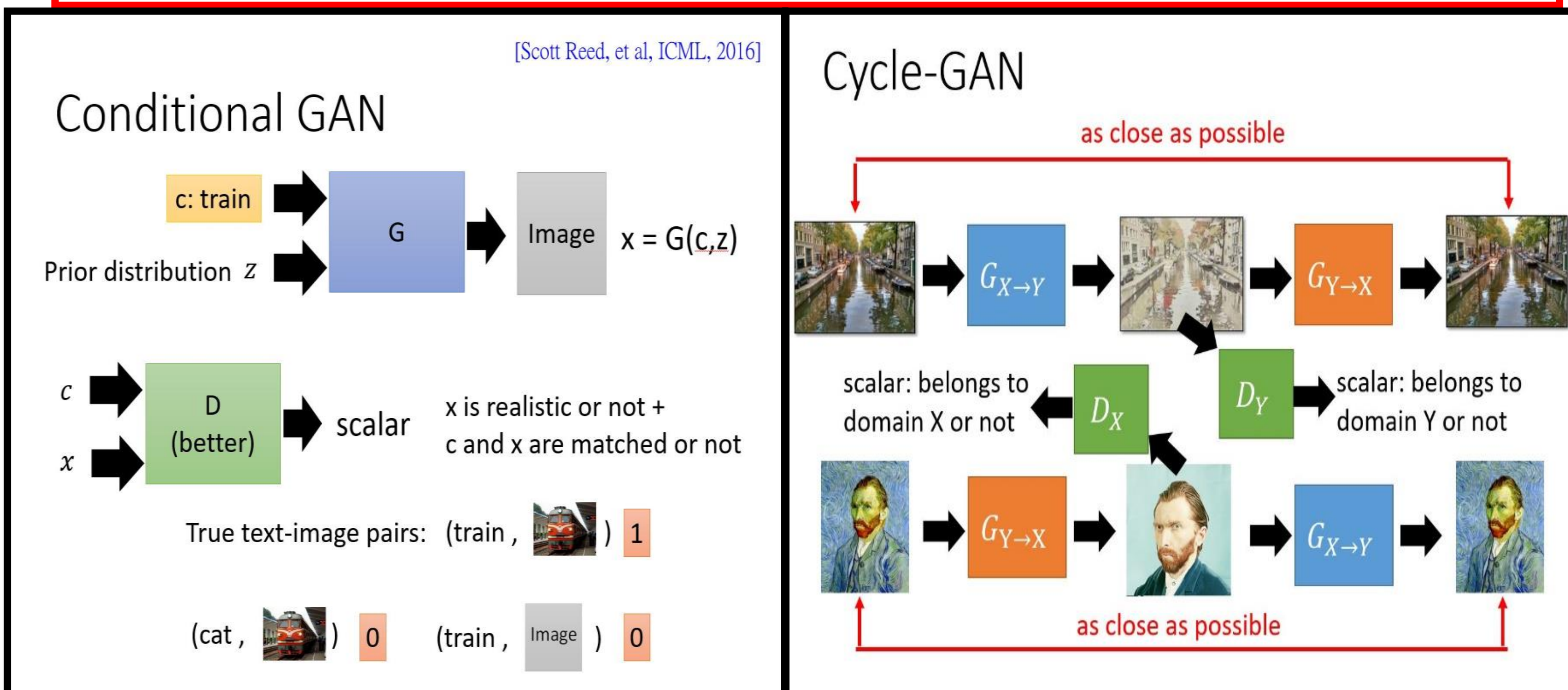
JNAS: Read; CSJ-SPS: Spontaneous (relax);
CSJ-APS: Spontaneous (formal);

1. $G_{T\to S}$ can transform acoustic features and effectively improve ASR results for both noisy and accented speech.
2. $G_{S\to T}$ can be used for model adaptation and effectively improve ASR results for noisy speech.

Outline of Part II

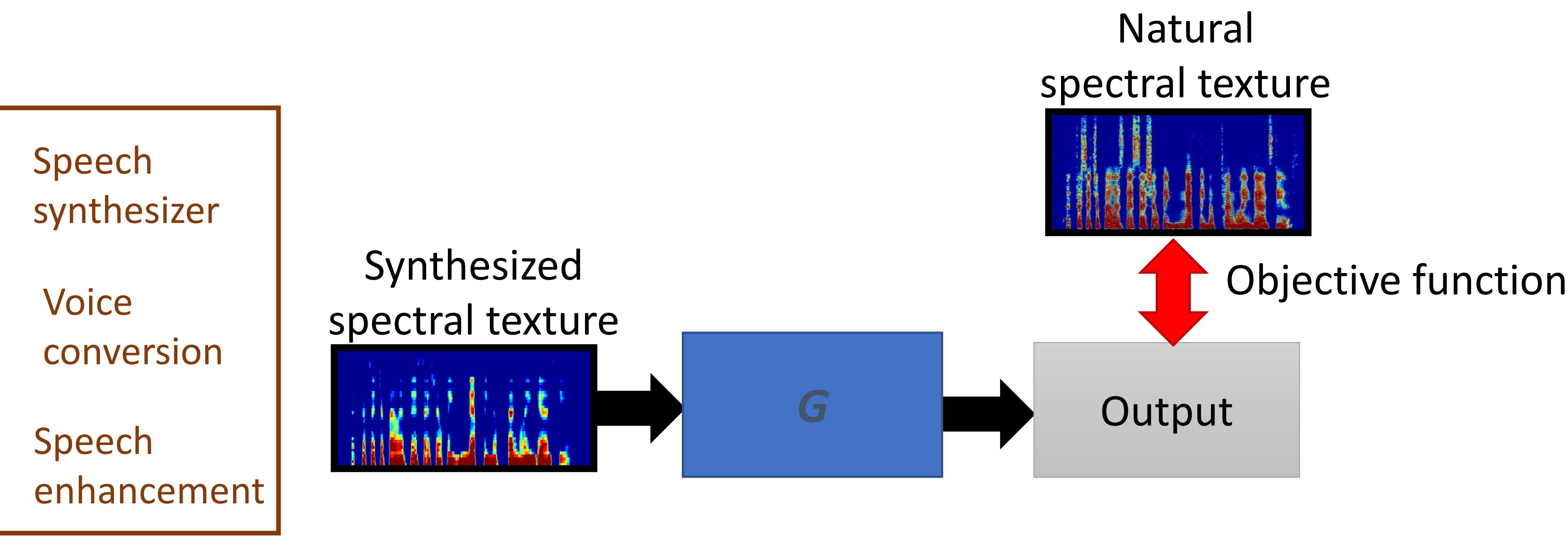
Speech Signal Generation

- Speech enhancement
- Postfilter, speech synthesis, voice conversion



Postfilter

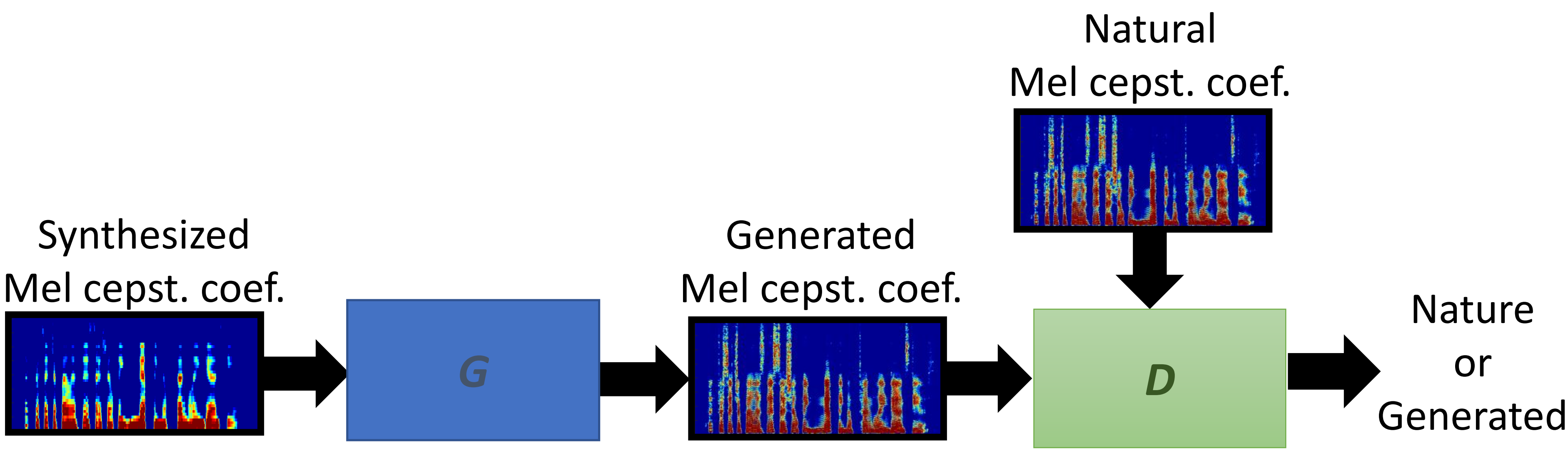
- Postfilter for synthesized or transformed speech



- ➢ Conventional postfilter approaches for ***G*** estimation include global variance (GV) [Toda et al., IEICE 2007], variance scaling (VS) [Sil’en et al., Interpseech 2012], modulation spectrum (MS) [Takamichi et al., ICASSP 2014],DNN with MSE criterion [Chen et al., Interspeech 2014; Chen et al., TASLP 2015].
- ➢ GAN is used a new objective function to estimate the parameters in ***G***.

Postfilter

- GAN postfilter [Kaneko et al., ICASSP 2017]

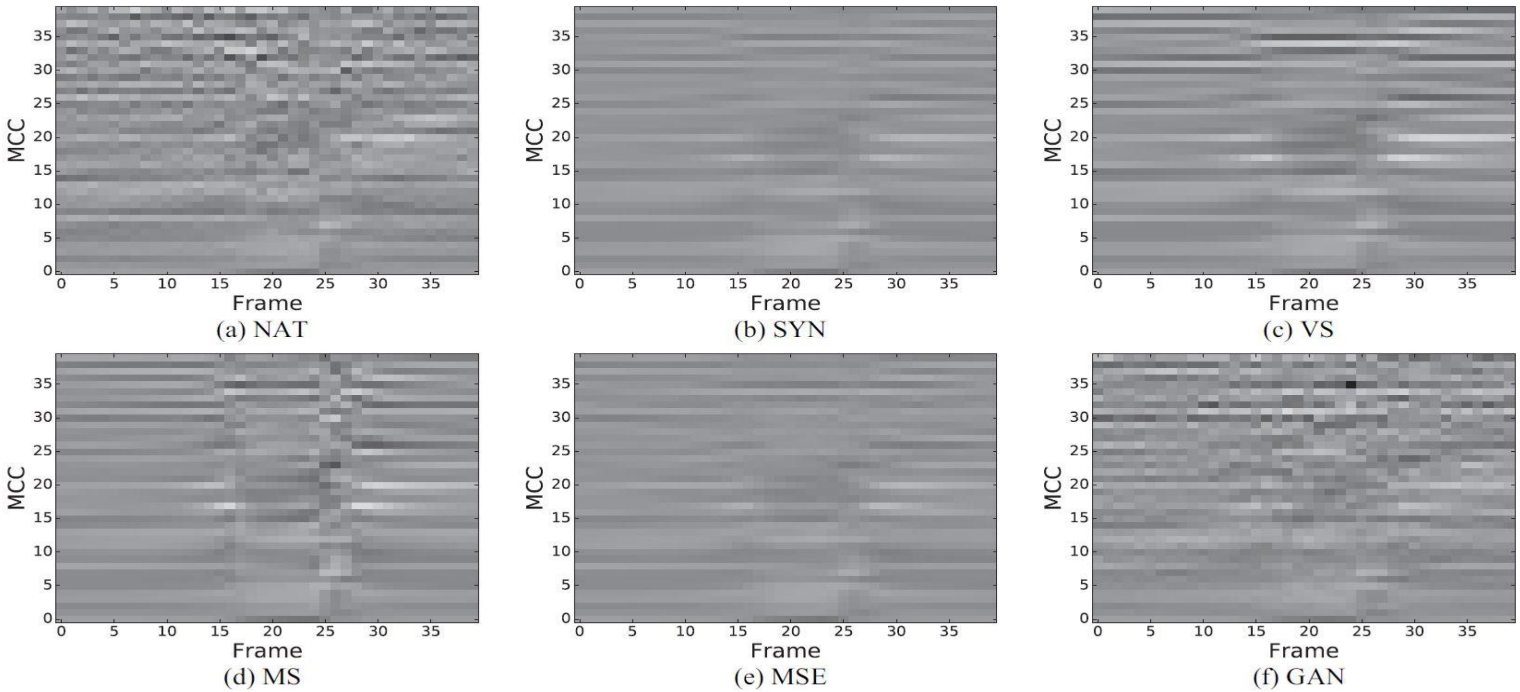


> ➢ Traditional MMSE criterion results in statistical averaging.
> ➢ GAN is used as a new objective function to estimate the parameters in ***G***.
> ➢ The proposed work intends to further improve the naturalness of synthesized speech or parameters from a synthesizer.

Postfilter (GAN-based Postfilter)

- Spectrogram analysis

Fig. 4: Spectrograms of: (a) NAT (nature); (b) SYN (synthesized); (c) VS (variance scaling); (d) MS (modulation spectrum); (e) MSE; (f) GAN postfilters.



GAN postfilter reconstructs spectral texture similar to the natural one.

Postfilter (GAN-based Postfilter)

- Objective evaluations

Fig. 5: Mel-cepstral trajectories (GANv: GAN was applied in voiced part).

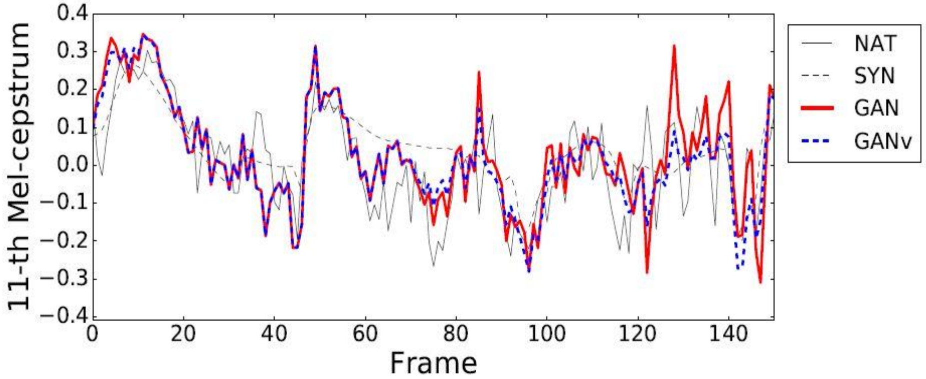
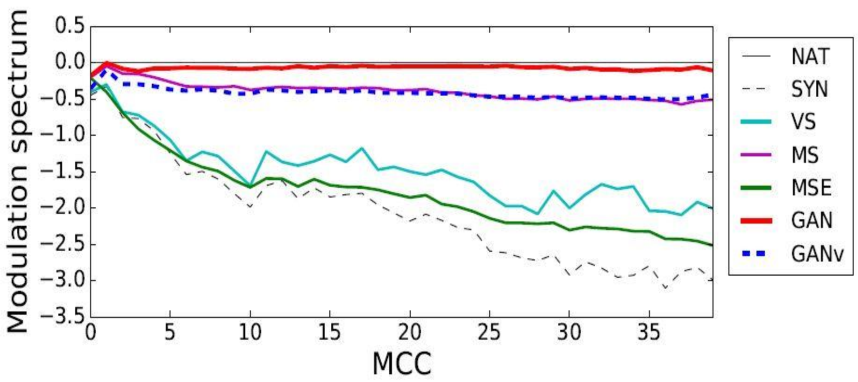


Fig. 6: Averaging difference in modulation spectrum per Mel-cepstral coefficient.



GAN postfilter reconstructs spectral texture similar to the natural one.

Postfilter (GAN-based Postfilter)

- Subjective evaluations

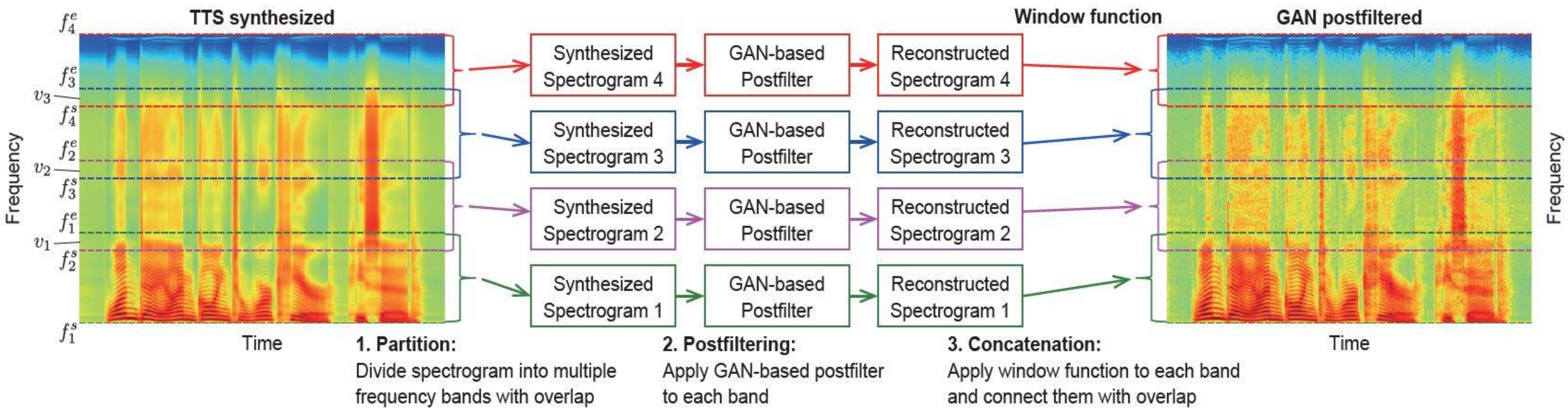
Table 10: Preference score (%). Bold font indicates the numbers over 30%.

	Former	Latter	Neutral
GAN vs. SYN	**56.5** ± 4.9	22.0 ± 4.1	21.5 ± 4.0
GAN vs. GANv	11.3 ± 3.1	**37.3** ± 4.8	**51.5** ± 4.9
GAN vs. NAT	16.8 ± 3.7	**53.5** ± 4.9	29.8 ± 4.5
GANv vs. NAT	**30.3** ± 4.5	**34.5** ± 4.7	**35.3** ± 4.7

1. GAN postfilter significantly improves the synthesized speech.
2. GAN postfilter is effective particularly in voiced segments.
3. GANv outperforms GAN and is comparable to NAT.

Postfilter (GAN-postfilter-SFTF)

- GAN post-filter for STFT spectrograms [Kaneko et al., Interspeech 2017]

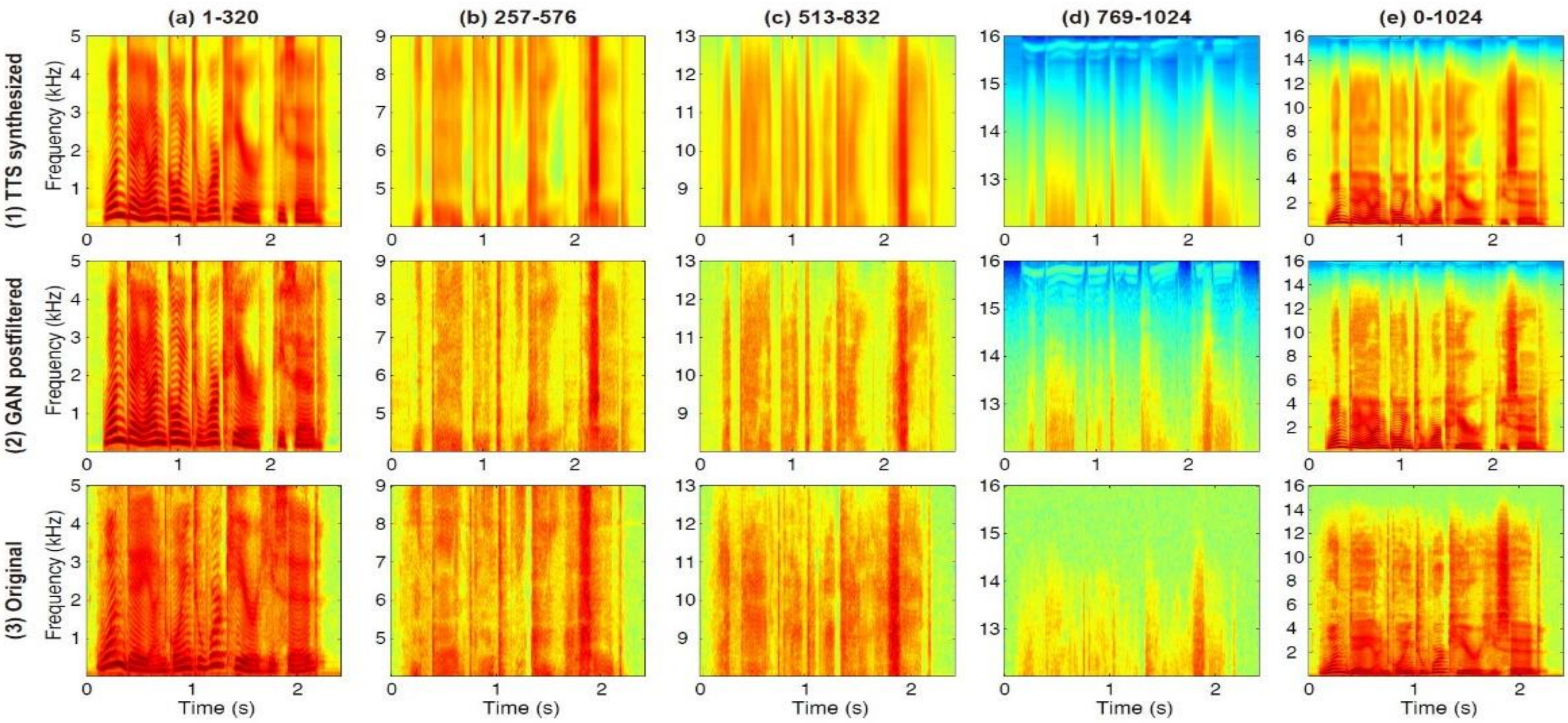


- ➢ GAN postfilter was applied on high-dimensional STFT spectrograms.
- ➢ The spectrogram was partitioned into *N* bands (each band overlaps its neighboring bands).
- ➢ The GAN-based postfilter was trained for each band.
- ➢ The reconstructed spectrogram from each band was smoothly connected.

Postfilter (GAN-postfilter-SFTF)

- Spectrogram analysis

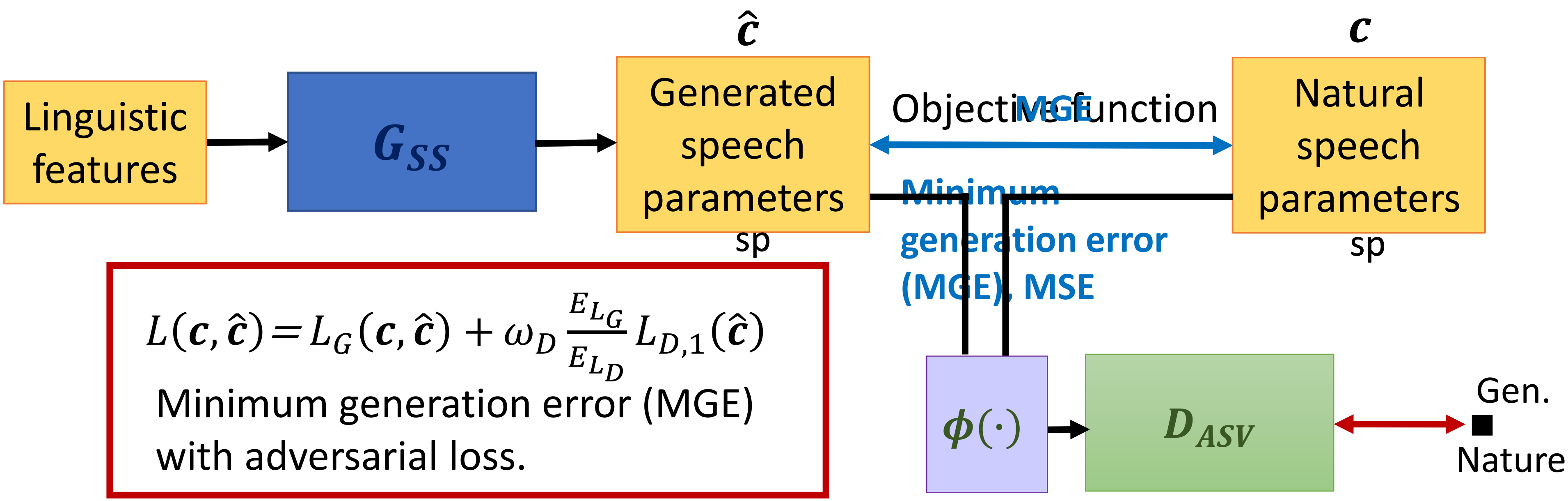
Fig. 7: Spectrograms of: (1) SYN, (2) GAN, (3) Original (NAT)



GAN postfilter reconstructs spectral texture similar to the natural one.

Speech Synthesis

- Speech synthesis with anti-spoofing verification (ASV)
  Input: linguistic features; Output: speech parameters
  [Saito et al., ICASSP 2017]



$$L(\boldsymbol{c},\hat{\boldsymbol{c}}) = L_G(\boldsymbol{c},\hat{\boldsymbol{c}}) + \omega_D \frac{E_{L_G}}{E_{L_D}} L_{D,1}(\hat{\boldsymbol{c}})$$

Minimum generation error (MGE) with adversarial loss.

$$L_D(\boldsymbol{c},\hat{\boldsymbol{c}}) = L_{D,1}(\boldsymbol{c}) + L_{D,0}(\hat{\boldsymbol{c}})$$

$$L_{D,1}(\boldsymbol{c}) = -\frac{1}{T}\sum_{t=1}^{T} \log(D(\boldsymbol{c}_t))\text{...NAT}$$

$$L_{D,0}(\hat{\boldsymbol{c}}) = -\frac{1}{T}\sum_{t=1}^{T} \log(1 - D(\hat{\boldsymbol{c}}_t))\text{...SYN}$$

Speech Synthesis (ASV)

- Objective and subjective evaluations

Fig. 8: Averaged GVs of MCCs.

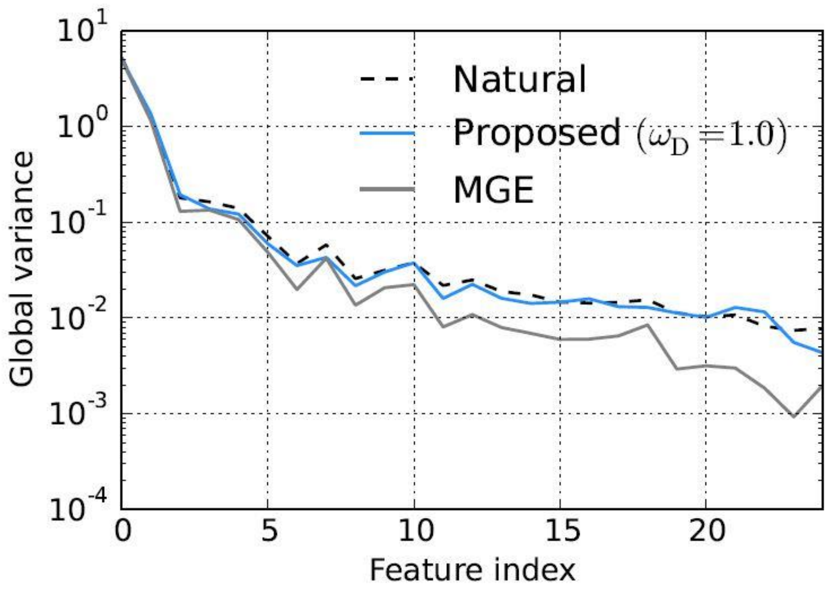
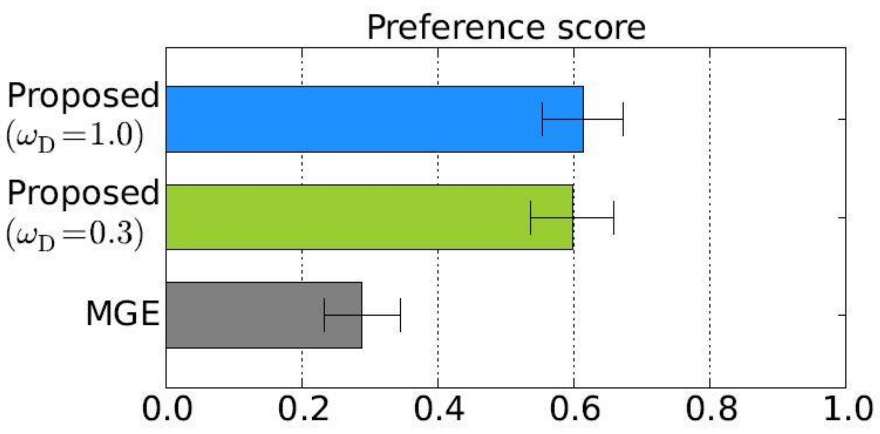


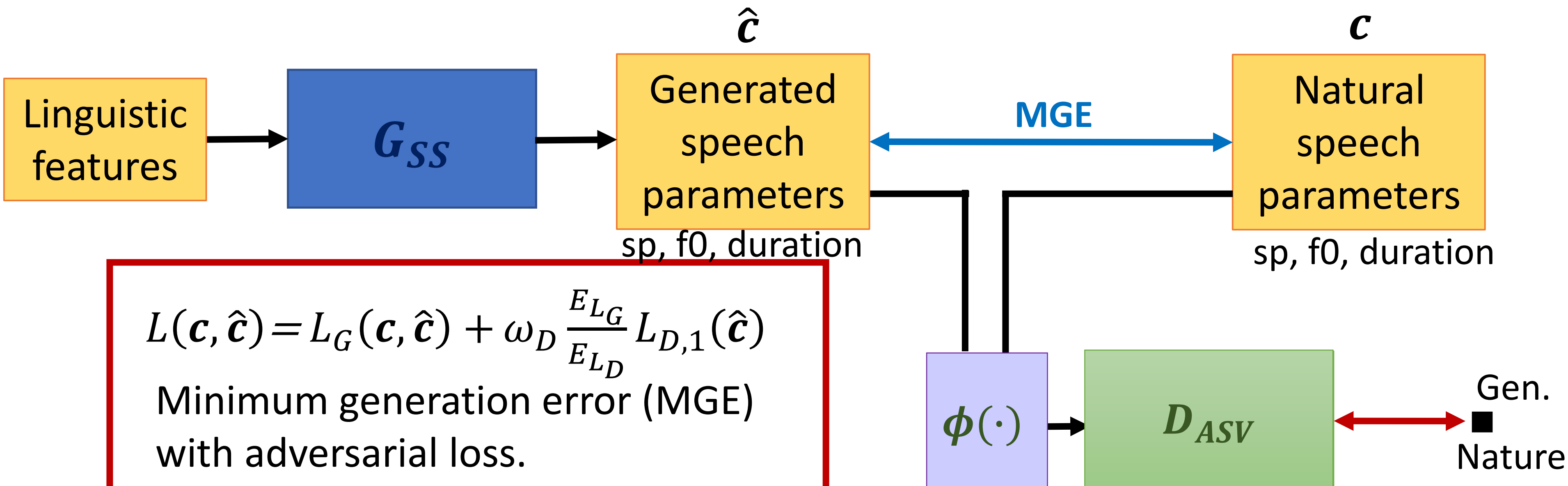
Fig. 9: Scores of speech quality.



1. The proposed algorithm generates MCCs similar to the natural ones.
2. The proposed algorithm outperforms conventional MGE training.

Speech Synthesis

- Speech synthesis with GAN (SS-GAN) [Saito et al., TASLP 2018]



$$L(\boldsymbol{c},\hat{\boldsymbol{c}}) = L_G(\boldsymbol{c},\hat{\boldsymbol{c}}) + \omega_D \frac{E_{L_G}}{E_{L_D}} L_{D,1}(\hat{\boldsymbol{c}})$$

Minimum generation error (MGE) with adversarial loss.

$$L_D(\boldsymbol{c},\hat{\boldsymbol{c}}) = L_{D,1}(\boldsymbol{c}) + L_{D,0}(\hat{\boldsymbol{c}})$$

$L_{D,1}(\boldsymbol{c}) = -\frac{1}{T}\sum_{t=1}^{T} \log(D(\boldsymbol{c}_t))$…NAT

$L_{D,0}(\hat{\boldsymbol{c}}) = -\frac{1}{T}\sum_{t=1}^{T} \log(1 - D(\hat{\boldsymbol{c}}_t))$…SYN

Speech Synthesis (SS-GAN)

- Subjective evaluations

Fig. 10: Scores of speech quality (sp).

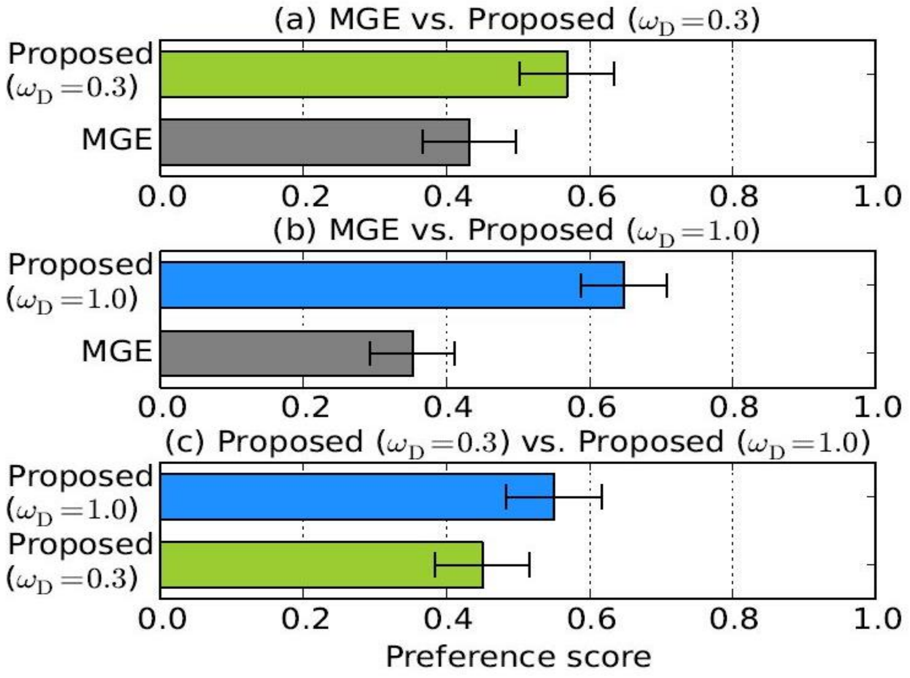
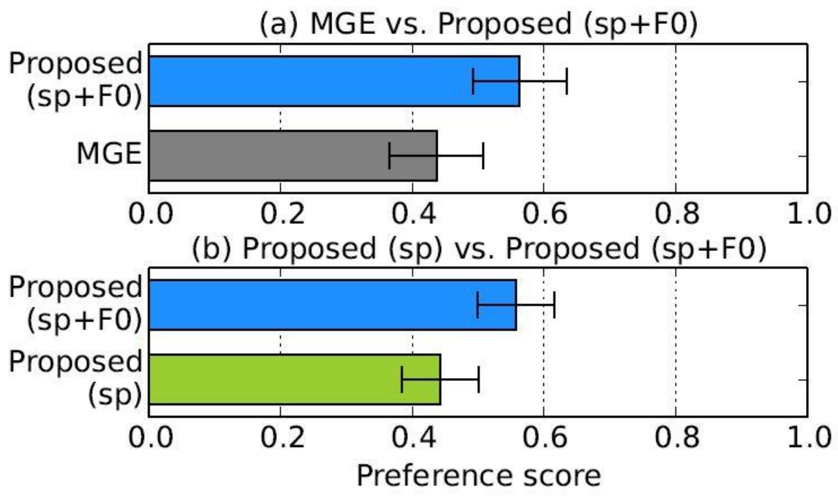


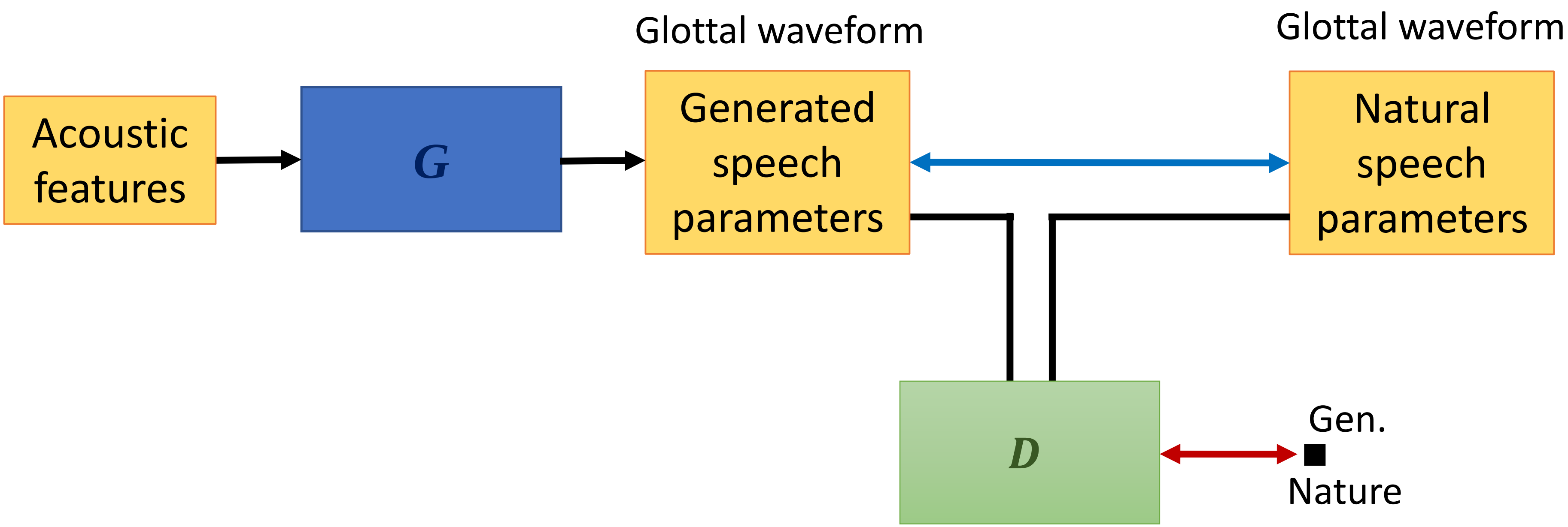
Fig. 11: Scores of speech quality (sp and F0).



The proposed algorithm works for both spectral parameters and F0.

Speech Synthesis

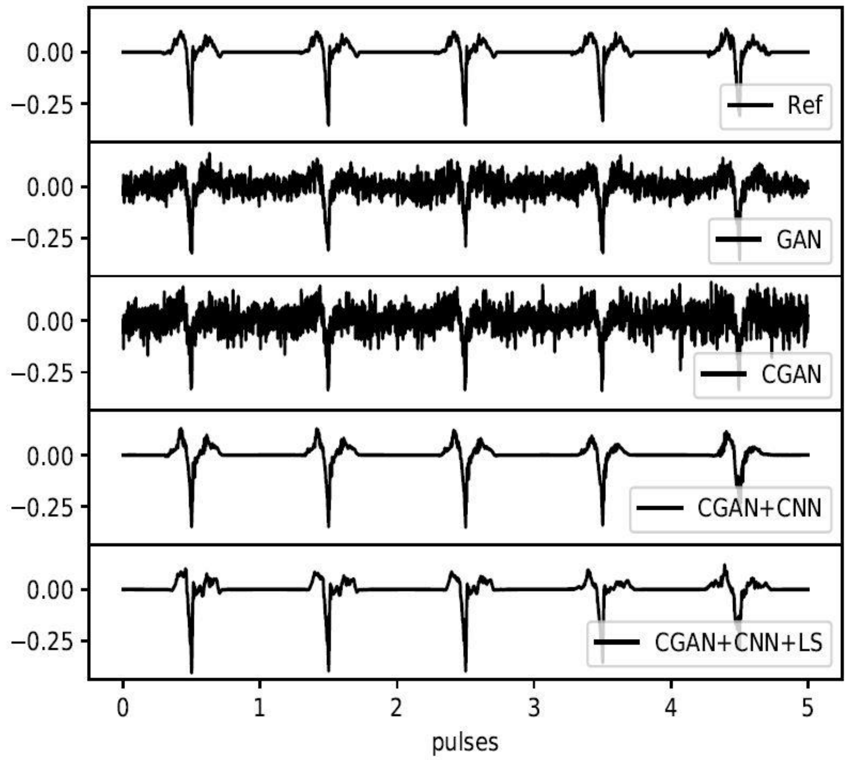
- Speech synthesis with GAN glottal waveform model (GlottGAN) [Bollepalli et al., Interspeech 2017]



Speech Synthesis (GlottGAN)

- Objective evaluations

Fig. 12: Glottal pulses generated by GANs.



G, D: DNN

G, D: conditional DNN

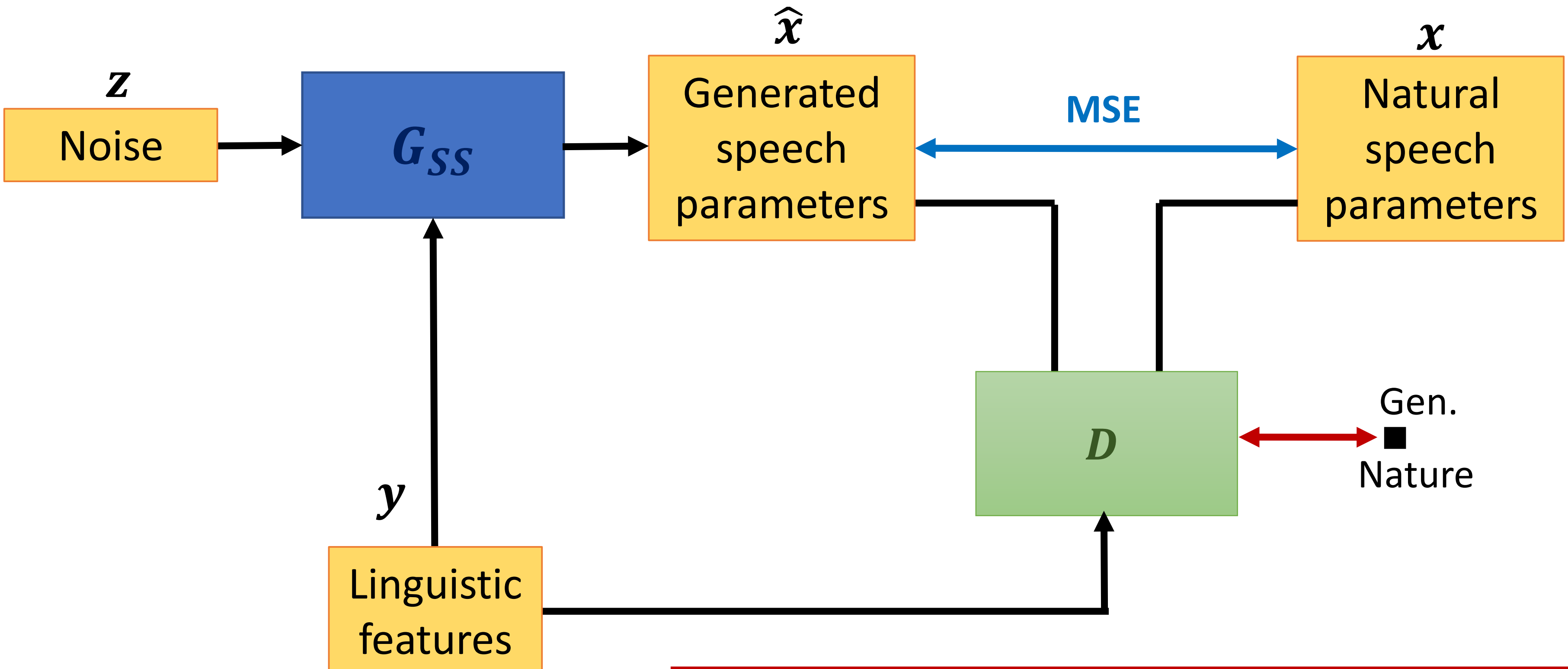
G, D: Deep CNN

G, D: Deep CNN + LS loss

The proposed GAN-based approach can generate glottal waveforms similar to the natural ones.

Speech Synthesis

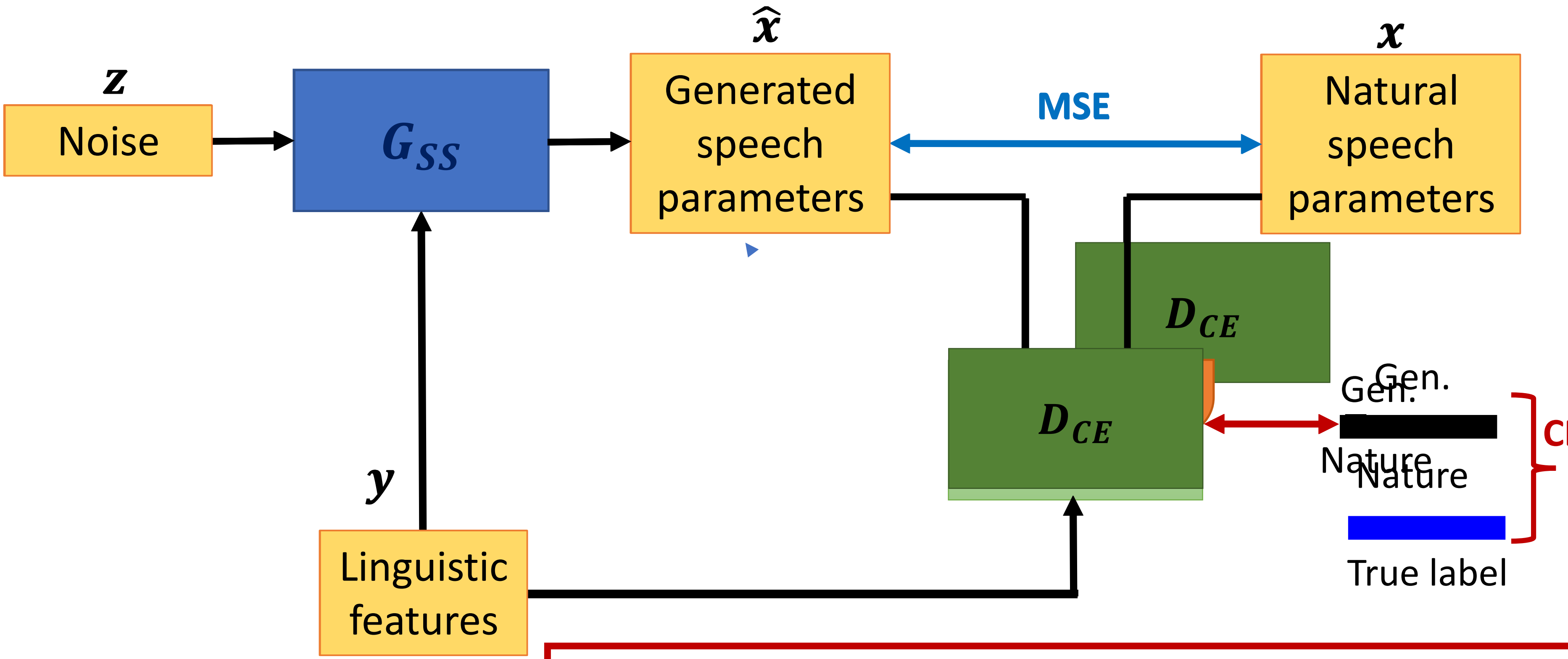
- Speech synthesis with GAN & multi-task learning (SS-GAN-MTL) [Yang et al., ASRU 2017]



$$
\begin{aligned}
V_{GAN}(G,D) = \; & E_{\boldsymbol{x}\sim p_{data}(x)}[log D(\boldsymbol{x}|\boldsymbol{y})] \\
& + E_{\boldsymbol{z}\sim p_{z},}\left[\log(1 - D\big(G(\boldsymbol{z}|\boldsymbol{y})\big)|\boldsymbol{y}\right] \\
V_{L2}(G) = \; & E_{\boldsymbol{z}\sim p_{z},}[G(\boldsymbol{z}|\boldsymbol{y}) - \boldsymbol{x}]^2
\end{aligned}
$$

Speech Synthesis (SS-GAN-MTL)

- Speech synthesis with GAN & multi-task learning (SS-GAN-MTL) [Yang et al., ASRU 2017]



$$V_{GAN}(G,D) = E_{x\sim p_{data}(x)}[\log D_{CE}(\boldsymbol{x}|\boldsymbol{y}, label)] + E_{z\sim p_z,}[\log(1 - D_{CE}(G(\boldsymbol{z}|\boldsymbol{y}))|\boldsymbol{y}, label]$$

$$V_{L2}(G) = E_{z\sim p_z,}[G(\boldsymbol{z}|\boldsymbol{y}) - \boldsymbol{x}]^2$$

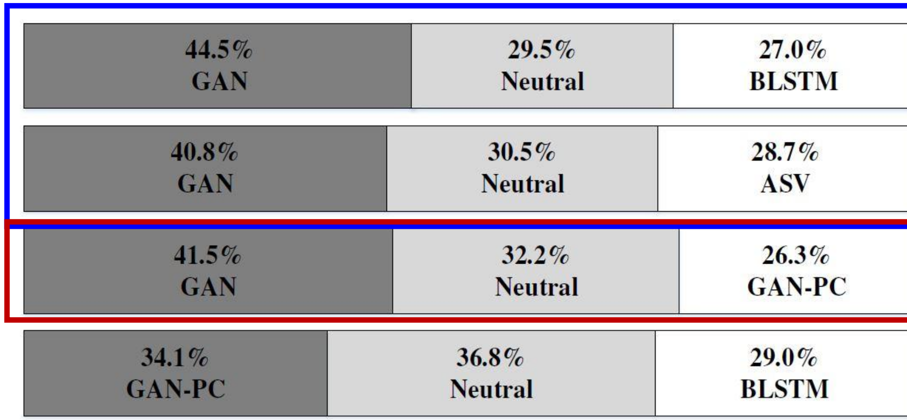
Speech Synthesis (SS-GAN-MTL)

- Objective and subjective evaluations

Table 11: Objective evaluation results.

Methods	MCD (dB)	F_0 RMSE (Hz)	V/UV (%)
BLSTM	4.624	18.544	6.447
ASV [16]	4.670	18.871	6.562
GAN	4.633	18.678	6.492
GAN-PC	4.628	18.616	6.464

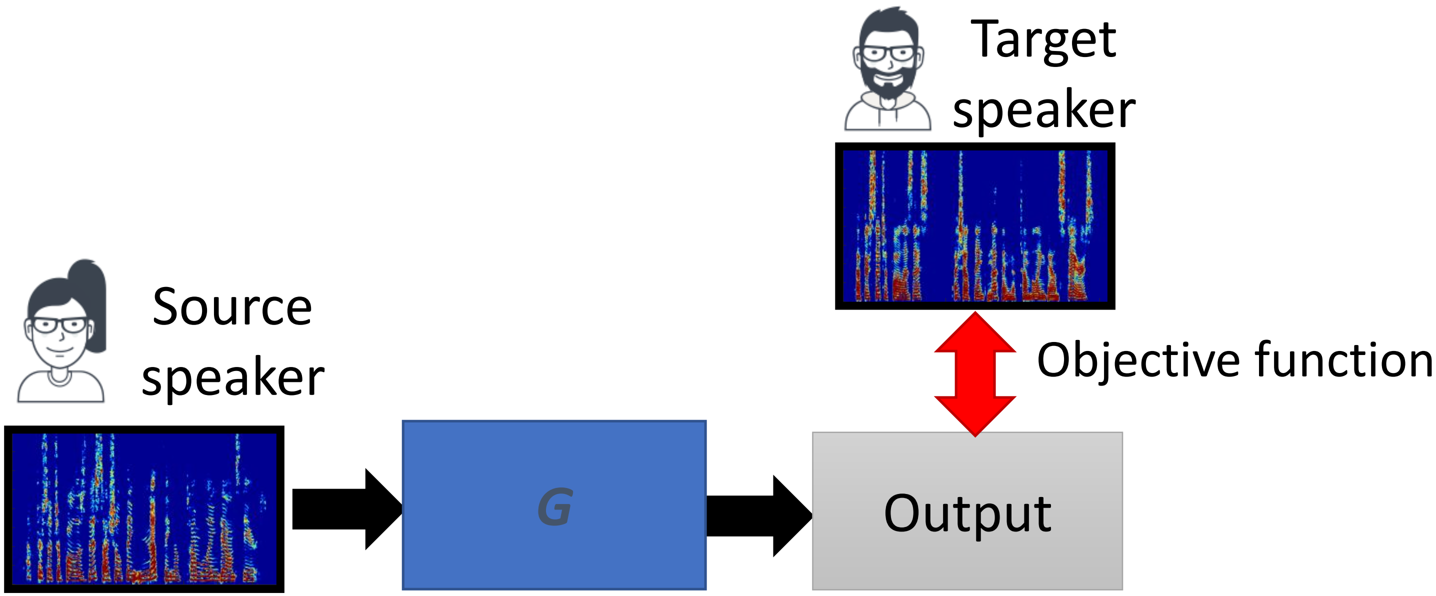
Fig. 13: The preference score (%).



1. From objective evaluations, no remarkable difference is observed.
2. From subjective evaluations, GAN outperforms BLSTM and ASV, while GAN-PC underperforms GAN.

Voice Conversion

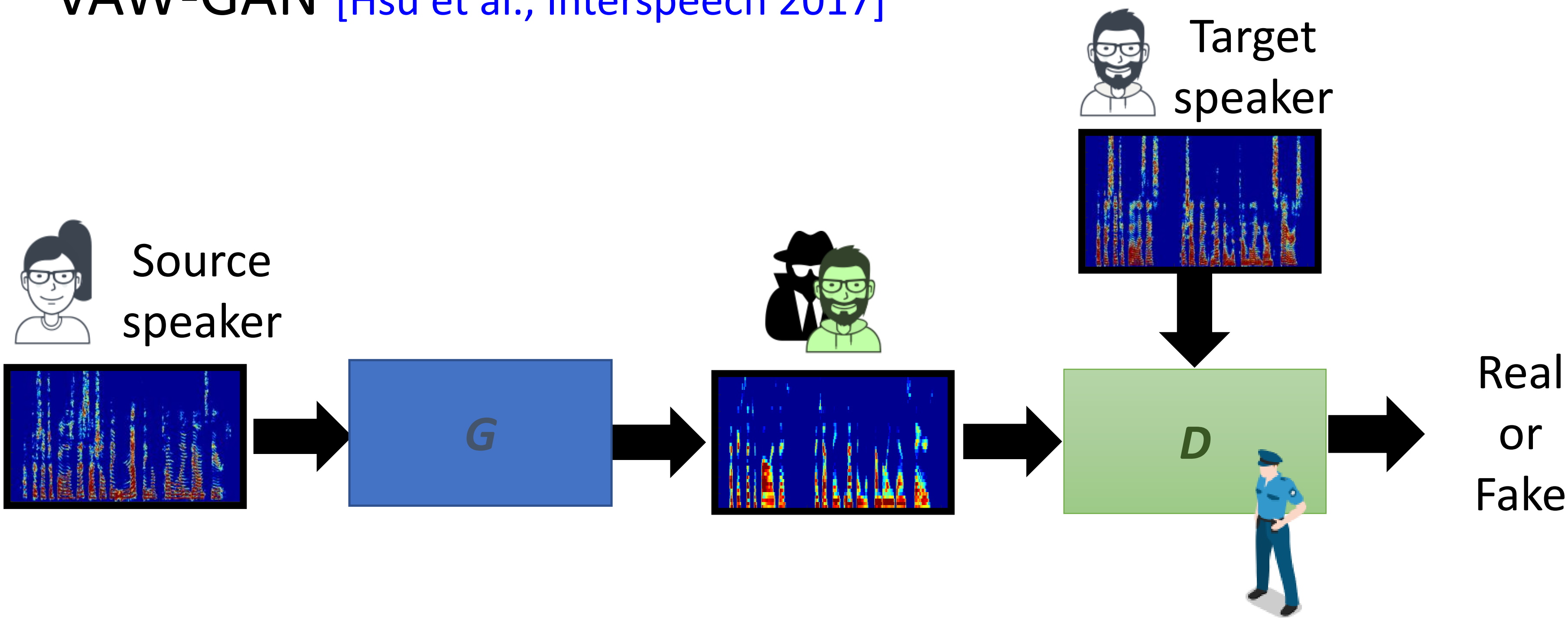
• Convert (transform) speech from source to target



➢ Conventional VC approaches include Gaussian mixture model (GMM) [Toda et al., TASLP 2007], non-negative matrix factorization (NMF) [Wu et al., TASLP 2014; Fu et al., TBME 2017], locally linear embedding (LLE) [Wu et al., Interspeech 2016], restricted Boltzmann machine (RBM) [Chen et al., TASLP 2014], feed forward NN [Desai et al., TASLP 2010], recurrent NN (RNN) [Nakashika et al., Interspeech 2014].

Voice Conversion

- VAW-GAN [Hsu et al., Interspeech 2017]



- ➢ Conventional MMSE approaches often encounter the "over-smoothing" issue.
- ➢ GAN is used a new objective function to estimate ***G***.
- ➢ The goal is to increase the naturalness, clarity, similarity of converted speech.

$$\boxed{V(G,D) = V_{GAN}(G,D) + \lambda V_{VAE}(\boldsymbol{x}|\boldsymbol{y})}$$

Voice Conversion (VAW-GAN)

- Objective and subjective evaluations

Fig. 14: The spectral envelopes.

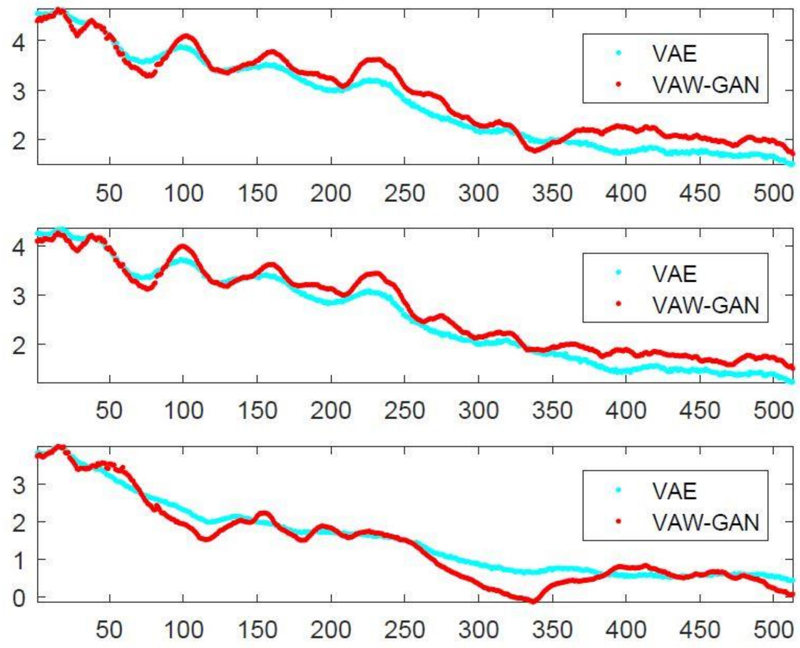
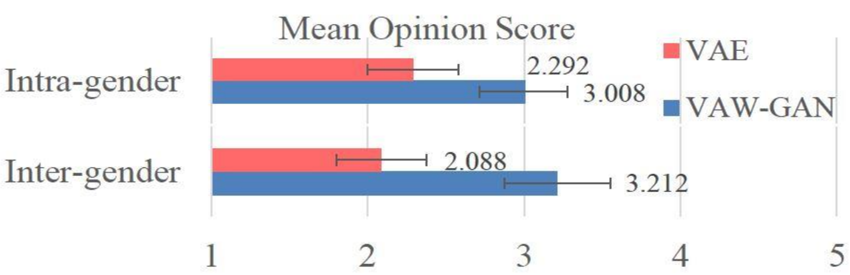


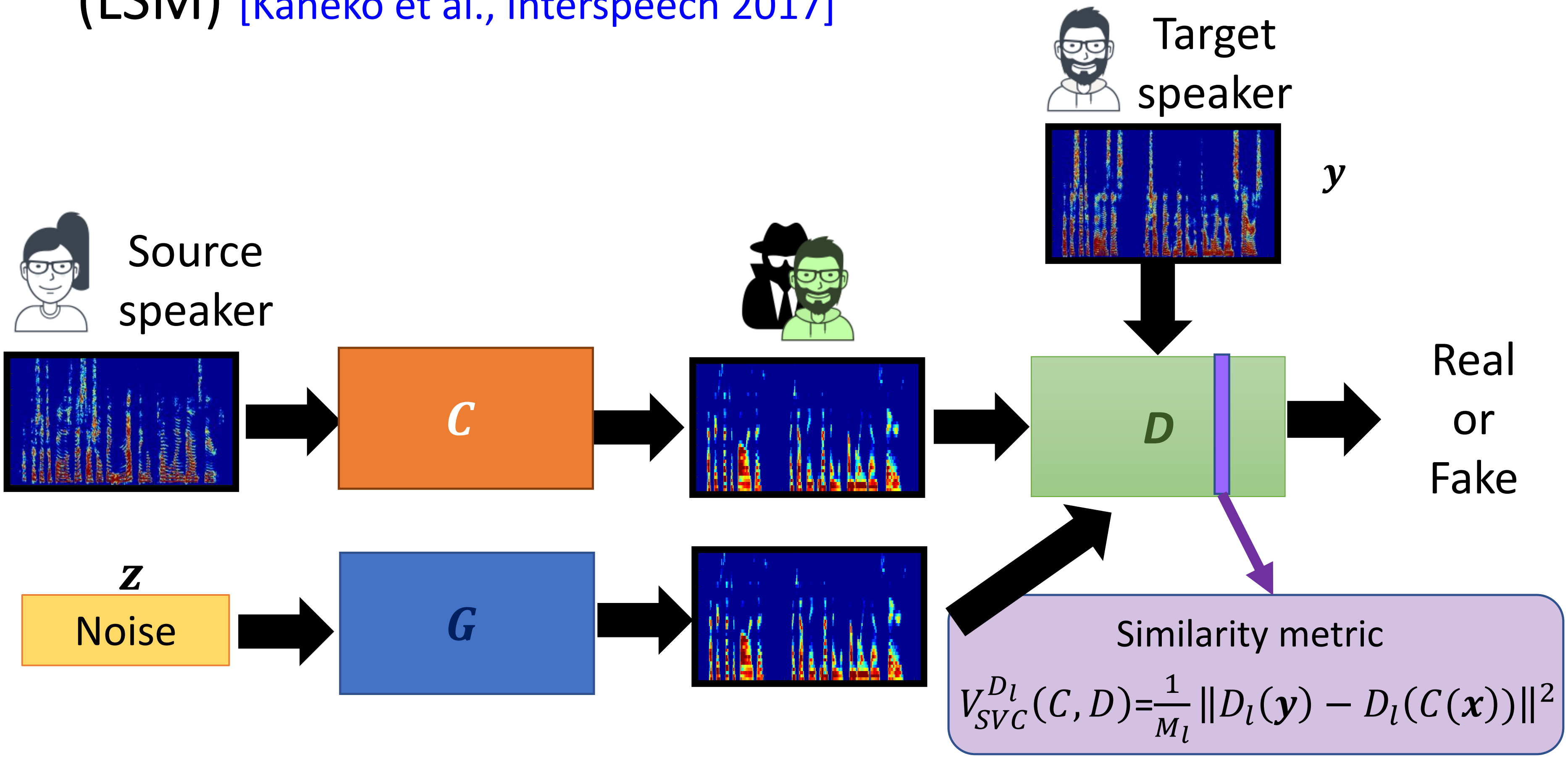
Fig. 15: MOS on naturalness.



VAW-GAN outperforms VAE in terms of objective and subjective evaluations with generating more structured speech.

Voice Conversion

- Sequence-to-sequence VC with learned similarity metric (LSM) [Kaneko et al., Interspeech 2017]

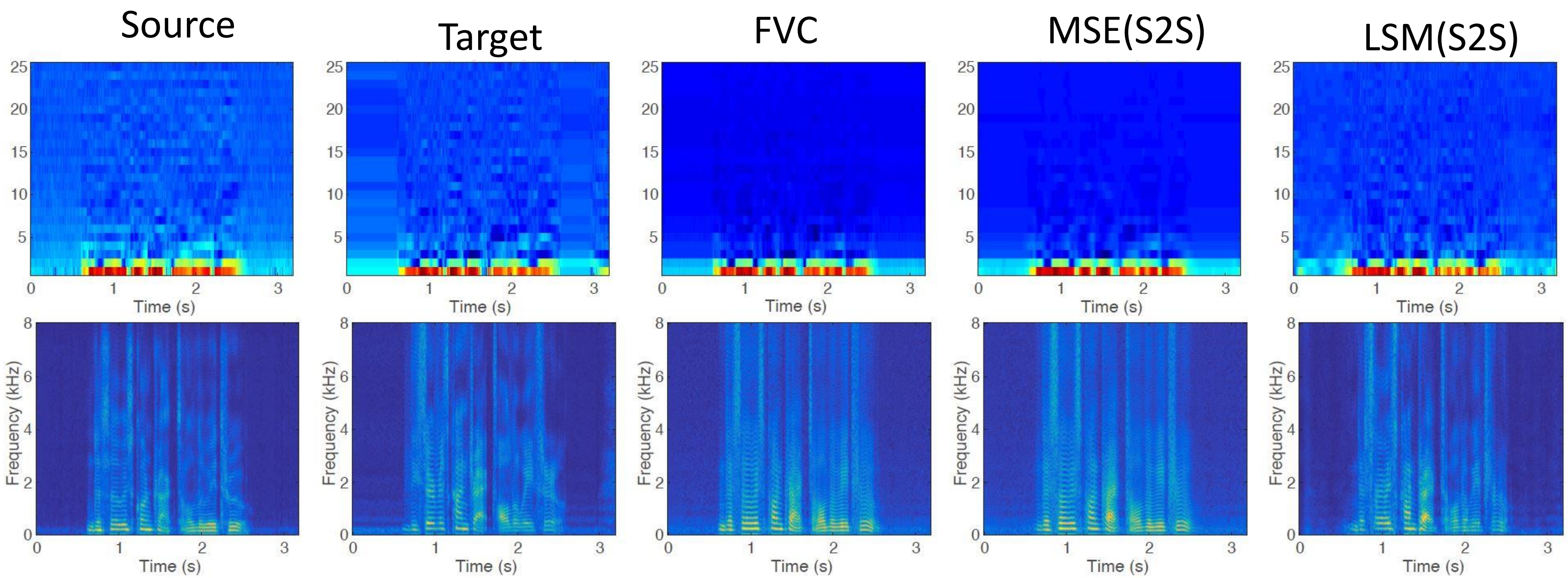


$$V(C,G,D)=V_{SVC}^{D_l}(C,D)+V_{GAN}(C,G,D)$$

Voice Conversion (LSM)

- Spectrogram analysis

Fig. 16: Comparison of MCCs (upper) and STFT spectrograms (lower).



The spectral textures of LSM are more similar to the target ones.

Voice Conversion (LSM)

- Subjective evaluations

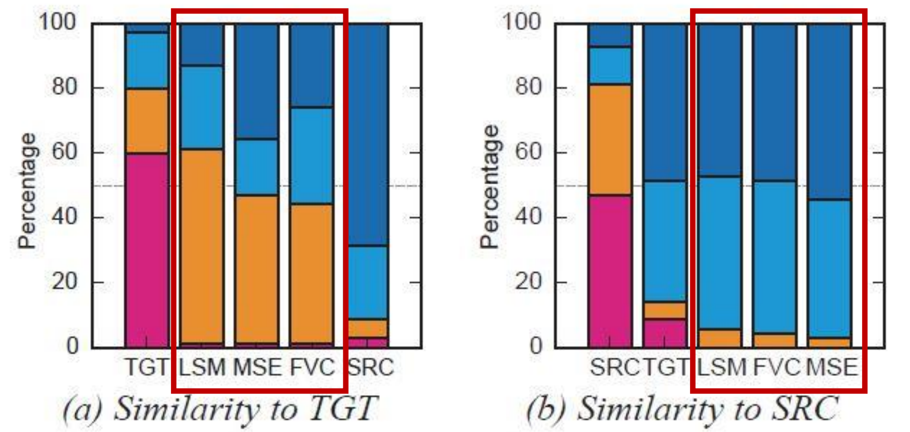
Table 12: Preference scores for naturalness.

	Former	Latter	Neutral
FVC vs. LSM	17.1 ± 6.3	**72.9** ± 7.5	10.0 ± 5.0
MSE vs. LSM	10.0 ± 5.0	**84.3** ± 6.1	5.7 ± 3.9

Table 12: Preference scores for clarity.

	Former	Latter	Neutral
FVC vs. LSM	32.9 ± 7.9	**54.3** ± 8.4	12.9 ± 5.6
MSE vs. LSM	27.1 ± 7.5	**65.0** ± 8.0	7.9 ± 4.5

Fig. 17: Similarity of TGT and SRC with VCs.



(a) Similarity to TGT
(b) Similarity to SRC

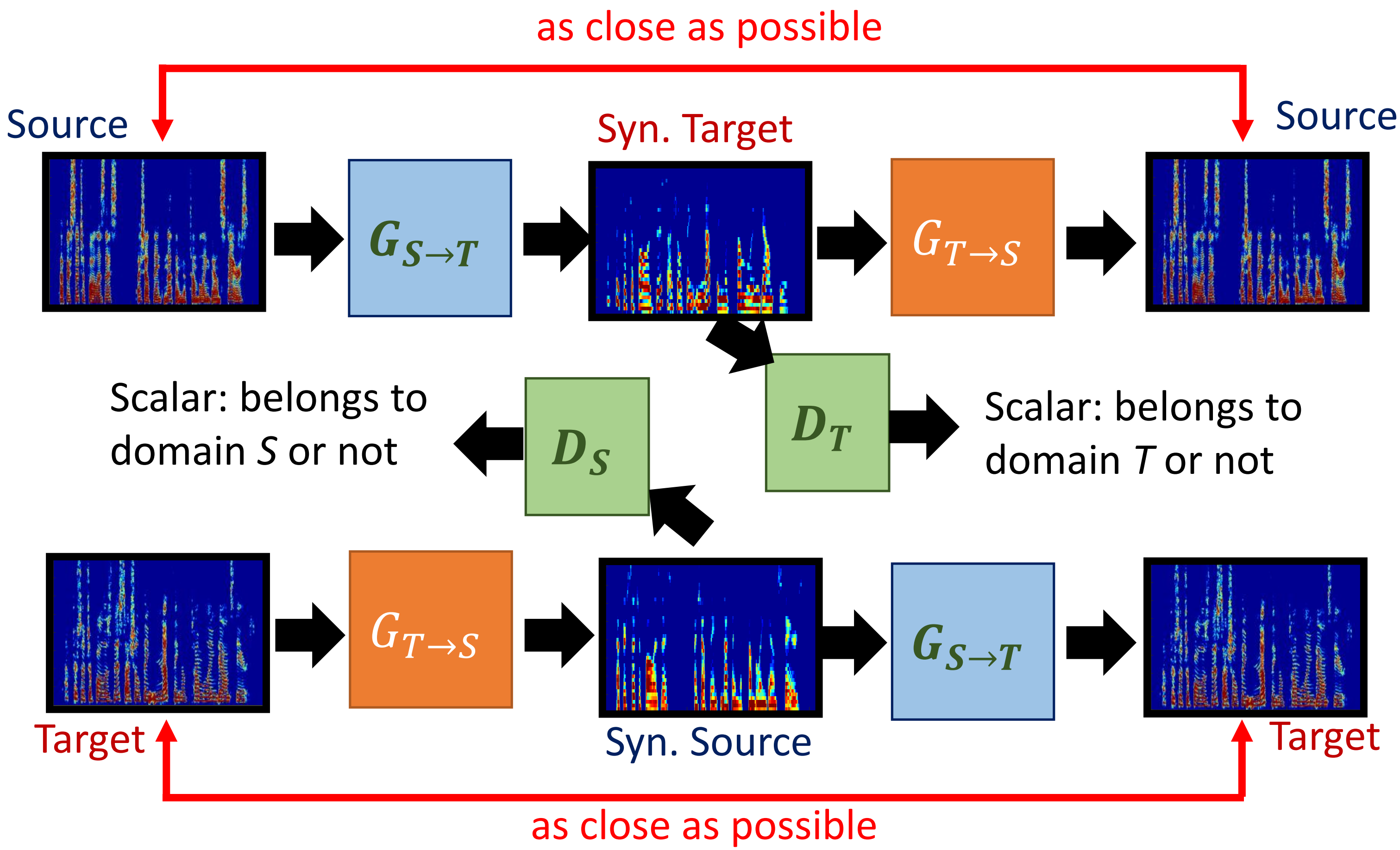

Target speaker


Source speaker

LSM outperforms FVC and MSE in terms of subjective evaluations.

Voice Conversion

- CycleGAN-VC [Kaneko et al., arXiv 2017]



$$V_{Full} = V_{GAN}(G_{X\to Y}, D_Y) + V_{GAN}(G_{X\to Y}, D_Y) + \lambda\, V_{Cyc}(G_{X\to Y}, G_{Y\to X})$$

Voice Conversion (CycleGAN-VC)

- Subjective evaluations

Fig. 18: MOS for naturalness.

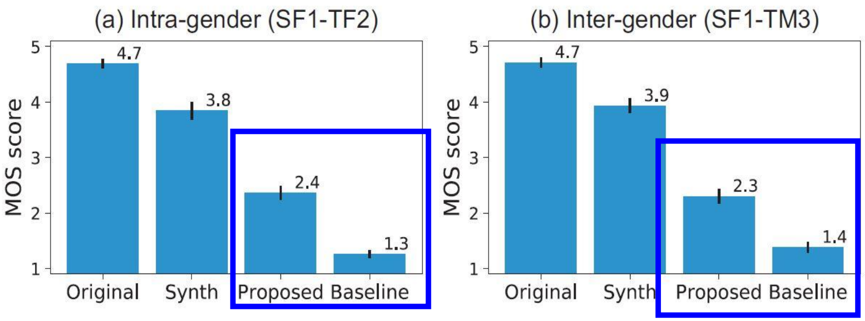
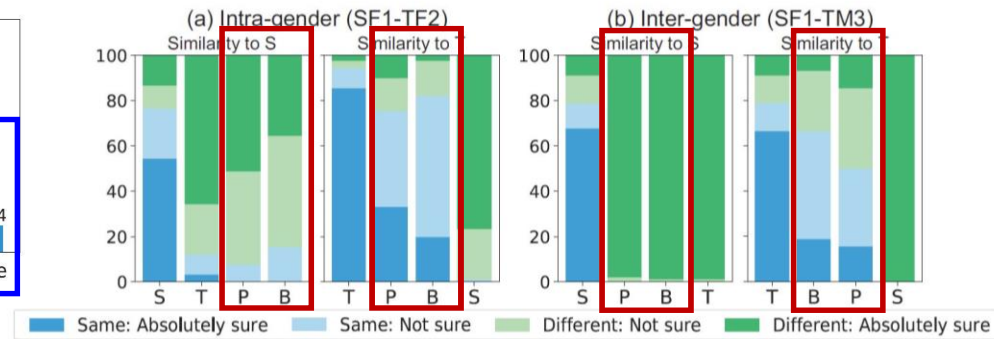


Fig. 19: Similarity of to source and to target speakers. S: Source; T:Target; P: Proposed; B:Baseline




Target speaker

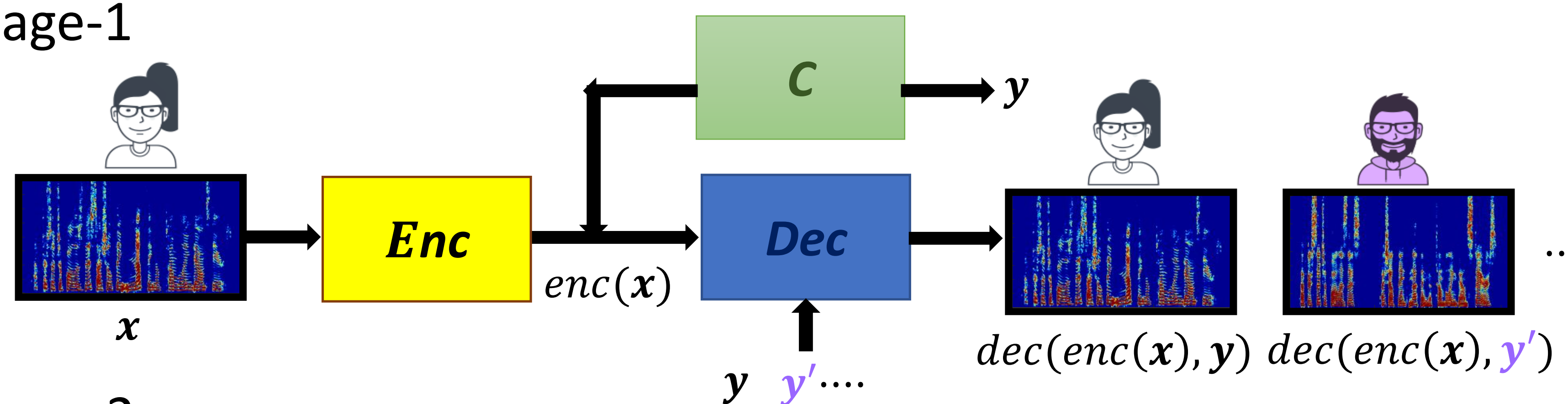

Source speaker

1. The proposed method uses **non-parallel** data.
2. For naturalness, the proposed method outperforms baseline.
3. For similarity, the proposed method is comparable to the baseline.

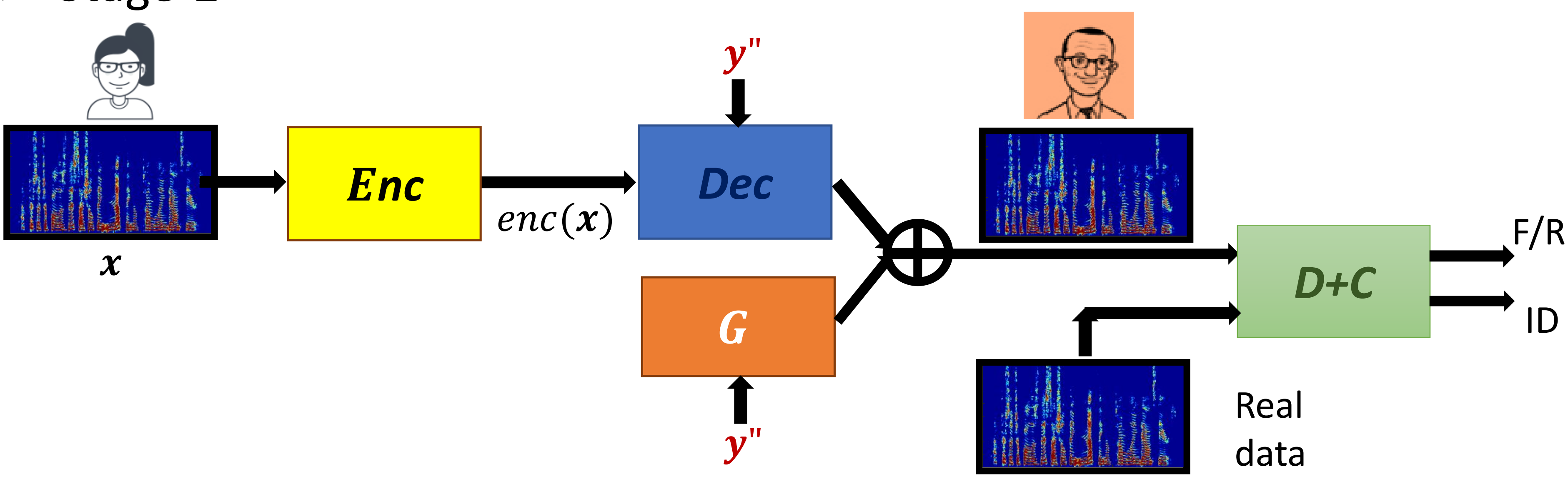
Voice Conversion

- Multi-target VC [Chou et al., arxiv 2018]

➢ Stage-1



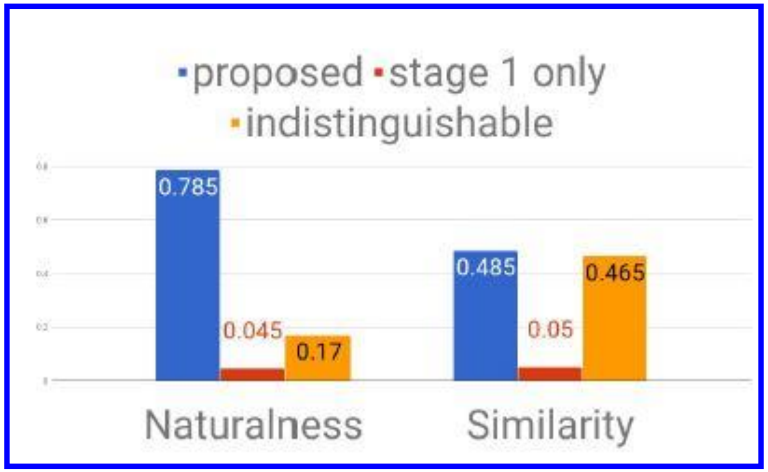
➢ Stage-2

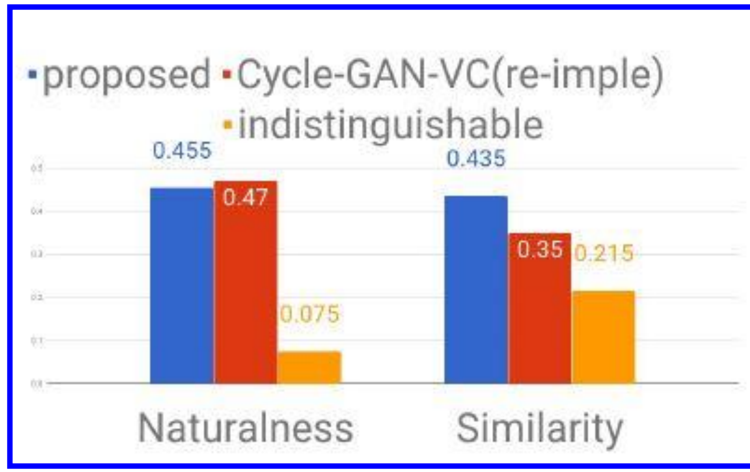


Voice Conversion (Multi-target VC)

- Subjective evaluations

Fig. 20: Preference test results





1. The proposed method uses **non-parallel** data.
2. The multi-target VC approach outperforms one-stage only.
3. The multi-target VC approach is comparable to Cycle-GAN-VC in terms of the naturalness and the similarity.

Outline of Part II

Speech Signal Generation

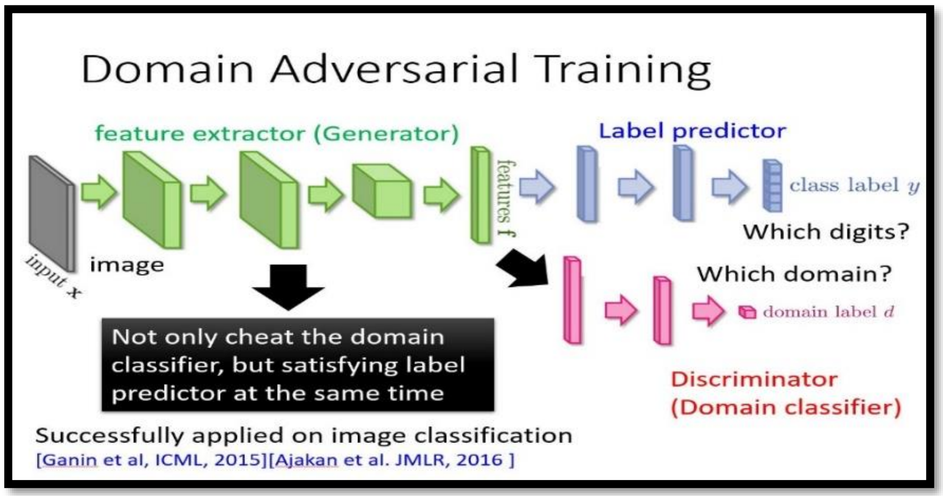
- Speech enhancement
- Postfilter, speech synthesis, voice conversion

Speech Signal Recognition

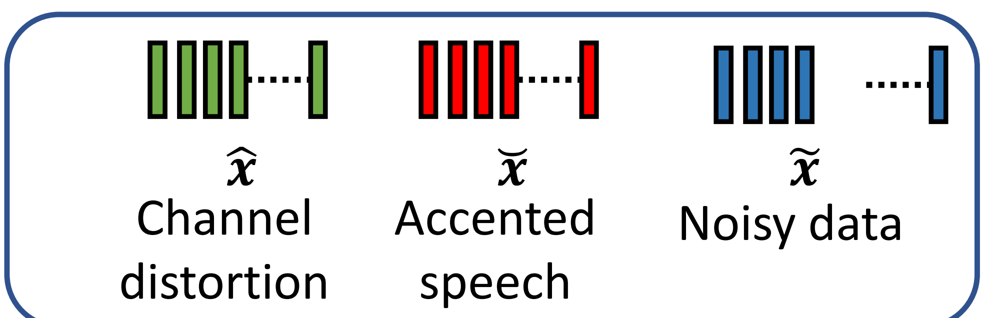
- Speech recognition
- Speaker recognition
- Speech emotion recognition
- Lip reading

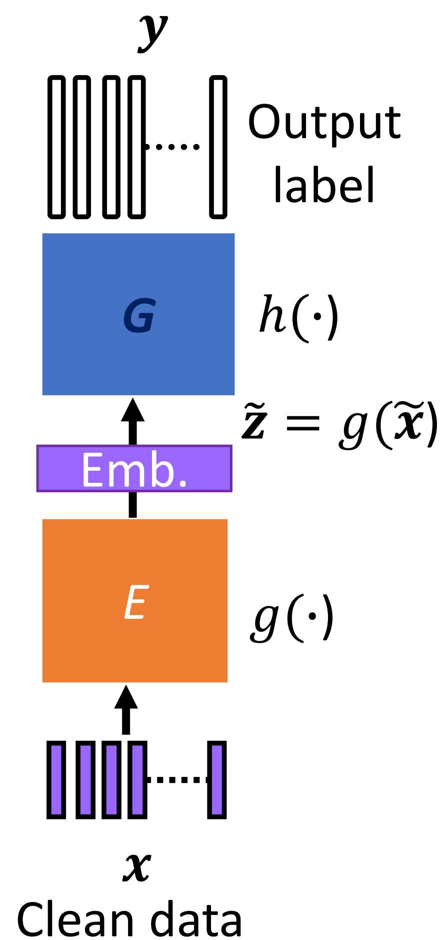
Conclusion

Speech, Speaker, Emotion Recognition and Lip-reading (Classification Task)



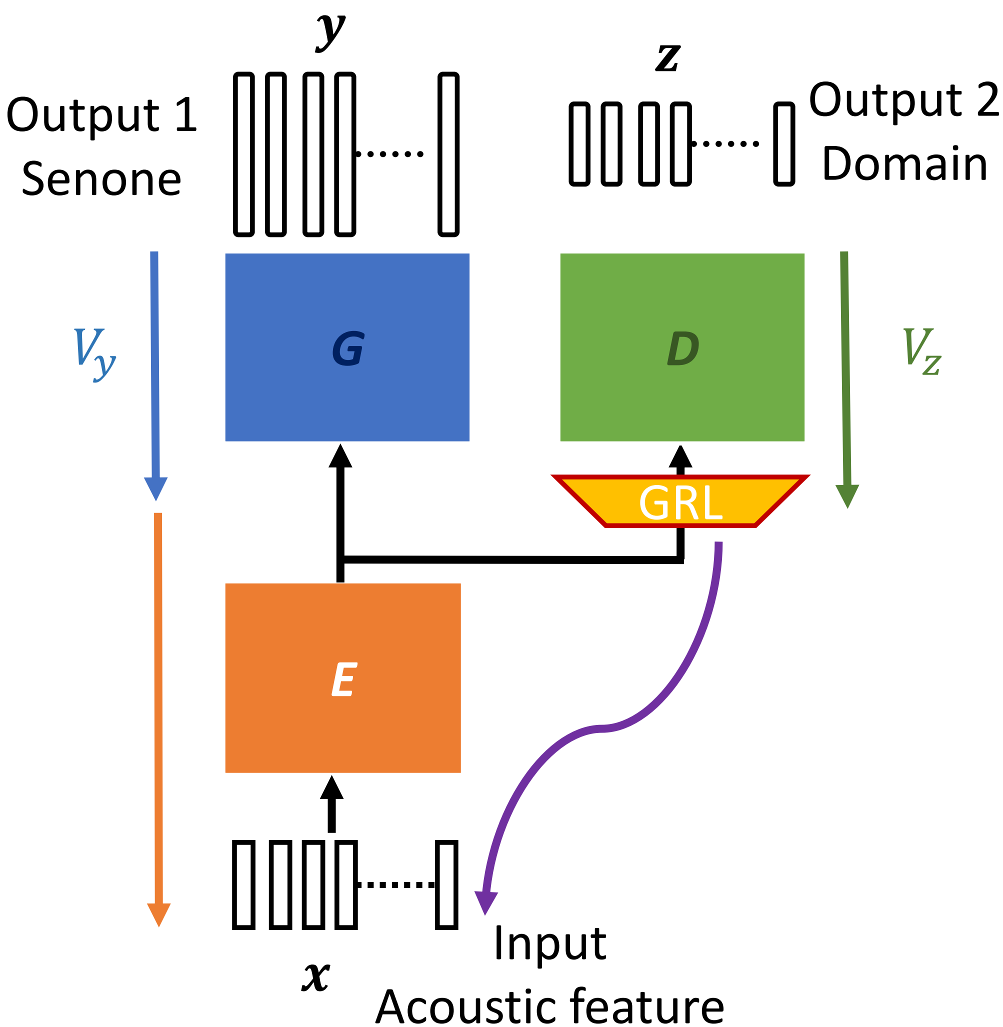
Acoustic Mismatch





Speech Recognition

- Adversarial multi-task learning (AMT)
  [Shinohara Interspeech 2016]



Objective function

$$V_y = -\sum_i \log P(y_i | x_i; \theta_E, \theta_G)$$

$$V_z = -\sum_i \log P(z_i | x_i; \theta_E, \theta_D)$$

Model update

$$\theta_G \leftarrow \theta_G - \epsilon \frac{\partial V_y}{\partial \theta_G}$$

Max classification accuracy

$$\theta_D \leftarrow \theta_D - \epsilon \frac{\partial V_z}{\partial \theta_D}$$

Max domain accuracy

$$\theta_E \leftarrow \theta_E - \epsilon \left(\frac{\partial V_y}{\partial \theta_G} \right) + \alpha \frac{\partial V_z}{\partial \theta_G}$$

Max classification accuracy
and Min domain accuracy

Speech Recognition (AMT)

- ASR results in known (k) and unknown (unk) noisy conditions

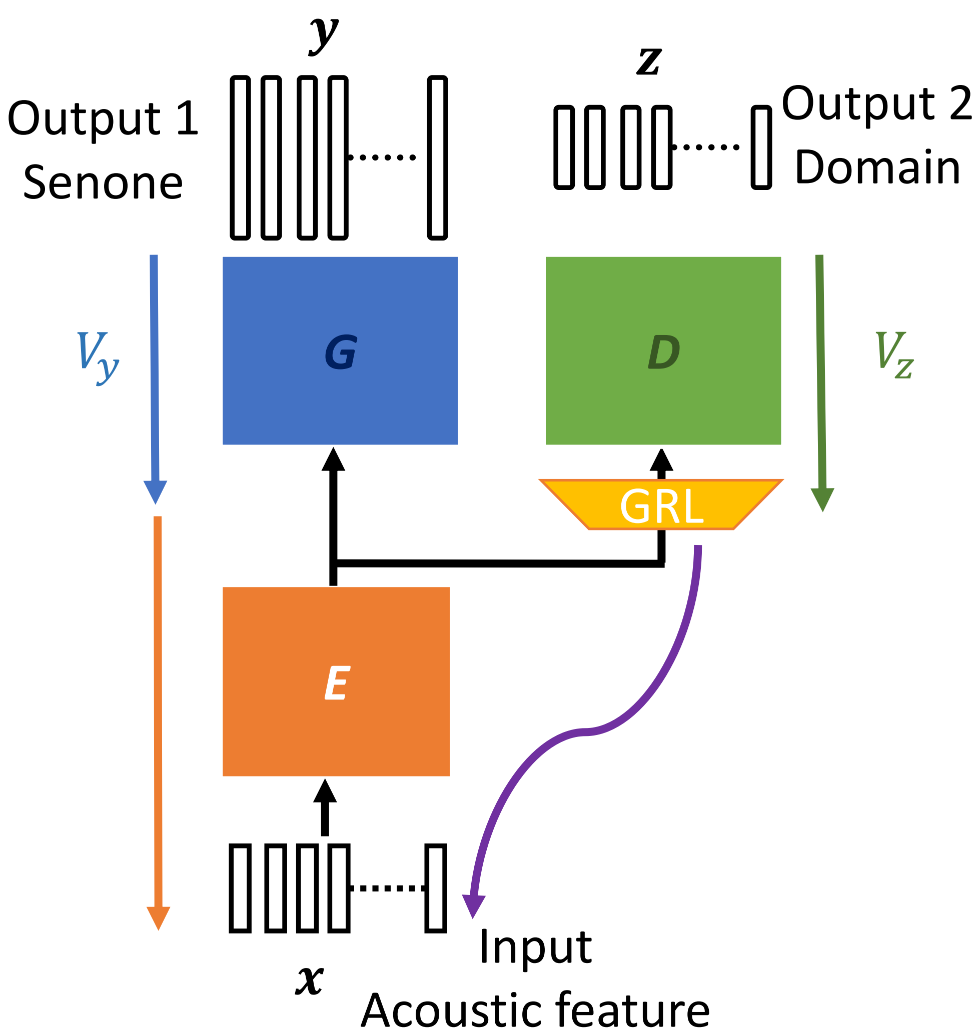
Table 13: WER of DNNs with single-task learning (ST) and AMT.

	noise	ST	AMT	RERR
k	car 2000cc	5.83	5.56	4.63
k	exhib. booth	6.80	6.66	2.06
k	station	7.89	7.76	1.65
k	crossing	6.96	6.65	4.45
unk	car 1500cc	5.58	5.46	2.15
unk	exhib. aisle	7.71	6.93	10.12
unk	factory	12.17	12.92	-6.16
unk	highway	9.73	9.52	2.16
unk	crowd	6.72	6.40	4.76
unk	server room	8.54	7.76	9.13
unk	air cond.	6.96	6.98	-0.29
unk	elev. hall	9.23	9.60	-4.01
-	average	7.84	7.68	2.04

The AMT-DNN outperforms ST-DNN with yielding lower WERs.

Speech Recognition

- Domain adversarial training for accented ASR (DAT)
  [Sun et al., ICASSP2018]



Objective function

$$V_y = -\sum_i \log P(y_i | x_i; \theta_E, \theta_G)$$

$$V_z = -\sum_i \log P(z_i | x_i; \theta_E, \theta_D)$$

Model update

$$\theta_G \leftarrow \theta_G - \epsilon \frac{\partial V_y}{\partial \theta_G}$$

Max classification accuracy

$$\theta_D \leftarrow \theta_D - \epsilon \frac{\partial V_z}{\partial \theta_D}$$

Max domain accuracy

$$\theta_E \leftarrow \theta_E - \epsilon \left(\frac{\partial V_y}{\partial \theta_G} \right) + \alpha \frac{\partial V_z}{\partial \theta_G}$$

Max classification accuracy
and Min domain accuracy

Speech Recognition (DAT)

- ASR results on accented speech

Table 14: WER of the baseline and adapted model.

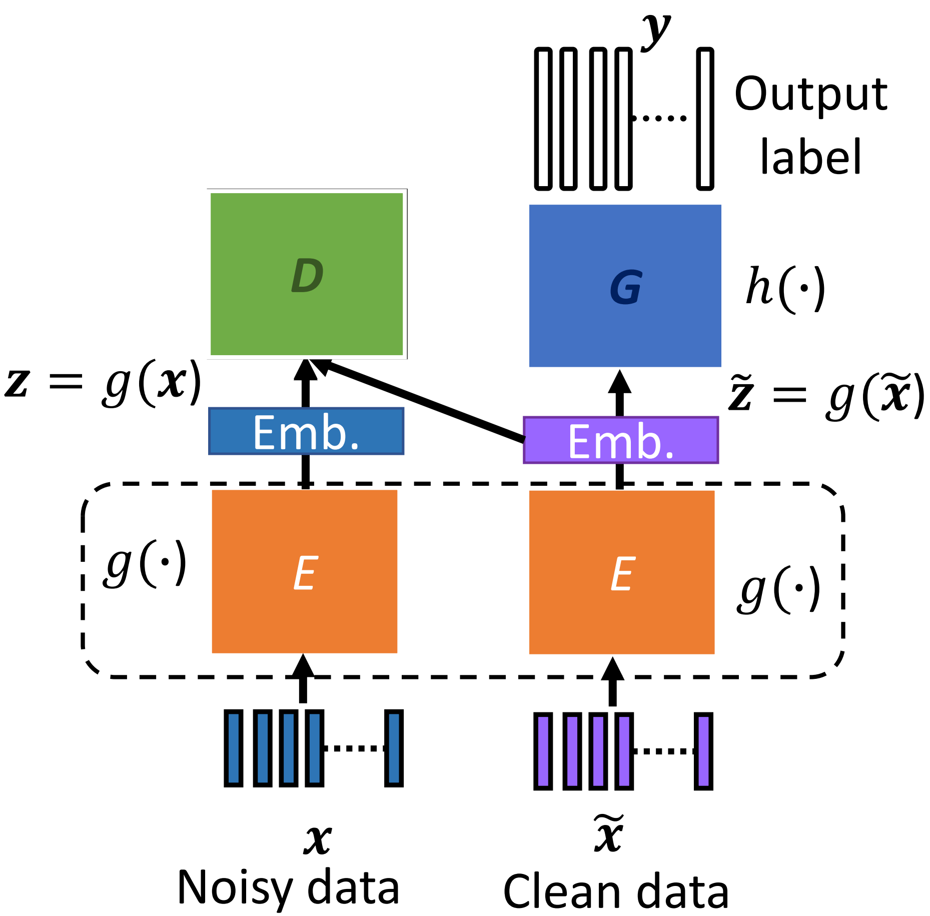
training data	λ	test							
		STD	FJ	JS	JX	SC	GD	HN	Avg.
STD	-	15.55	23.58	15.75	14.08	15.62	15.32	19.34	17.28
STD + (600hrs with trans)	-	14.22	14.84	9.41	8.68	9.13	9.62	11.89	10.60
STD + (600hrs no trans)	0.03	15.37	22.96	14.48	13.79	15.35	14.86	18.24	16.61

STD: standard speech

1. With labeled transcriptions, ASR performance notably improves.
2. DAT is effective in learning features invariant to domain differences with and without labeled transcriptions.

Speech Recognition

- Robust ASR using GAN enhancer (GAN-Enhancer)
  [Sriram et al., arXiv 2017]



Cross entropy with L1 Enhancer:

$$H(h(\tilde{\mathbf{z}}),\mathbf{y}) + \lambda \frac{\|\boldsymbol{z} - \tilde{\mathbf{z}}\|_1}{\|\boldsymbol{z}\|_1 + \|\tilde{\mathbf{z}}\|_1 + \epsilon}$$

Cross entropy with GAN Enhancer:

$$H(h(\tilde{\mathbf{z}}),\mathbf{y}) + \lambda\, V_{adv}\left(g(\boldsymbol{x})\,,\, g(\tilde{\boldsymbol{x}})\right)$$

Speech Recognition (GAN-Enhancer)

- ASR results on far-field speech:

Fig. 15: WER of GAN enhancer and the baseline methods.

Model	Near-Field		Far-Field	
	CER	WER	CER	WER
seq-to-seq	7.43%	21.18%	23.76%	50.84%
seq-to-seq + far-field Augmentation	7.69%	21.32%	12.47%	30.59%
seq-to-seq + L^1-Distance Penalty	7.54%	20.45%	12.00%	29.19%
seq-to-seq + GAN Enhancer	7.78%	21.07%	**11.26%**	**28.12%**

GAN Enhancer outperforms the Augmentation and L1-Enhancer approaches on far-field speech.

Outline of Part II

Speech Signal Generation

- Speech enhancement
- Postfilter, speech synthesis, voice conversion

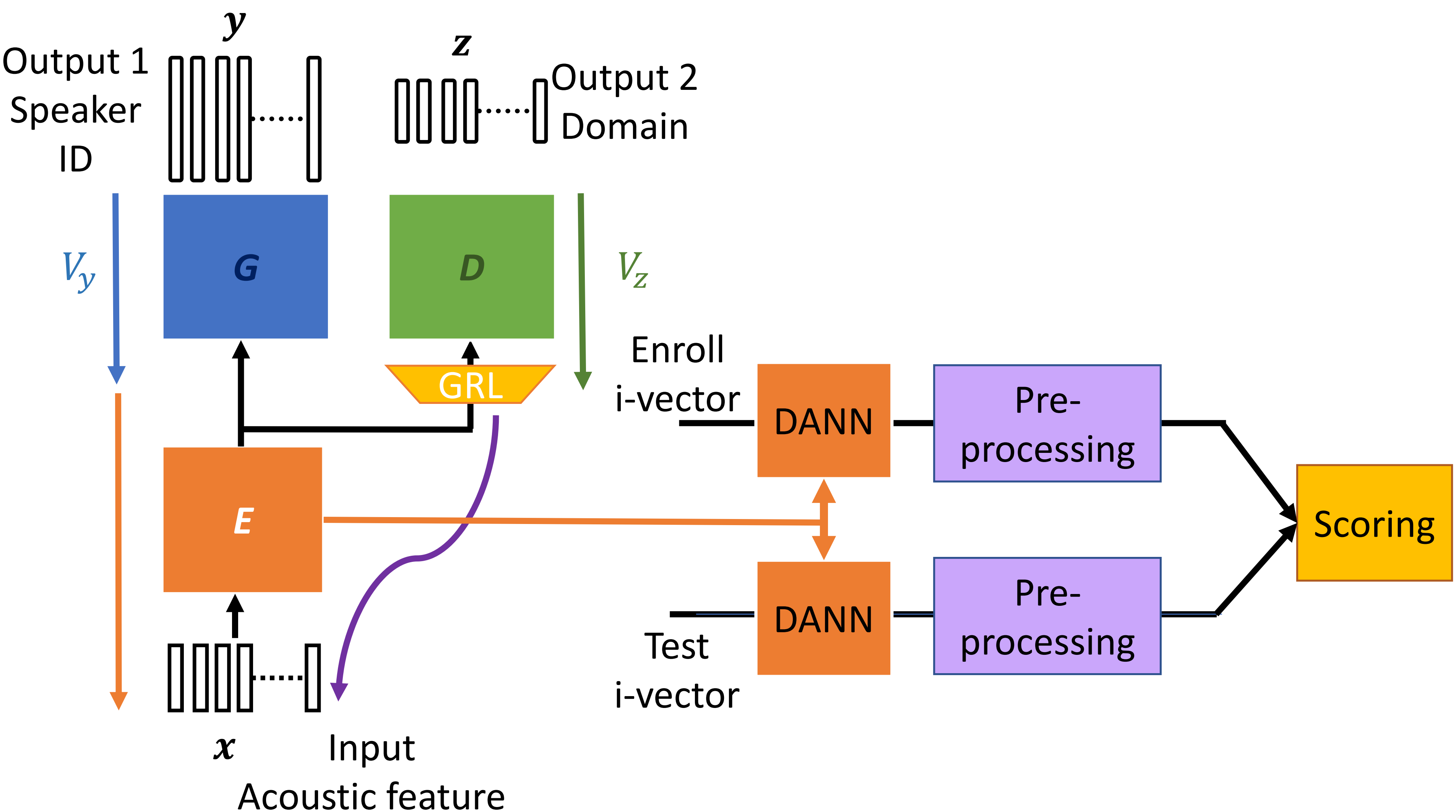
Speech Signal Recognition

- Speech recognition
- Speaker recognition
- Speech emotion recognition
- Lip reading

Conclusion

Speaker Recognition

- Domain adversarial neural network (DANN)
  [Wang et al., ICASSP 2018]



Speaker Recognition (DANN)

- Recognition results of domain mismatched conditions

Table 16: Performance of DAT and the state-of-the-art methods.

Systems#	Adaptation Methods	EER%	DCF10 [21]	DCF08
1	–	9.35	0.724	0.520
2	–	5.66	0.633	0.427
3	Interpolated [6] [12]	6.55	0.652	0.454
4	IDV [9] [12]	6.15	0.676	0.476
5	DICN [11] [12]	4.99	0.623	0.416
6	DAE [22] [12]	4.81	0.610	0.398
7	AEDA [12]	4.50	0.589	0.362
8	**DAT**	**3.73**	**0.541**	**0.335**

The DAT approach outperforms other methods with achieving lowest EER and DCF scores.

Outline of Part II

Speech Signal Generation

- Speech enhancement
- Postfilter, speech synthesis, voice conversion

Speech Signal Recognition

- Speech recognition
- Speaker recognition
- Speech emotion recognition
- Lip reading

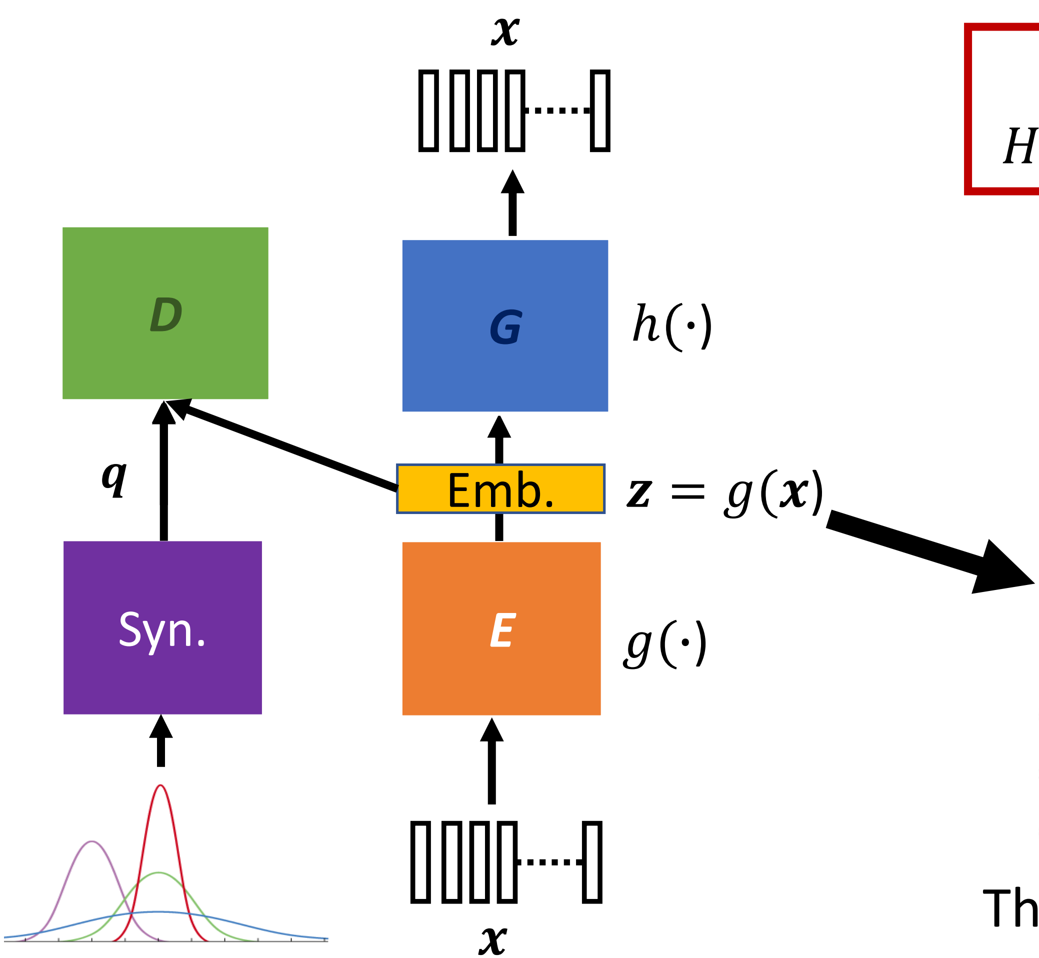
Conclusion

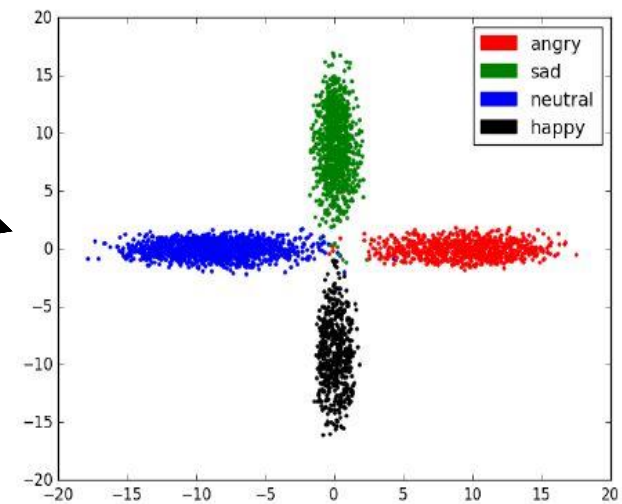
Emotion Recognition

- Adversarial AE for emotion recognition (AAE-ER)
  [Sahu et al., Interspeech 2017]

AE with GAN :

$$H(h(\boldsymbol{z}), \boldsymbol{x}) + \lambda V_{GAN}(\boldsymbol{q}, g(\boldsymbol{x}))$$





The distribution of code vectors

Emotion Recognition (AAE-ER)

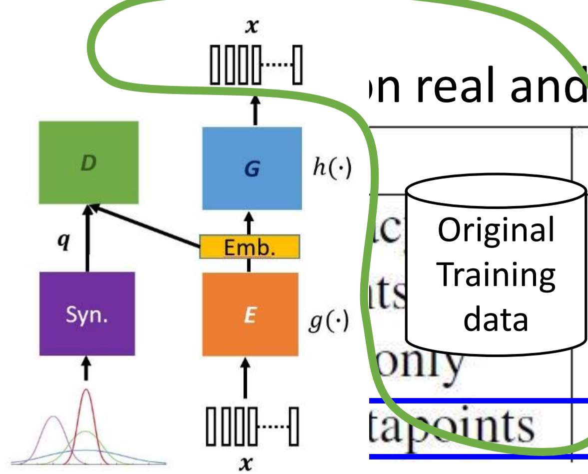
- Recognition results of domain mismatched conditions:

Table 17: Classification results on different systems.

	OpenSmile features (1582-D)	Code vectors (2-D)	Auto-encoder (100-D)	LDA (2-D)	PCA (2-D)
UAR (%)	57.88	56.38	53.92	48.67	43.12

Table 18: Class[illegible]n real and synthesized features.

	UAR (%)
[illegible]c	25.00
S [illegible]ts	33.75
[illegible]only	57.88
Sy[illegible]tapoints	**58.38**



1. AAE alone could not yield performance improvements.
2. Using synthetic data from AAE can yield higher UAR.

Outline of Part II

Speech Signal Generation

- Speech enhancement
- Postfilter, speech synthesis, voice conversion

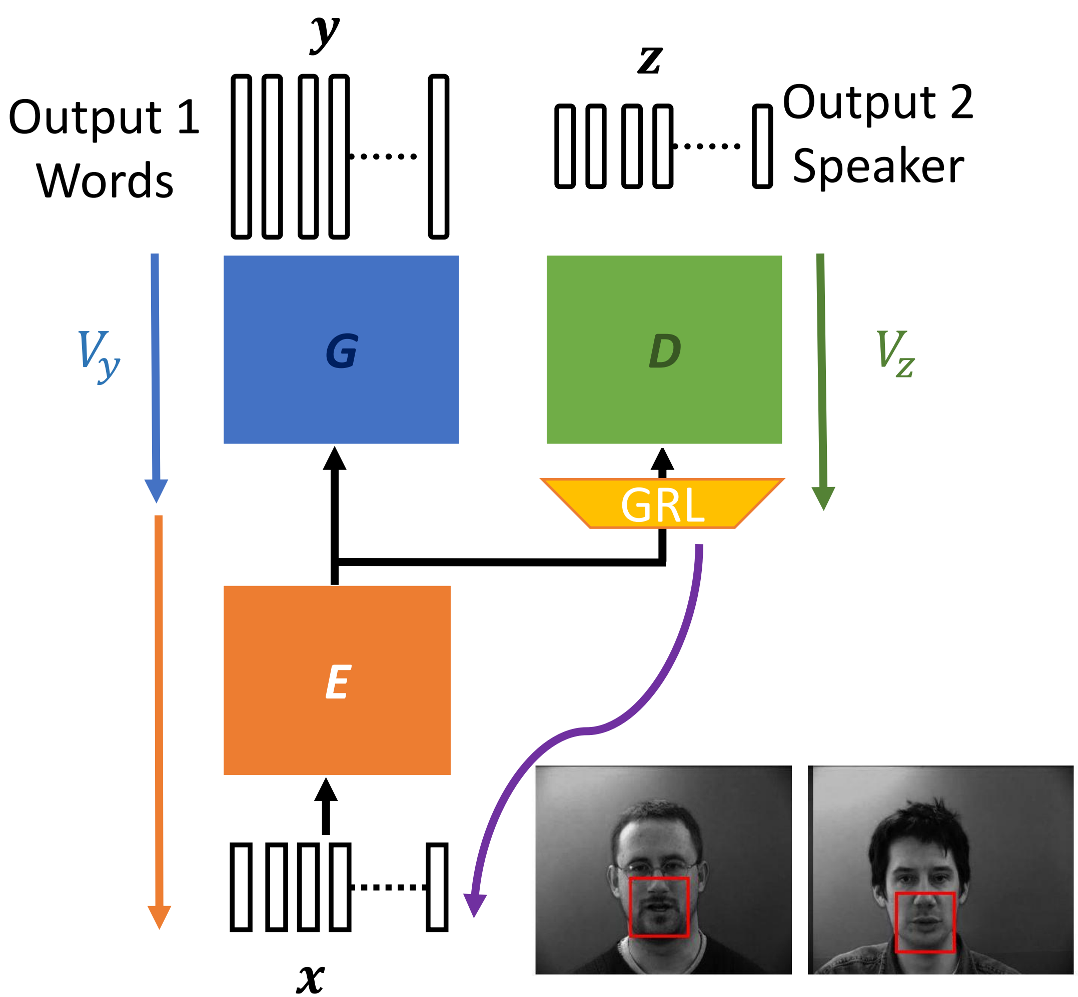
Speech Signal Recognition

- Speech recognition
- Speaker recognition
- Speech emotion recognition
- Lip reading

Conclusion

Lip-reading

- Domain adversarial training for lip-reading (DAT-LR) [Wand et al., arXiv 2017]



~80% WAC

Objective function

$$V_y = -\sum_i \log P(y_i | x_i; \theta_E, \theta_G)$$

$$V_z = -\sum_i \log P(z_i | x_i; \theta_E, \theta_D)$$

Model update

$$\theta_G \leftarrow \theta_G - \epsilon \frac{\partial V_y}{\partial \theta_G}$$

Max classification accuracy

$$\theta_D \leftarrow \theta_D - \epsilon \frac{\partial V_z}{\partial \theta_D}$$

Max domain accuracy

$$\theta_E \leftarrow \theta_E - \epsilon \left(\frac{\partial V_y}{\partial \theta_G} \right) + \alpha \frac{\partial V_z}{\partial \theta_G}$$

Max classification accuracy and Min domain accuracy

Lip-reading (DAT-LR)

- Recognition results of speaker mismatched conditions

Table 19: Performance of DAT and the baseline.

Adversarial Training on	Number of training spk	Target Test acc.	Relative Improvement	p-value
None	1	18.7%	-	-
	4	39.4%	-	-
	8	46.5%	-	-
All Target Sequences	1	25.4%	35.8%	0.0030∗
	4	43.6%	10.7%	0.0261∗
	8	49.3%	6.0%	0.0266∗
50 Target Sequences	1	24.1%	28.9%	0.0045∗
	4	41.5%	5.3%	0.1367
	8	47.0%	1.1%	0.3555

The DAT approach notably enhances the recognition accuracies in different conditions.

Outline of Part II

Speech Signal Generation

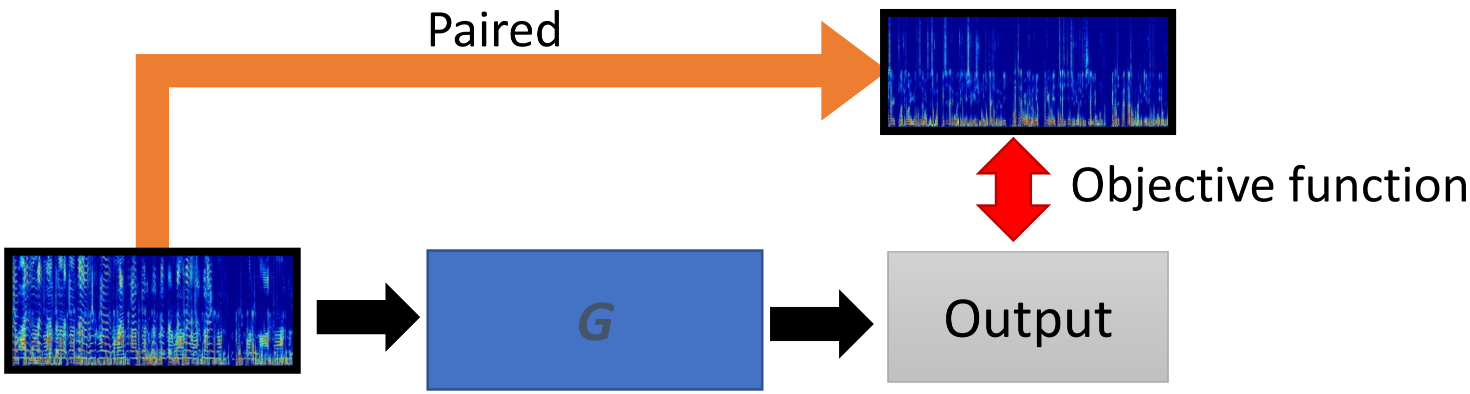
- Speech enhancement
- Postfilter, speech synthesis, voice conversion

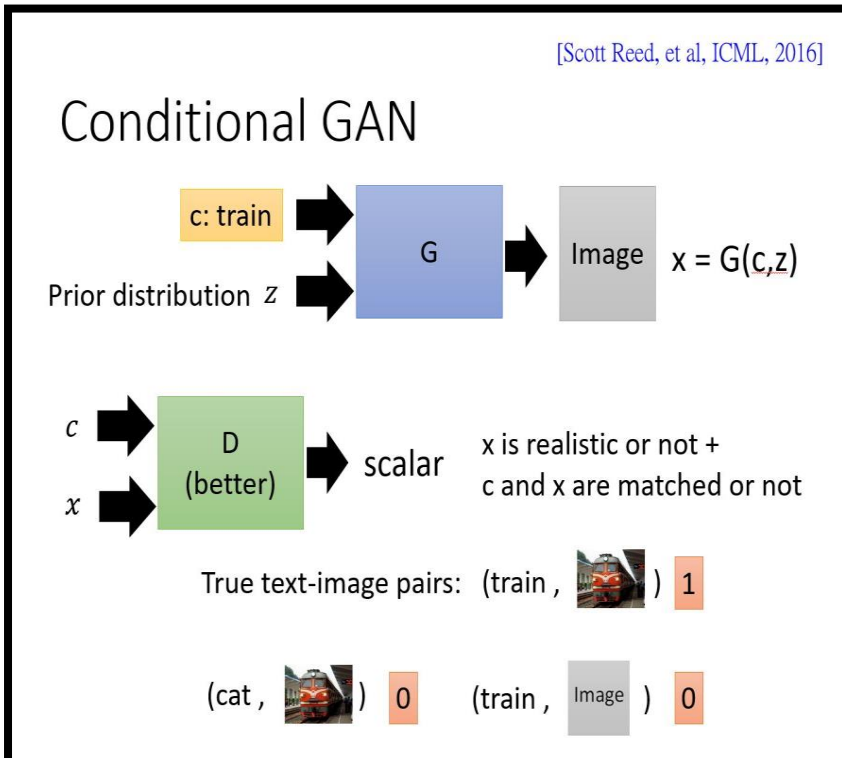
Speech Signal Recognition

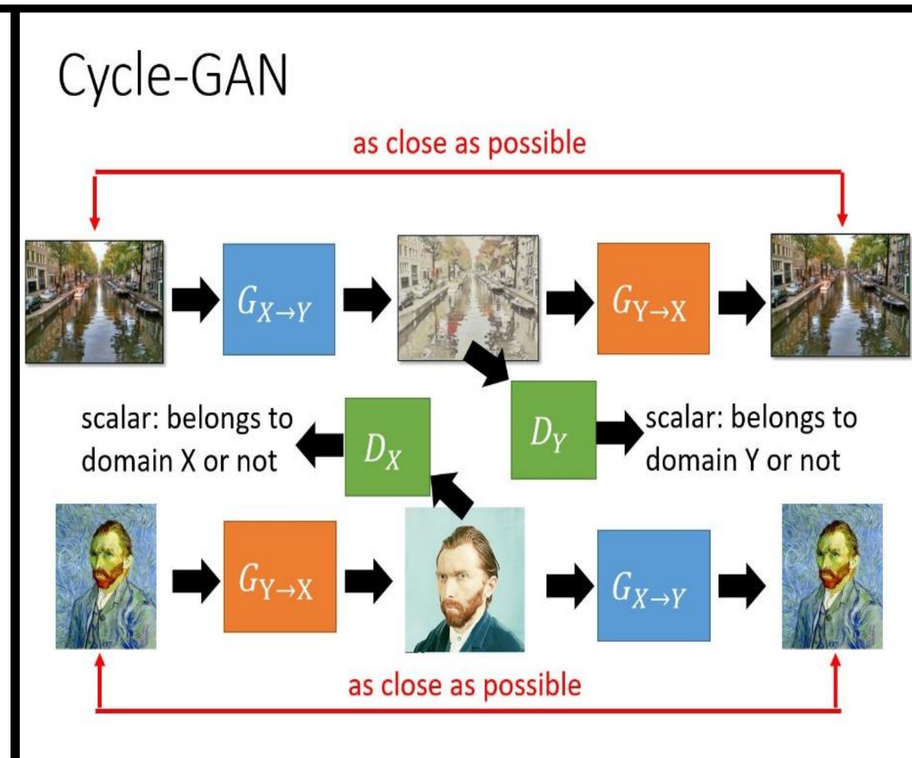
- Speech recognition
- Speaker recognition
- Speech emotion recognition
- Lip reading

Conclusion

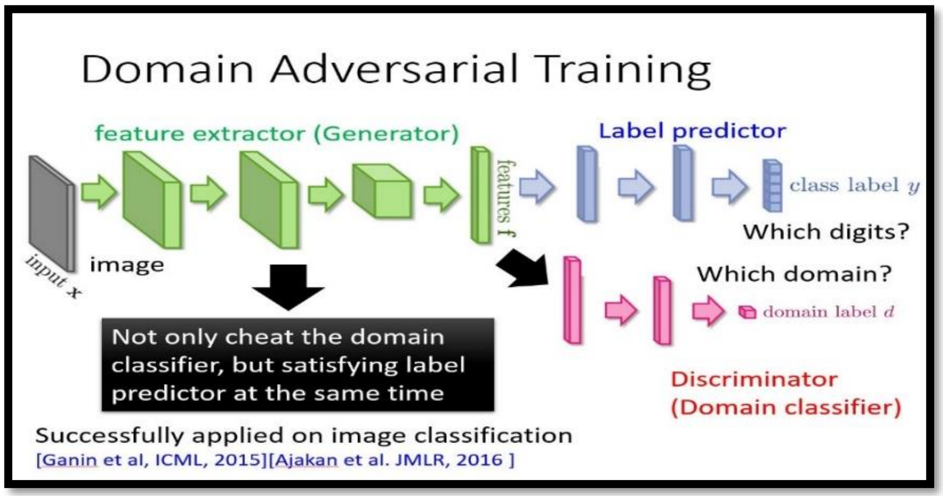
Speech Signal Generation (Regression Task)



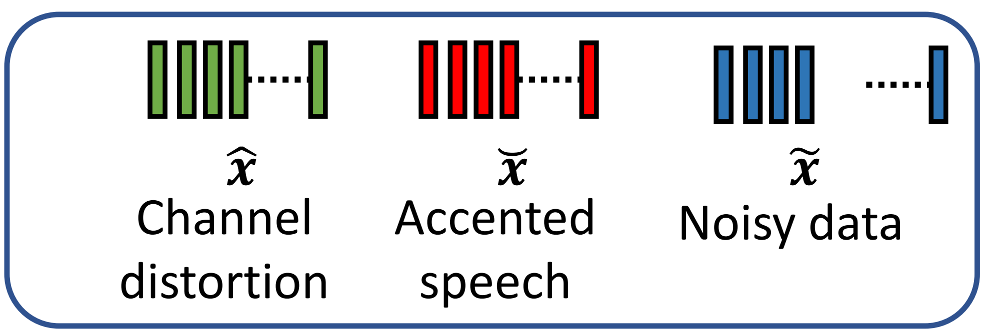


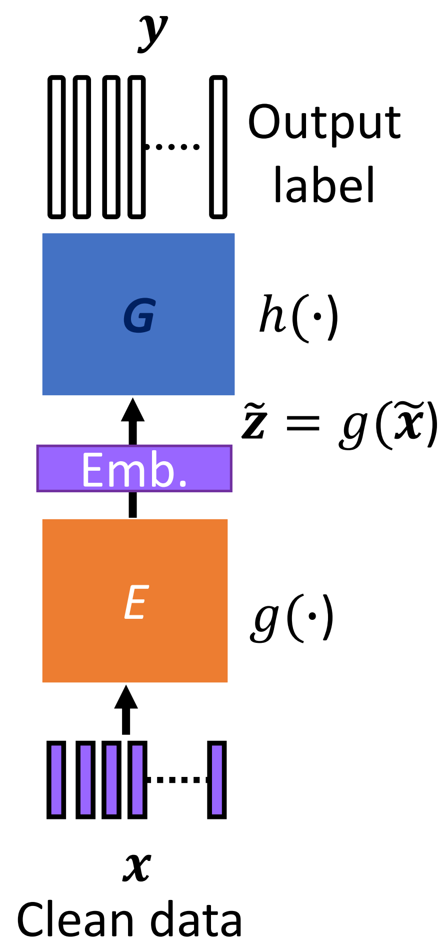


Speech, Speaker, Emotion Recognition and Lip-reading (Classification Task)



Acoustic Mismatch





More GANs in Speech

Diagnosis of autism spectrum
Jun Deng, Nicholas Cummins, Maximilian Schmitt, Kun Qian, Fabien Ringeval, and Björn Schuller, Speech-based Diagnosis of Autism Spectrum Condition by Generative Adversarial Network Representations, ACM DH, 2017.

Emotion recognition
Jonathan Chang, and Stefan Scherer, Learning Representations of Emotional Speech with Deep Convolutional Generative Adversarial Networks, ICASSP, 2017.

Robust ASR
Dmitriy Serdyuk, Kartik Audhkhasi, Philémon Brakel, Bhuvana Ramabhadran, Samuel Thomas, and Yoshua Bengio, Invariant Representations for Noisy Speech Recognition, arXiv, 2016.

Speaker verification
Hong Yu, Zheng-Hua Tan, Zhanyu Ma, and Jun Guo, Adversarial Network Bottleneck Features for Noise Robust Speaker Verification, arXiv, 2017.

References

Speech enhancement (conventional methods)

- Yuxuan Wang and Deliang Wang, Cocktail Party Processing via Structured Prediction, NIPS 2012.
- Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee, An Experimental Study on Speech Enhancement Based on Deep Neural Networks," IEEE SPL, 2014.
- Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee, A Regression Approach to Speech Enhancement Based on Deep Neural Networks, IEEE/ACM TASLP, 2015.
- Xugang Lu, Yu Tsao, Shigeki Matsuda, Chiori Hori, Speech Enhancement Based on Deep Denoising Autoencoder, Interspeech 2012.
- Zhuo Chen, Shinji Watanabe, Hakan Erdogan, John R. Hershey, Integration of Speech Enhancement and Recognition Using Long-short term Memory Recurrent Neural Network, Interspeech 2015.
- Felix Weninger, Hakan Erdogan, Shinji Watanabe, Emmanuel Vincent, Jonathan Le Roux, John R. Hershey, and Bjorn Schuller, Speech Enhancement with LSTM Recurrent Neural Networks and Its Application to Noise-robust ASR, LVA/ICA, 2015.
- Szu-Wei Fu, Yu Tsao, and Xugang Lu, SNR-aware Convolutional Neural Network Modeling for Speech Enhancement, Interspeech, 2016.
- Szu-Wei Fu, Yu Tsao, Xugang Lu, and Hisashi Kawai, End-to-end Waveform Utterance Enhancement for Direct Evaluation Metrics Optimization by Fully Convolutional Neural Networks, arXiv, IEEE/ACM TASLP, 2018.

Speech enhancement (GAN-based methods)

- Pascual Santiago, Bonafonte Antonio, and Serra Joan, SEGAN: Speech Enhancement Generative Adversarial Network, Interspeech, 2017.
- Michelsanti Daniel, and Zheng-Hua Tan, Conditional Generative Adversarial Networks for Speech Enhancement and Noise-robust Speaker Verification, Interspeech, 2017.
- Donahue Chris, Li Bo, and Prabhavalkar Rohit, Exploring Speech Enhancement with Generative Adversarial Networks for Robust Speech Recognition, ICASPP, 2018.
- Higuchi Takuya, Kinoshita Keisuke, Delcroix Marc, and Nakatani Tomohiro, Adversarial Training for Data-driven Speech Enhancement without Parallel Corpus, ASRU, 2017.

References

Postfilter (conventional methods)

- Toda Tomoki, and Tokuda Keiichi, A Speech Parameter Generation Algorithm Considering Global Variance for HMM-based Speech Synthesis, IEICE Trans. Inf. Syst., 2007.
- Sil’en Hanna, Helander Elina, Nurminen Jani, and Gabbouj Moncef, Ways to Implement Global Variance in Statistical Speech Synthesis, Interspeech, 2012.
- Takamichi Shinnosuke, Toda Tomoki, Neubig Graham, Sakti Sakriani, and Nakamura Satoshi, A Postfilter to Modify the Modulation Spectrum in HMM-based Speech Synthesis, ICASSP, 2014.
- Ling-Hui Chen, Tuomo Raitio, Cassia Valentini-Botinhao, Junichi Yamagishi, and Zhen-Hua Ling, DNN-based Stochastic Postfilter for HMM-based Speech Synthesis, Interspeech, 2014.
- Ling-Hui Chen, Tuomo Raitio, Cassia Valentini-Botinhao, Zhen-Hua Ling, and Junichi Yamagishi, A Deep Generative Architecture for Postfiltering in Statistical Parametric Speech Synthesis, IEEE/ACM TASLP, 2015.

Postfilter (GAN-based methods)

- Kaneko Takuhiro, Kameoka Hirokazu, Hojo Nobukatsu, Ijima Yusuke, Hiramatsu Kaoru, and Kashino Kunio, Generative Adversarial Network-based Postfilter for Statistical Parametric Speech Synthesis, ICASSP, 2017.
- Kaneko Takuhiro, Takaki Shinji, Kameoka Hirokazu, and Yamagishi Junichi, Generative Adversarial Network-Based Postfilter for STFT Spectrograms, Interspeech, 2017.
- Saito Yuki, Takamichi Shinnosuke, and Saruwatari Hiroshi, Training Algorithm to Deceive Anti-spoofing Verification for DNN-based Speech Synthesis, ICASSP, 2017.
- Saito Yuki, Takamichi Shinnosuke, Saruwatari Hiroshi, Saito Yuki, Takamichi Shinnosuke, and Saruwatari Hiroshi, Statistical Parametric Speech Synthesis Incorporating Generative Adversarial Networks, IEEE/ACM TASLP, 2018.
- Bajibabu Bollepalli, Lauri Juvela, and Alku Paavo, Generative Adversarial Network-based Glottal Waveform Model for Statistical Parametric Speech Synthesis, Interspeech, 2017.
- Yang Shan, Xie Lei, Chen Xiao, Lou Xiaoyan, Zhu Xuan, Huang Dongyan, and Li Haizhou, Statistical Parametric Speech Synthesis Using Generative Adversarial Networks Under a Multi-task Learning Framework, ASRU, 2017.

References

VC (conventional methods)

- Toda Tomoki, Black Alan W, and Tokuda Keiichi, Voice Conversion Based on Maximum Likelihood Estimation of Spectral Parameter Trajectory, IEEE/ACM TASLP, 2007.
- Ling-Hui Chen, Zhen-Hua Ling, Li-Juan Liu, and Li-Rong Dai, Voice Conversion Using Deep Neural Networks with Layer-wise Generative Training, IEEE/ACM TASLP, 2014.
- Srinivas Desai, Alan W Black, B. Yegnanarayana, and Kishore Prahallad, Spectral mapping Using artificial Neural Networks for Voice Conversion, IEEE/ACM TASLP, 2010.
- Nakashika Toru, Takiguchi Tetsuya, Ariki Yasuo, High-order Sequence Modeling Using Speaker-dependent Recurrent Temporal Restricted Boltzmann Machines for Voice Conversion, Interspeech, 2014.
- Takuhiro Kaneko, Hirokazu Kameoka, Kaoru Hiramatsu, and Kunio Kashino, Sequence-to-sequence Voice Conversion with Similarity Metric Learned Using Generative Adversarial Networks, Interspeech, 2017.
- Zhizheng Wu, Tuomas Virtanen, Eng-Siong Chng, and Haizhou Li, Exemplar-based Sparse Representation with Residual Compensation for Voice Conversion, IEEE/ACM TASLP, 2014.
- Szu-Wei Fu, Pei-Chun Li, Ying-Hui Lai, Cheng-Chien Yang, Li-Chun Hsieh, and Yu Tsao, Joint Dictionary Learning-based Non-negative Matrix Factorization for Voice Conversion to Improve Speech Intelligibility After Oral Surgery, IEEE TBME, 2017.
- Yi-Chiao Wu, Hsin-Te Hwang, Chin-Cheng Hsu, Yu Tsao, and Hsin-Min Wang, Locally Linear Embedding for Exemplar-based Spectral Conversion, Interspeech, 2016.

VC (GAN-based methods)

- Chin-Cheng Hsu, Hsin-Te Hwang, Yi-Chiao Wu, Yu Tsao, and Hsin-Min Wang Voice Conversion from Unaligned Corpora Using Variational Autoencoding Wasserstein Generative Adversarial Networks, arXiv, 2017.
- Takuhiro Kaneko, Hirokazu Kameoka, Kaoru Hiramatsu, and Kunio Kashino, Sequence-to-sequence Voice Conversion with Similarity Metric Learned Using Generative Adversarial Networks, Interspeech, 2017.
- Takuhiro Kaneko, and Hirokazu Kameoka. Parallel-Data-Free Voice Conversion Using Cycle-Consistent Adversarial Networks, arXiv, 2017.

References

ASR

- Yusuke Shinohara, Adversarial Multi-Task Learning of Deep Neural Networks for Robust Speech Recognition. Interspeech, 2016.
- Sining Sun, Ching-Feng Yeh, Mei-Yuh Hwang, Mari Ostendorf, and Lei Xie, Domain Adversarial Training for Accented Speech Recognition, ICASSP, 2018
- Masato Mimura, Shinsuke Sakai, and Tatsuya Kawahara, Cross-domain Speech Recognition Using Nonparallel Corpora with Cycle-consistent Adversarial Networks, ASRU, 2017.
- Anuroop Sriram, Heewoo Jun, Yashesh Gaur, and Sanjeev Satheesh, Robust Speech Recognition Using Generative Adversarial Networks, arXiv, 2017.

Speaker recognition

- Qing Wang, Wei Rao, Sining Sun, Lei Xie, Eng Siong Chng, and Haizhou Li, Unsupervised Domain Adaptation via Domain Adversarial Training for Speaker Recognition, ICASSP, 2018.

Emotion recognition

- Saurabh Sahu, Rahul Gupta, Ganesh Sivaraman, Wael AbdAlmageed, and Carol Espy-Wilson, Adversarial Auto-encoders for Speech Based Emotion Recognition. Interspeech, 2017.

Lipreading

- Michael Wand, and Jürgen Schmidhuber, Improving Speaker-Independent Lipreading with Domain-Adversarial Training, arXiv, 2017.

GANs in ICASSP 2018

- Saurabh Sahu, Rahul Gupta, Ganesh Sivaraman, Carol Espy-Wilson, Smoothing Model Predictions using Adversarial Training Procedures for Speech Based Emotion Recognition
- Fuming Fang, Junichi Yamagishi, Isao Echizen, Jaime Lorenzo-Trueba, High-quality Nonparallel Voice Conversion Based on Cycle-consistent Adversarial Network
- Lauri Juvela, Bajibabu Bollepalli, Xin Wang, Hirokazu Kameoka, Manu Airaksinen, Junichi Yamagishi, Paavo Alku, Speech Waveform Synthesis from MFCC Sequences with Generative Adversarial Networks
- Zhong Meng, Jinyu Li, Yifan Gong, Biing-Hwang (Fred) Juang, Adversarial Teacher-Student Learning for Unsupervised Domain Adaptation
- Zhong Meng, Jinyu Li, Zhuo Chen, Yong Zhao, Vadim Mazalov, Yifan Gong, Biing-Hwang (Fred) Juang, Speaker-Invariant Training via Adversarial Learning
- Sen Li, Stephane Villette, Pravin Ramadas, Daniel Sinder, Speech Bandwidth Extension using Generative Adversarial Networks
- Qing Wang, Wei Rao, Sining Sun, Lei Xie, Eng Siong Chng, Haizhou Li, Unsupervised Domain Adaptation via Domain Adversarial Training for Speaker Recognition
- Hu Hu, Tian Tan, Yanmin Qian, Generative Adversarial Networks Based Data Augmentation for Noise Robust Speech Recognition
- Yuki Saito, Shinnosuke Takamichi, Hiroshi Saruwatari, Text-to-speech Synthesis using STFT Spectra Based on Low-/multi-resolution Generative Adversarial Networks
- Jiangyan Yi, Jianhua Tao, Zhengqi Wen, Ye Bai, Adversarial Multilingual Training for Low-resource Speech Recognition
- Meet H. Soni, Neil Shah, Hemant A. Patil, Time-frequency Masking-based Speech Enhancement using Generative Adversarial Network
- Taira Tsuchiya, Naohiro Tawara, Tetsuji Ogawa, Tetsunori Kobayashi, Speaker Invariant Feature Extraction for Zero-resource Languages with Adversarial Learning

GANs in ICASSP 2018

- Jing Han, Zixing Zhang, Zhao Ren, Fabien Ringeval, Bjoern Schuller, Towards Conditional Adversarial Training for Predicting Emotions from Speech
- Chenxing Li, Lei Zhu, Shuang Xu, Peng Gao, Bo Xu, CBLDNN-based Speaker-independent Speech Separation via Generative Adversarial Training
- Anuroop Sriram, Heewoo Jun, Yashesh Gaur, Sanjeev Satheesh, Robust Speech Recognition using Generative Adversarial Networks
- Cem Subakan, Paris Smaragdis, Generative Adversarial Source Separation,
- Ashutosh Pandey, Deliang Wang, On Adversarial Training and Loss Functions for Speech Enhancement
- Bin Liu, Shuai Nie, Yaping Zhang, Dengfeng Ke, Shan Liang, Wenju Liu, Boosting Noise Robustness of Acoustic Model via Deep Adversarial Training
- Yang Gao, Rita Singh, Bhiksha Raj, Voice Impersonation using Generative Adversarial Networks
- Aditay Tripathi, Aanchan Mohan, Saket Anand, Maneesh Singh, Adversarial Learning of Raw Speech Features for Domain Invariant Speech Recognition
- Zhe-Cheng Fan, Yen-Lin Lai, Jyh-Shing Jang, SVSGAN: Singing Voice Separation via Generative Adversarial Network
- Santiago Pascual, Maruchan Park, Joan Serra, Antonio Bonafonte, Kang-Hun Ahn, Language and Noise Transfer in Speech Enhancement Generative Adversarial Network

A promising research direction and still has room for further improvements in the speech signal processing domain

Thank You Very Much

Tsao, Yu Ph.D.
yu.tsao@citi.sinica.edu.tw
https://www.citi.sinica.edu.tw/pages/yu.tsao/contact_zh.html

Generative Adversarial Network

and its Applications to Signal Processing
and Natural Language Processing

Part III: Natural Language Processing

Outline of Part III

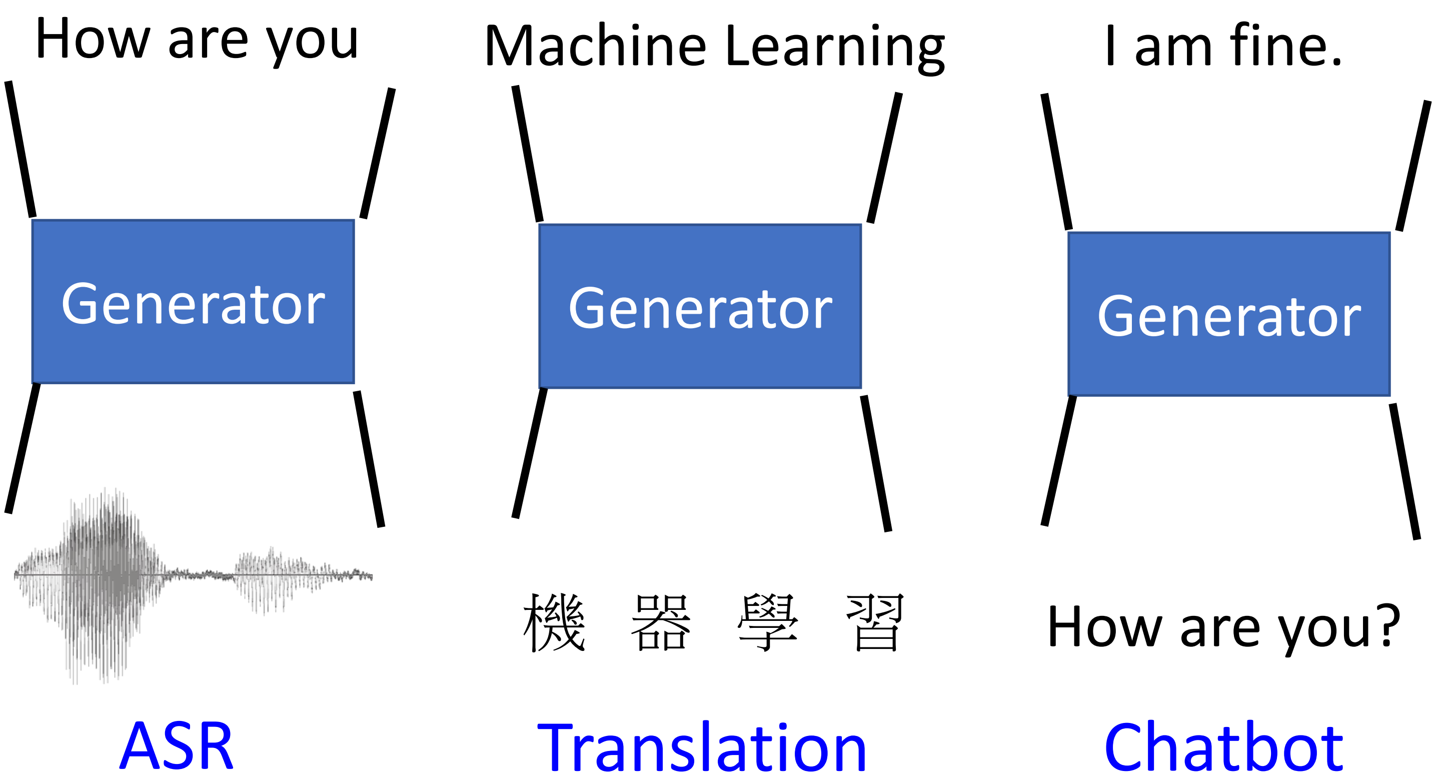
Conditional Sequence Generation

- RL (human feedback)
- GAN (discriminator feedback)

Unsupervised Conditional Sequence Generation

- Text Style Transfer
- Unsupervised Abstractive Summarization
- Unsupervised Translation

Conditional Sequence Generation



The generator is a typical seq2seq model.

With GAN, you can train seq2seq model in another way.

Review: Sequence-to-sequence

- Chat-bot as example

Output:	Not bad	I'm John.
Human	better	
Training Criterion		better

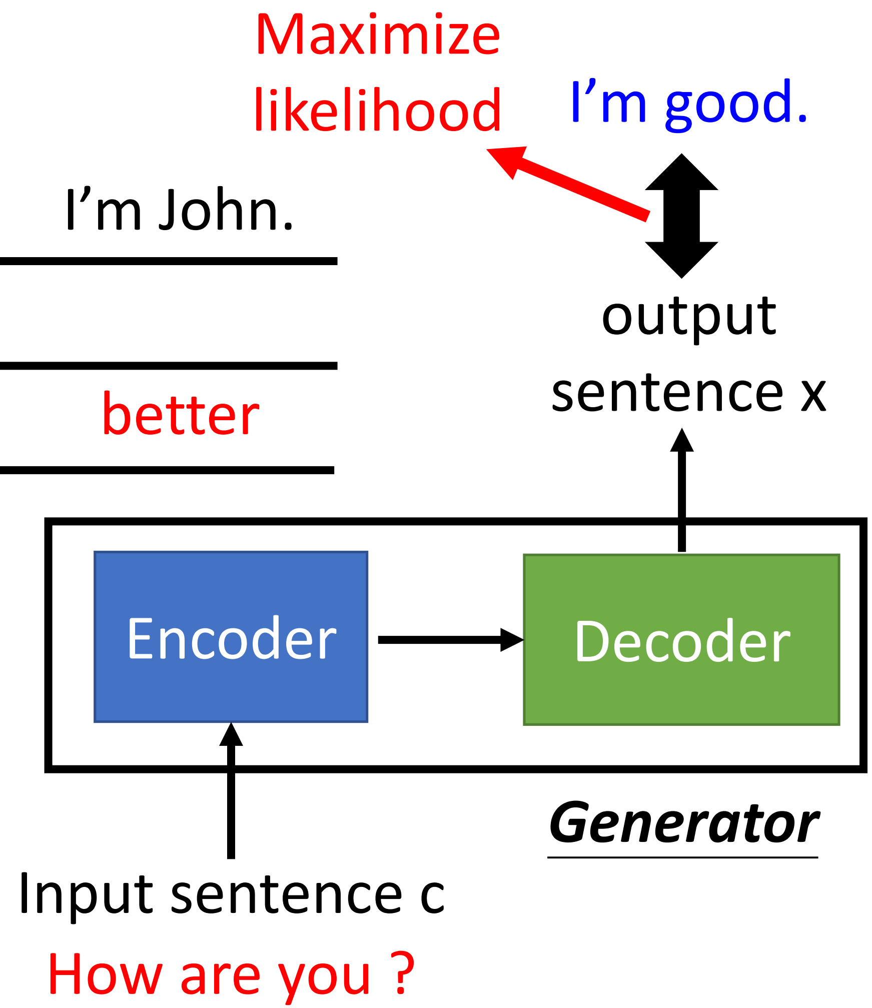
Training data:

⋮

A: How are you ?

B: I'm good.

⋮



Outline of Part III

Improving Supervised Seq-to-seq Model

- RL (human feedback)
- GAN (discriminator feedback)

Unsupervised Seq-to-seq Model

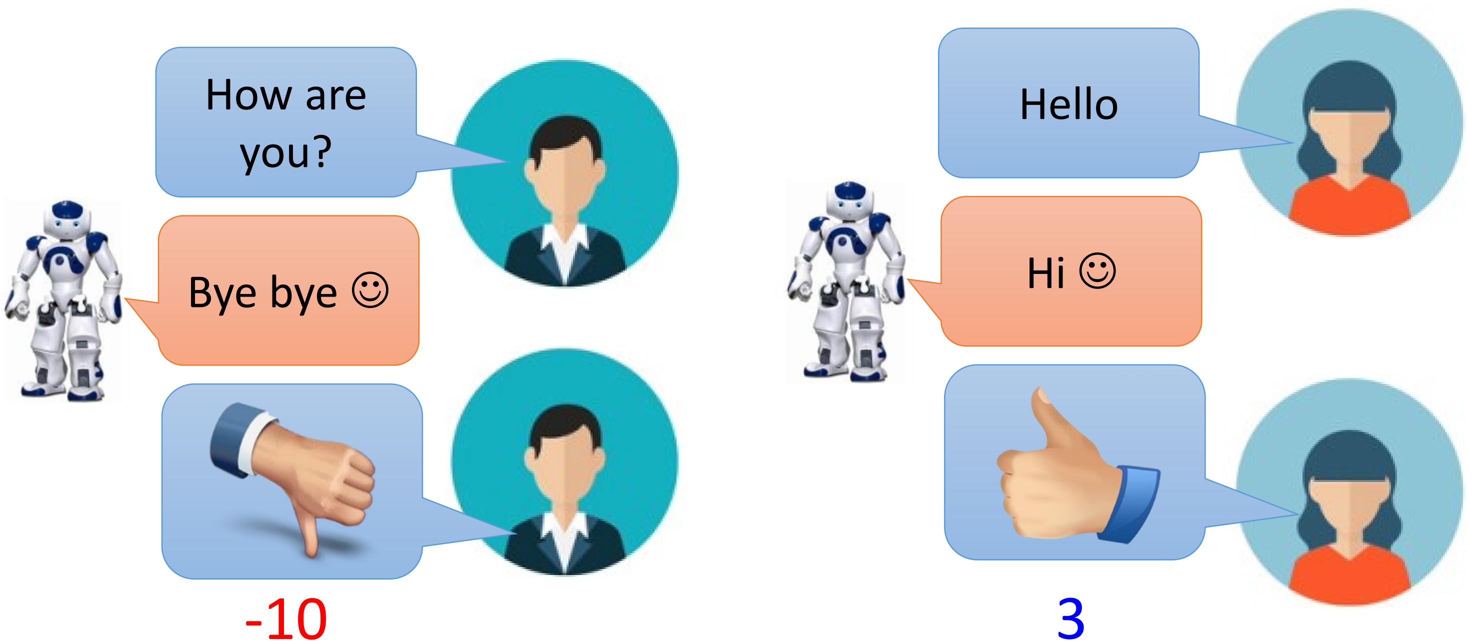
- Text Style Transfer
- Unsupervised Abstractive Summarization
- Unsupervised Translation

Introduction

https://image.freepik.com/free-vector/variety-of-human-avatars_23-2147506285.jpg

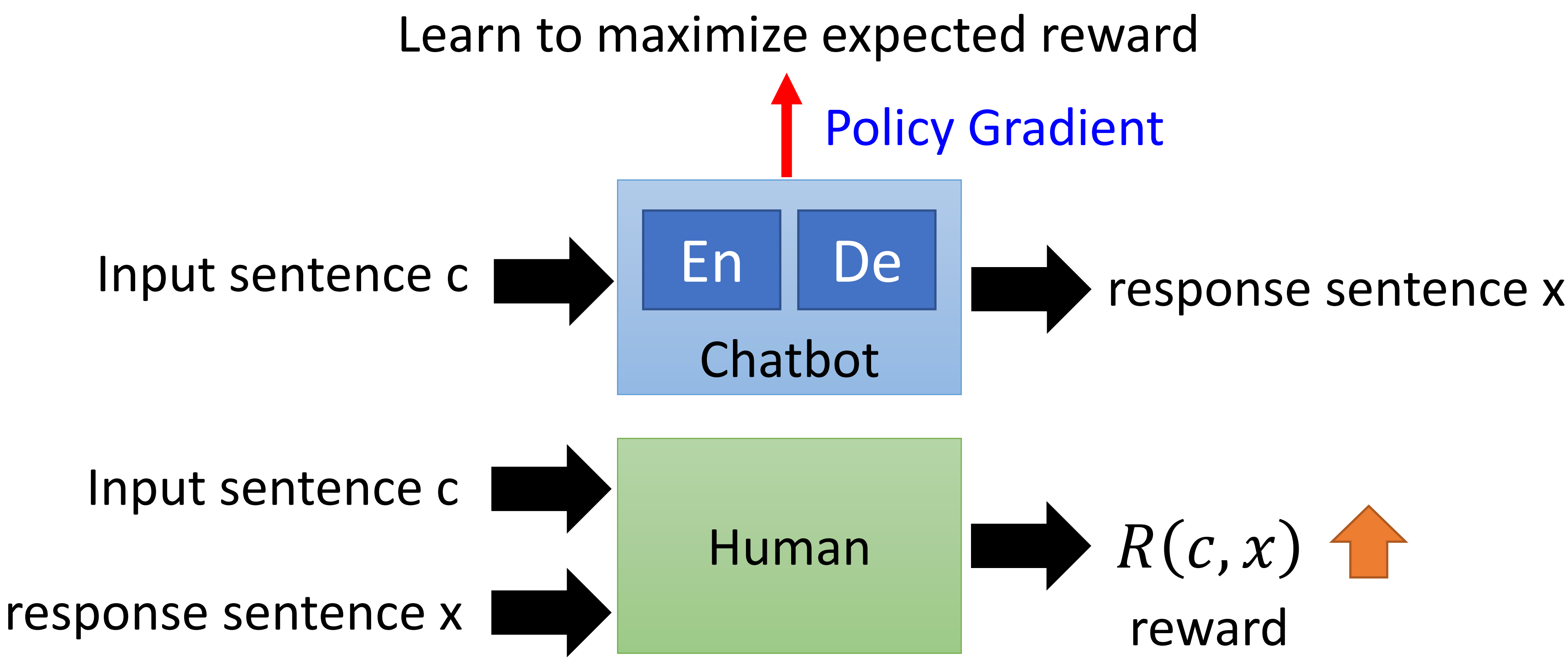
http://www.freepik.com/free-vector/variety-of-human-avatars_766615.htm

- Machine obtains feedback from user



- Chat-bot learns to maximize the ***expected reward***

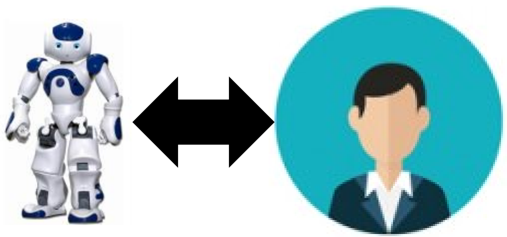
Maximizing Expected Reward



[Li, et al., EMNLP, 2016]

Policy Gradient - Implemenation

θ^t



(c^1, x^1) $\quad R(c^1, x^1)$

(c^2, x^2) $\quad R(c^2, x^2)$

$\vdots \qquad \vdots$

(c^N, x^N) $\quad R(c^N, x^N)$

$$\theta^{t+1} \leftarrow \theta^t + \eta \boxed{\nabla \bar{R}_{\theta^t}}$$

$$\frac{1}{N}\sum_{i=1}^{N} R\left(c^i, x^i\right) \nabla log P_{\theta^t}\left(x^i | c^i\right)$$

$R\left(c^i, x^i\right)$ is positive

Updating θ to increase $P_\theta\left(x^i | c^i\right)$

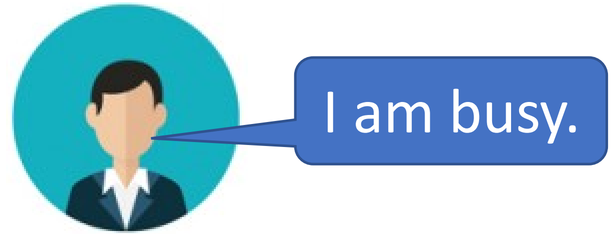
$R\left(c^i, x^i\right)$ is negative

Updating θ to decrease $P_\theta\left(x^i | c^i\right)$

Comparison

	Maximum Likelihood	Reinforcement Learning
Objective Function	$\frac{1}{N}\sum_{i=1}^{N} logP_\theta(\hat{x}^i \mid c^i)$	$\frac{1}{N}\sum_{i=1}^{N} R(c^i, x^i) logP_\theta(x^i \mid c^i)$
Gradient	$\frac{1}{N}\sum_{i=1}^{N} \nabla logP_\theta(\hat{x}^i \mid c^i)$	$\frac{1}{N}\sum_{i=1}^{N} R(c^i, x^i) \nabla logP_\theta(x^i \mid c^i)$
Training Data	$\{(c^1, \hat{x}^1), \ldots, (c^N, \hat{x}^N)\}$ $R(c^i, \hat{x}^i) = 1$	$\{(c^1, x^1), \ldots, (c^N, x^N)\}$ obtained from interaction weighted by $R(c^i, x^i)$

Outline of Part III



Improving Supervised Seq-to-seq Model

- RL (human feedback)
- GAN (discriminator feedback)

Unsupervised Seq-to-seq Model

- Text Style Transfer
- Unsupervised Abstractive Summarization
- Unsupervised Translation

http://www.nipic.com/show/3/83/3936650kd7476069.html

Conditional GAN

Input sentence c

En De

Chatbot

response sentence x

Input sentence c

response sentence x

Discriminator

Real or fake

"reward"

human dialogues

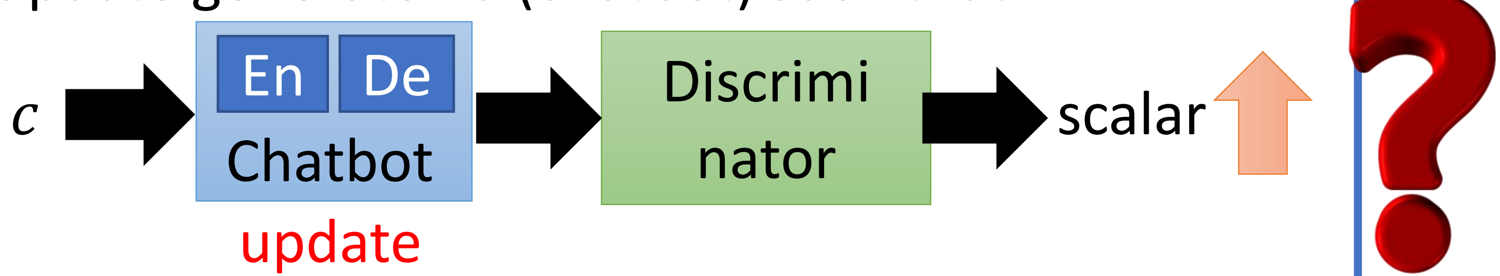
[Li, et al., EMNLP, 2017]

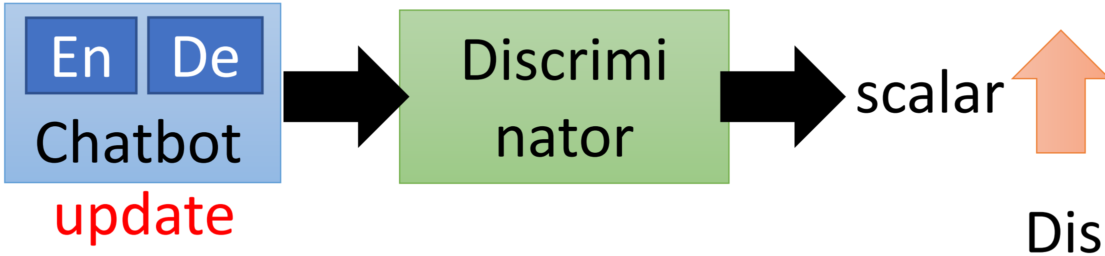
Algorithm

Training data:

Pairs of conditional input c and response x

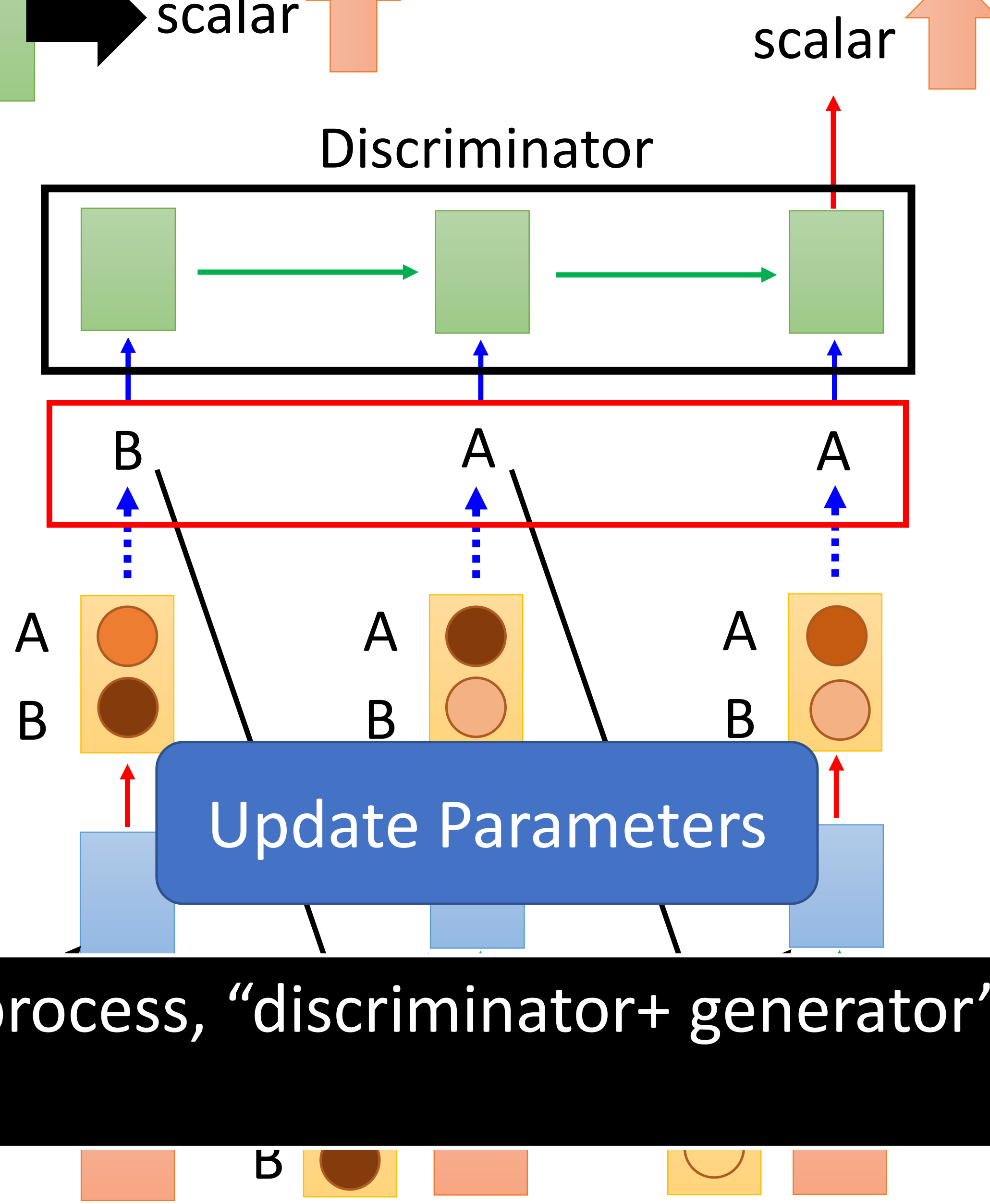
- Initialize generator G (chatbot) and discriminator D
- In each iteration:
  - Sample input c and response x from training set
  - Sample input c' from training set, and generate response $\tilde{x}$ by G(c')
  - Update D to increase $D(c, x)$ and decrease $D(c', \tilde{x})$
  - Update generator G (chatbot) such that





Can we use gradient ascent?

NO!



Due to the sampling process, “discriminator+ generator” is not differentiable

Three Categories of Solutions

Gumbel-softmax

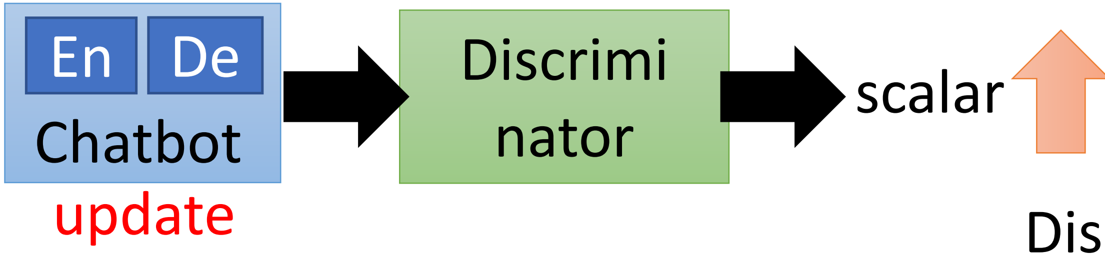
- [Matt J. Kusner, et al, arXiv, 2016]

Continuous Input for Discriminator

- [Sai Rajeswar, et al., arXiv, 2017][Ofir Press, et al., ICML workshop, 2017][Zhen Xu, et al., EMNLP, 2017][Alex Lamb, et al., NIPS, 2016][Yizhe Zhang, et al., ICML, 2017]

“Reinforcement Learning”

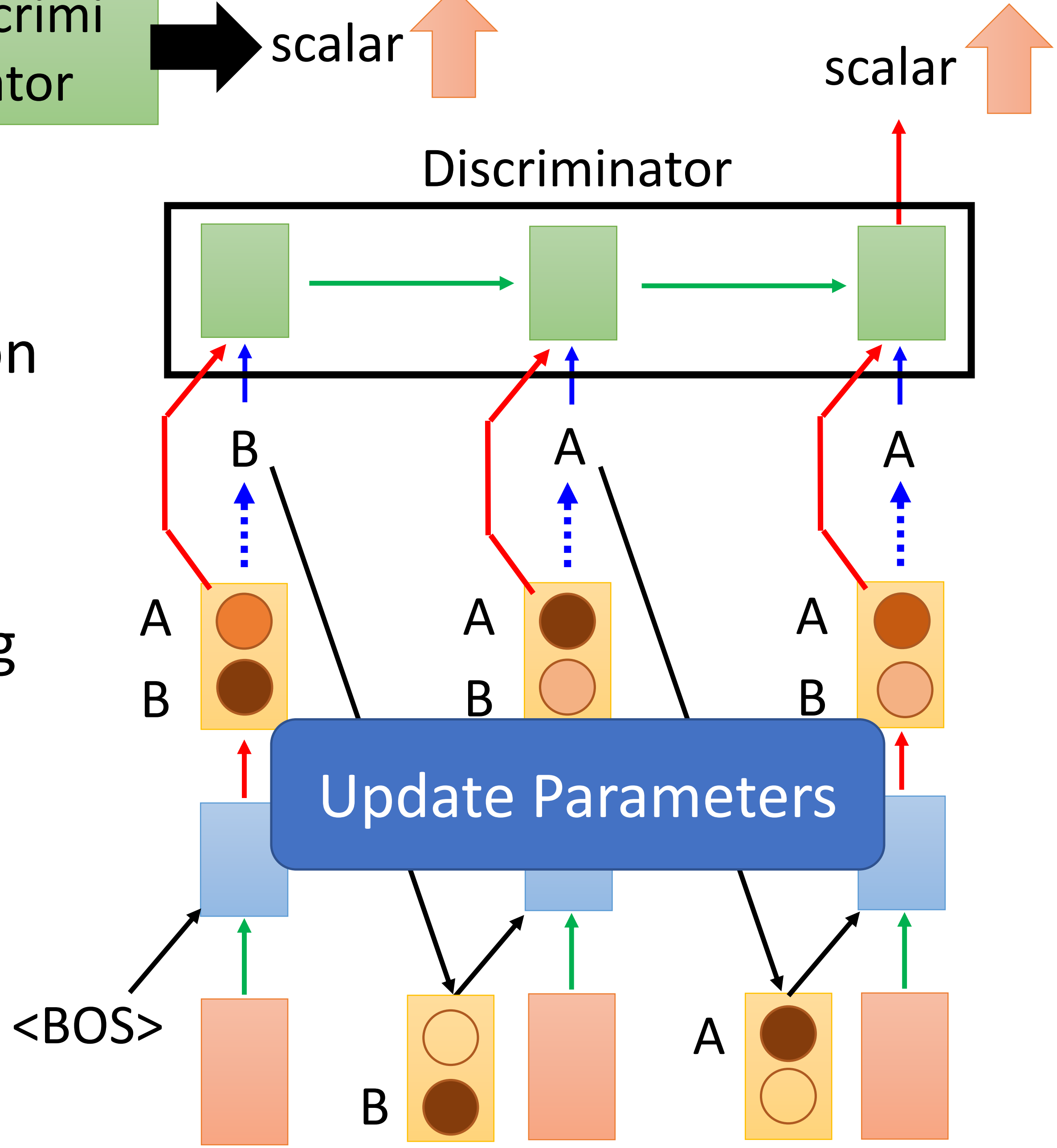
- [Yu, et al., AAAI, 2017][Li, et al., EMNLP, 2017][Tong Che, et al, arXiv, 2017][Jiaxian Guo, et al., AAAI, 2018][Kevin Lin, et al, NIPS, 2017][William Fedus, et al., ICLR, 2018]



Use the distribution as the input of discriminator

Avoid the sampling process

We can do backpropagation now.



What is the problem?

- Real sentence

1	0	0	0	0
0	1	0	0	0
0	0	1	0	0
0	0	0	1	0
0	0	0	0	1

- Generated

0.9	0.1	0.1	0	0
0.1	0.9	0.1	0	0
0	0	0.7	0.1	0
0	0	0.1	0.8	0.1
0	0	0	0.1	0.9

Can never be 1-of-N

Discriminator can immediately find the difference.

WGAN is helpful

Three Categories of Solutions

Gumbel-softmax

- [Matt J. Kusner, et al, arXiv, 2016]

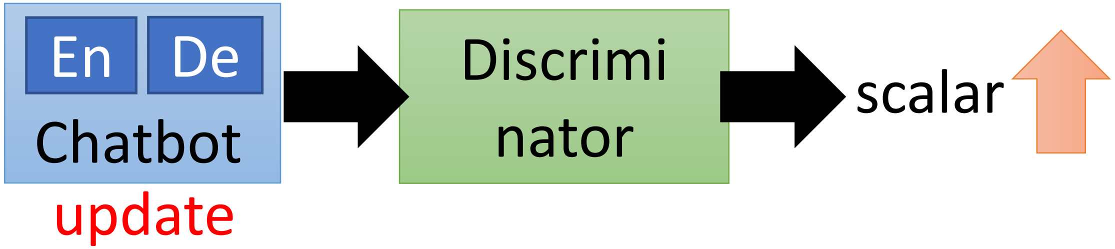
Continuous Input for Discriminator

- [Sai Rajeswar, et al., arXiv, 2017][Ofir Press, et al., ICML workshop, 2017][Zhen Xu, et al., EMNLP, 2017][Alex Lamb, et al., NIPS, 2016][Yizhe Zhang, et al., ICML, 2017]

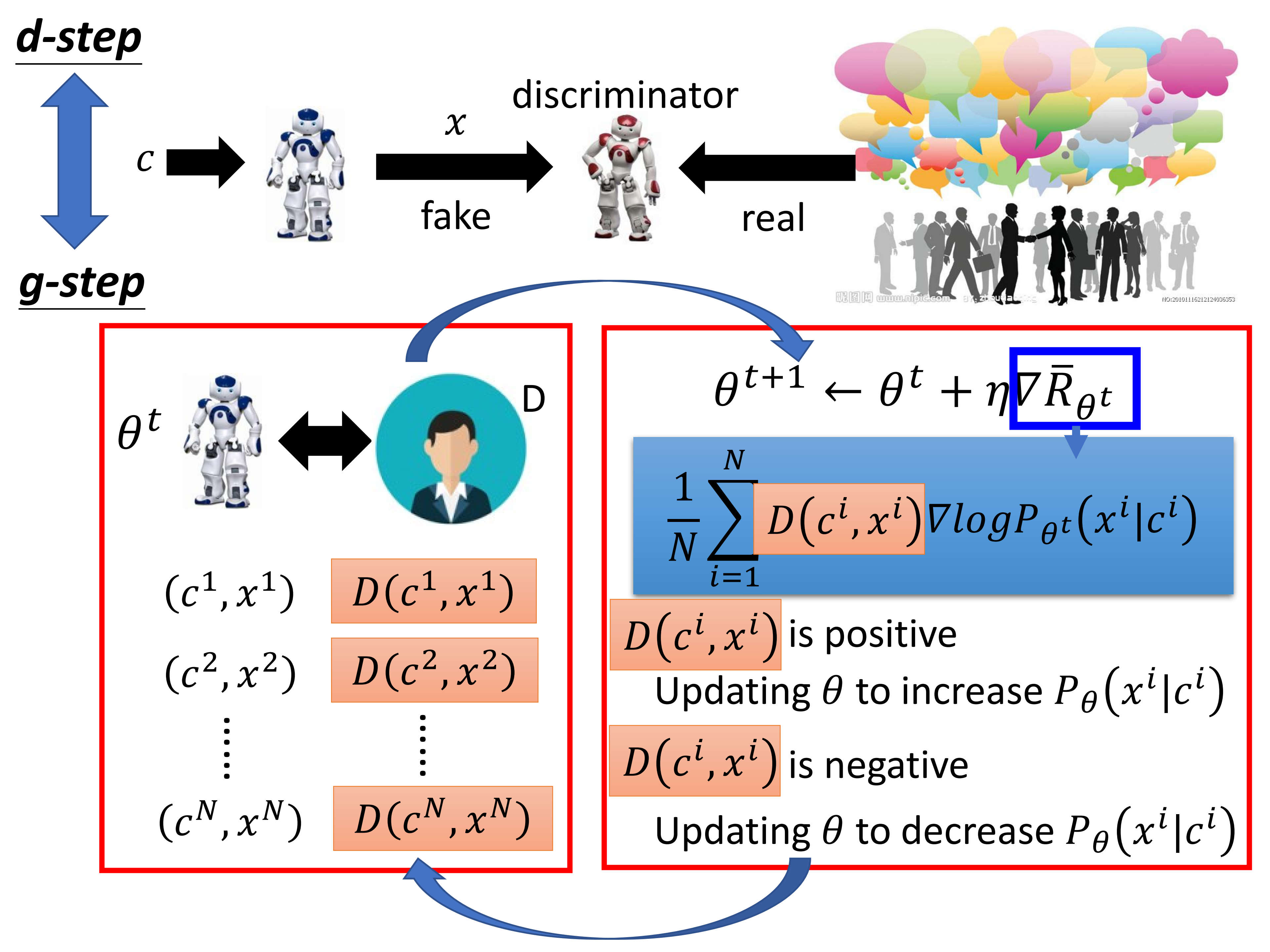
"Reinforcement Learning"

- [Yu, et al., AAAI, 2017][Li, et al., EMNLP, 2017][Tong Che, et al, arXiv, 2017][Jiaxian Guo, et al., AAAI, 2018][Kevin Lin, et al, NIPS, 2017][William Fedus, et al., ICLR, 2018]

Reinforcement Learning?



- Consider the output of discriminator as reward
  - Update generator to increase discriminator = to get maximum reward
  - Using the formulation of policy gradient, replace reward $R(c,x)$ with discriminator output $\mathrm{D}(c,x)$
- Different from typical RL
  - The discriminator would update



d-step
g-step
c
x
discriminator
fake
real
θ^t
D
(c^1, x^1)
D(c^1, x^1)
(c^2, x^2)
D(c^2, x^2)
(c^N, x^N)
D(c^N, x^N)
θ^{t+1} ← θ^t + η∇R̄_{θ^t}
(1/N) Σ_{i=1}^{N} D(c^i, x^i) ∇logP_{θ^t}(x^i|c^i)
D(c^i, x^i) is positive
Updating θ to increase P_θ(x^i|c^i)
D(c^i, x^i) is negative
Updating θ to decrease P_θ(x^i|c^i)

Reward for Every Generation Step

$$\nabla \bar{R}_\theta \approx \frac{1}{N}\sum_{i=1}^{N} D(c^i, x^i)\nabla log P_\theta(x^i|c^i)$$

c^i = "What is your name?" $D(c^i, x^i)$ is negative

x^i = "I don't know" Update θ to decrease log$P_\theta(x^i|c^i)$

$$log P_\theta(x^i|c^i) = log P(x_1^i|c^i) + log P(x_2^i|c^i, x_1^i) + log P(x_3^i|c^i, x_{1:2}^i)$$

$P("I"|c^i)$ ↓ ? ↓ ↓

c^i = "What is your name?" $D(c^i, x^i)$ is positive

x^i = "I am John" Update θ to increase log$P_\theta(x^i|c^i)$

$$log P_\theta(x^i|c^i) = log P(x_1^i|c^i) + log P(x_2^i|c^i, x_1^i) + log P(x_3^i|c^i, x_{1:2}^i)$$

$P("I"|c^i)$ ↑ ↑ ↑

Reward for Every Generation Step

h^i = "What is your name?" x^i = "I don't know"

$$logP_\theta(x^i|h^i) = \underline{logP(x_1^i|c^i)} + \underline{logP(x_2^i|c^i, x_1^i)} + \underline{logP(x_3^i|c^i, x_{1:2}^i)}$$

$$P("I"|c^i) \quad P("don't"|c^i, "I") \quad P("know"|c^i, "I\ don't")$$

$$\nabla \bar{R}_\theta \approx \frac{1}{N}\sum_{i=1}^{N} \underline{D(c^i, x^i)}\,\underline{\nabla logP_\theta(x^i|c^i)}$$

$$\nabla \bar{R}_\theta \approx \frac{1}{N}\sum_{i=1}^{N}\sum_{\underline{t=1}}^{T} (\underline{Q(c^i, x_{1:t}^i) - b})\,\underline{\nabla logP_\theta(x_t^i|c^i, x_{1:t-1}^i)}$$

Method 1. Monte Carlo (MC) Search [Yu, et al., AAAI, 2017]

Method 2. Discriminator For Partially Decoded Sequences

[Li, et al., EMNLP, 2017]

Experimental Results

Input	We've got to look for another route.
MLE	I'm sorry.
GAN	You're not going to be here for a while.

Input	You can save him by talking.
MLE	I don't know.
GAN	You know what's going on in there, you know what I mean?

- MLE frequently generates "I'm sorry", "I don't know", etc. (corresponding to fuzzy images?)
- GAN generates longer and more complex responses (however, no strong evidence shows that they are better)

Find more comparison in the survey papers.

[Lu, et al., arXiv, 2018][Zhu, et al., arXiv, 2018]

More Applications

- Supervised machine translation [Wu, et al., arXiv 2017][Yang, et al., arXiv 2017]
- Supervised abstractive summarization [Liu, et al., AAAI 2018]
- Image/video caption generation [Rakshith Shetty, et al., ICCV 2017][Liang, et al., arXiv 2017]

If you are using seq2seq models,
consider to improve them by GAN.

Outline of Part III

Conditional Sequence Generation

- RL (human feedback)
- GAN (discriminator feedback)

Unsupervised Conditional Sequence Generation

- Text Style Transfer
- Unsupervised Abstractive Summarization
- Unsupervised Translation

Text Style Transfer

Domain X

Domain Y

male

female

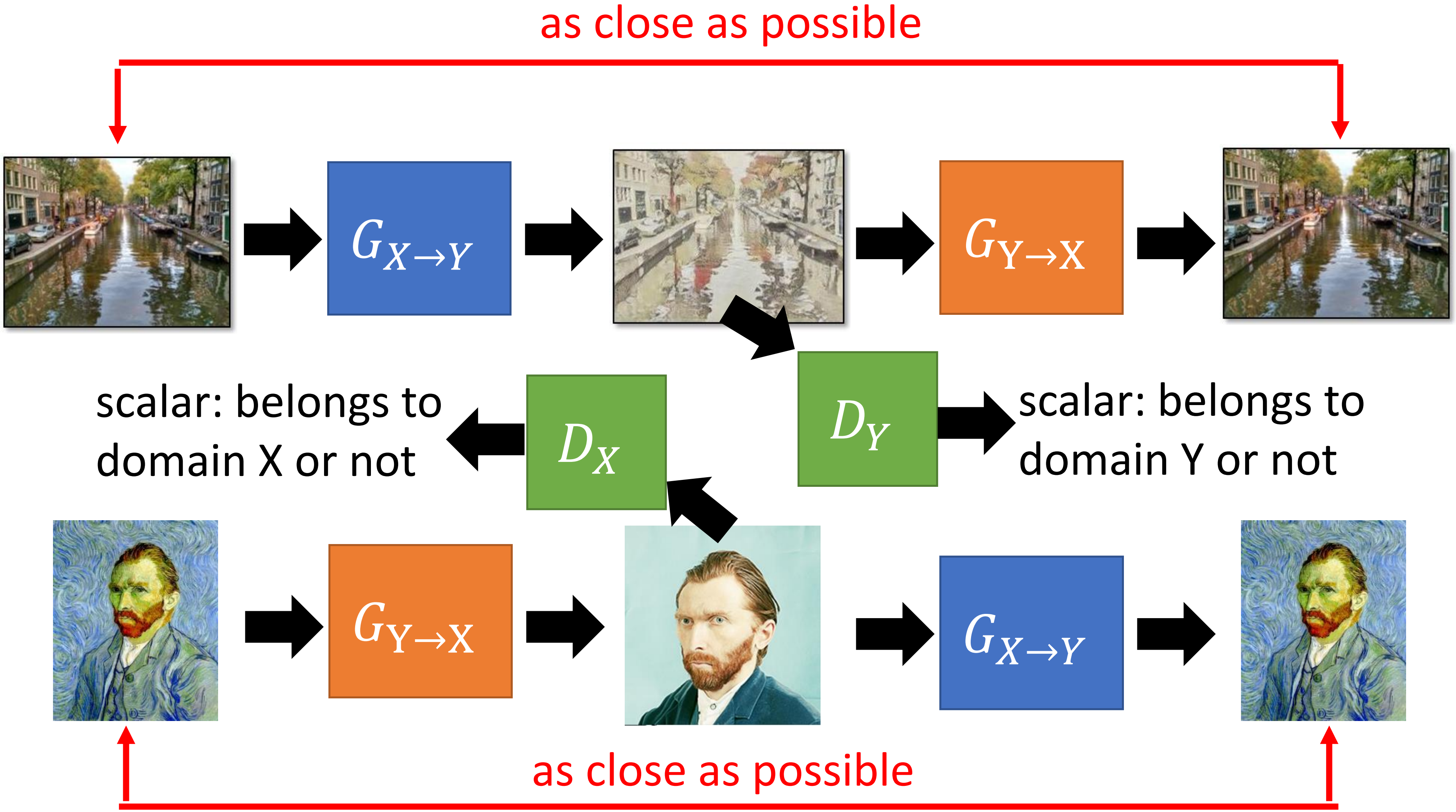
It is good.
It's a good day.
I love you.

positive sentences

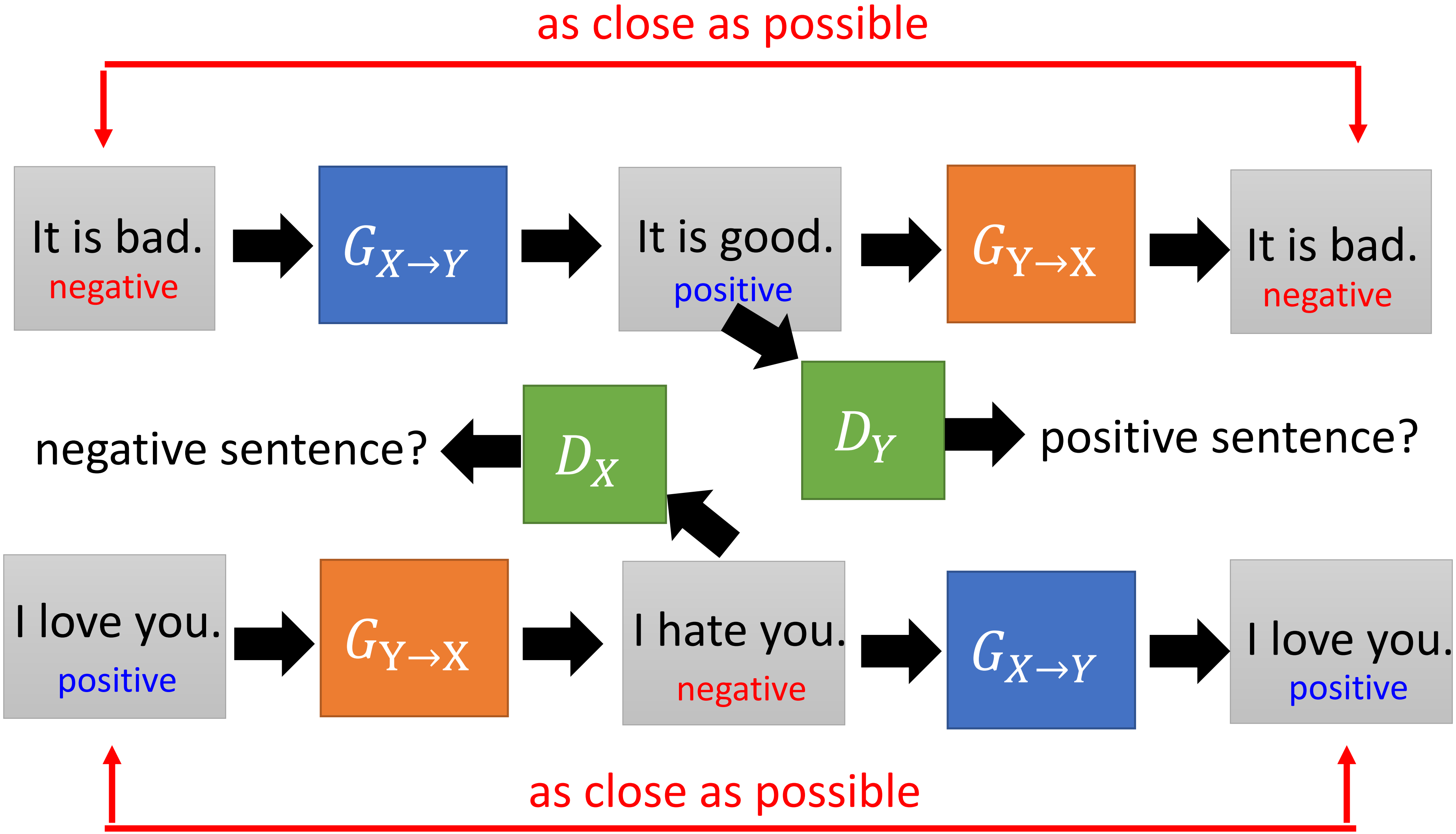
It is bad.
It's a bad day.
I don't love you.

negative sentences

Direct Transformation



Direct Transformation

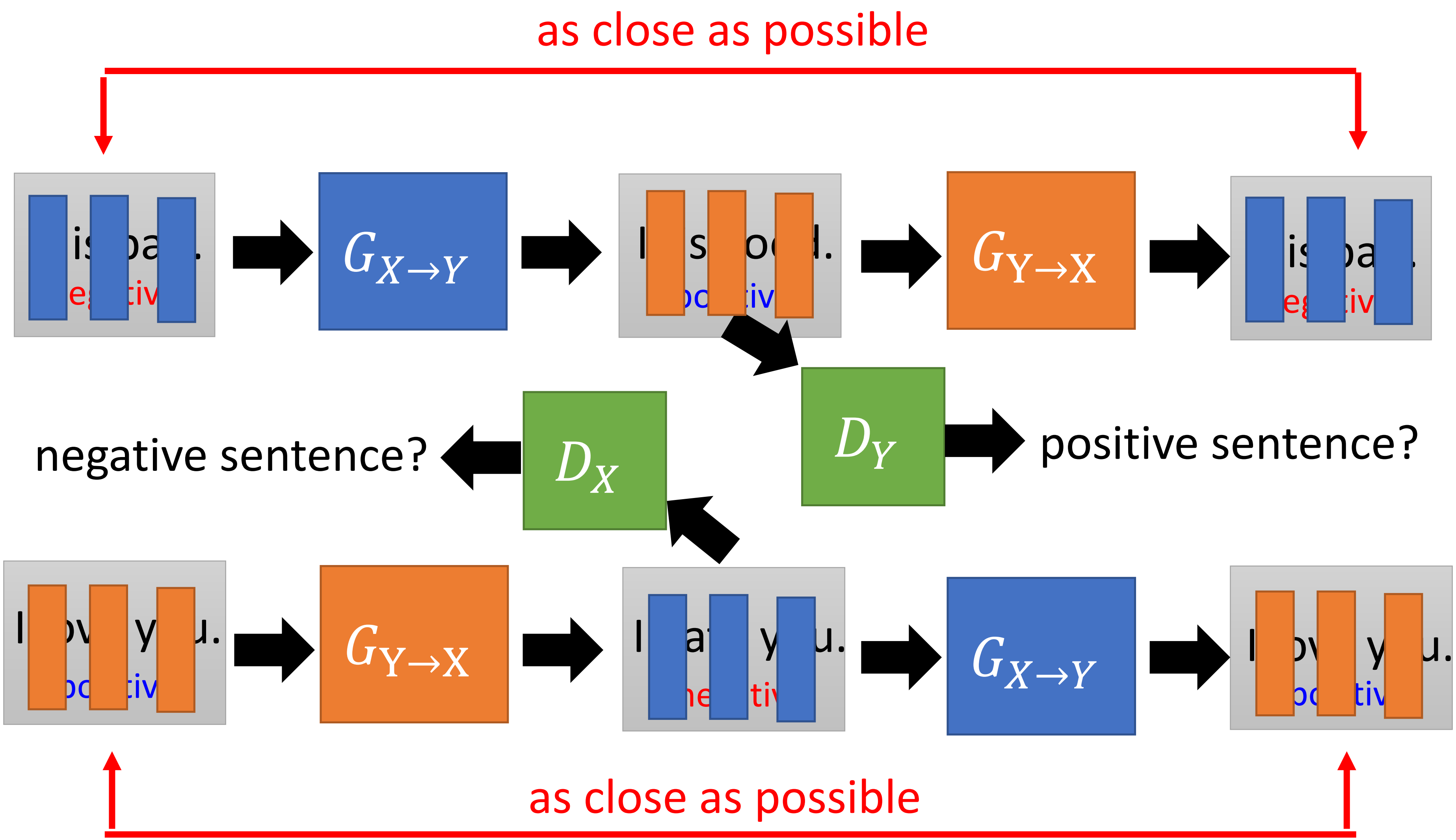


Direct Transformation

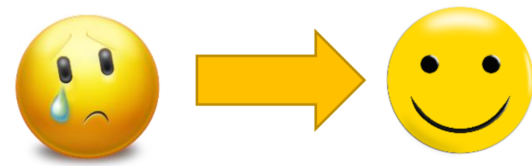
Discrete?

Word embedding

[Lee, et al., ICASSP, 2018]

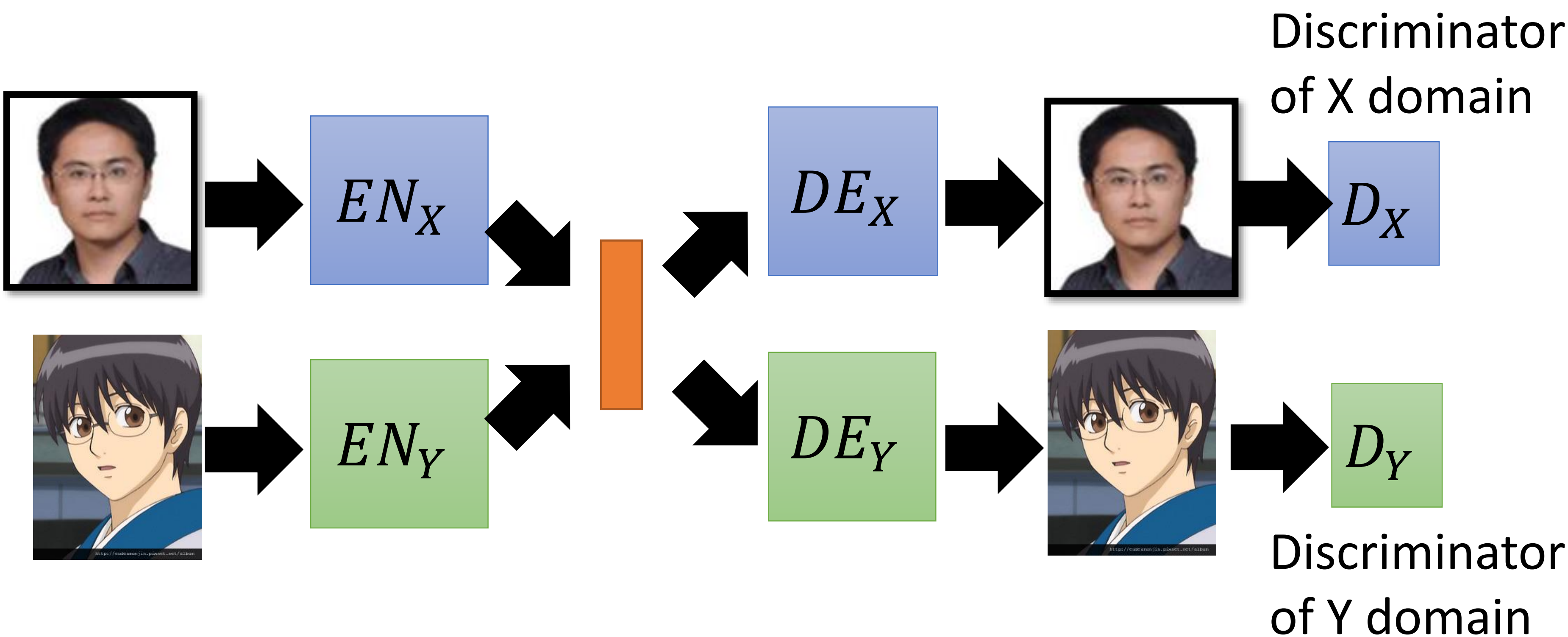


- Negative sentence to positive sentence:
  it's a crappy day → it's a great day
  i wish you could be here → you could be here
  it's not a good idea → it's good idea
  i miss you → i love you
  i don't love you → i love you
  i can't do that → i can do that
  i feel so sad → i happy
  it's a bad day → it's a good day
  it's a dummy day → it's a great day
  sorry for doing such a horrible thing → thanks for doing a great thing
  my doggy is sick → my doggy is my doggy
  my little doggy is sick → my little doggy is my little doggy

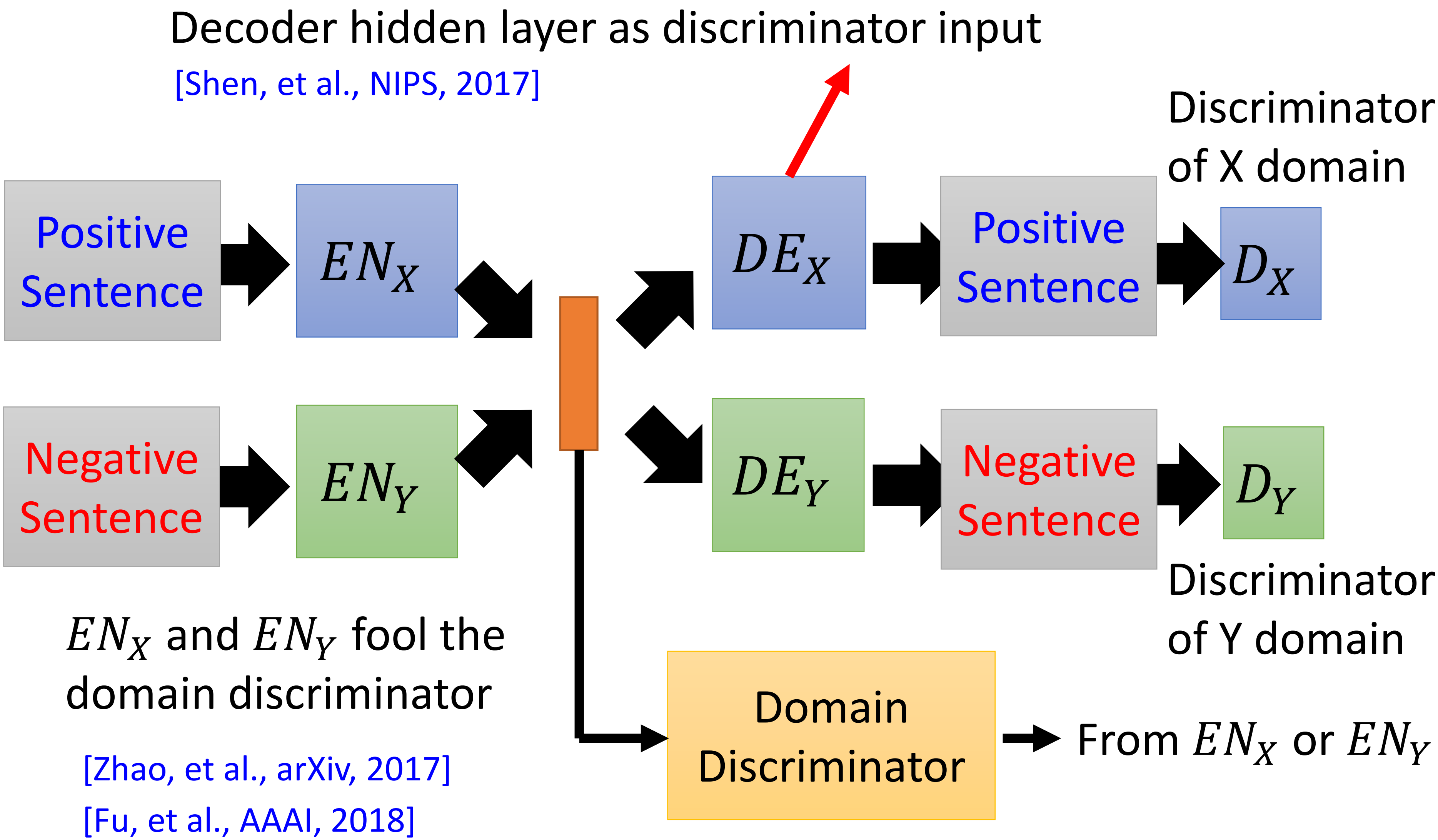


Title: SCALABLE SENTIMENT FOR SEQUENCE-TO-SEQUENCE CHATBOT RESPONSE WITH PERFORMANCE ANALYSIS
Session: Dialog Systems and Applications
Time: Wednesday, April 18, 08:30 - 10:30
Authors: Chih-Wei Lee, Yau-Shian Wang, Tsung-Yuan Hsu, Kuan-Yu Chen, Hung-Yi Lee, Lin-Shan Lee

Projection to Common Space



Projection to Common Space



Outline of Part III

Improving Supervised Seq-to-seq Model

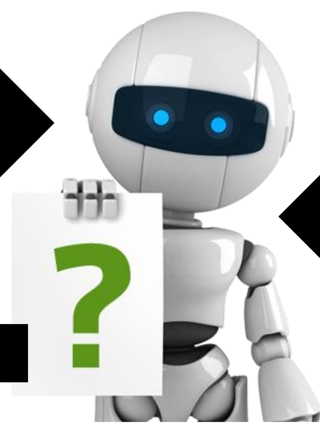
- RL (human feedback)
- GAN (discriminator feedback)

Unsupervised Seq-to-seq Model

- Text Style Transfer
- Unsupervised Abstractive Summarization
- Unsupervised Translation

Abstractive Summarization

- Now machine can do **abstractive summary** by seq2seq (write summaries in its own words)



summary
(in its own words)

seq2seq

summary 1

summary 2

summary 3

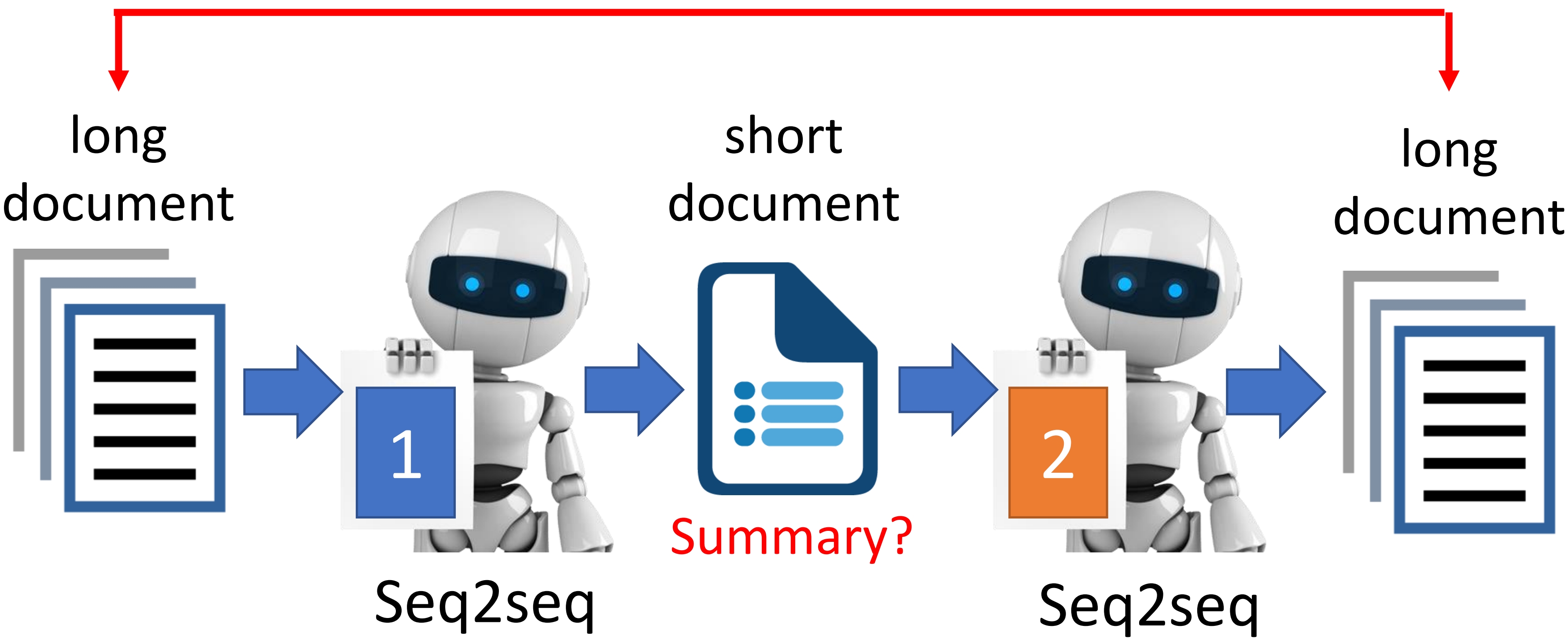
Supervised: We need lots of labelled training data.

Training Data

Unsupervised Abstractive Summarization

Only need a lot of documents to train the model

The two seq2seq models are jointly learn to minimize the reconstruction error.

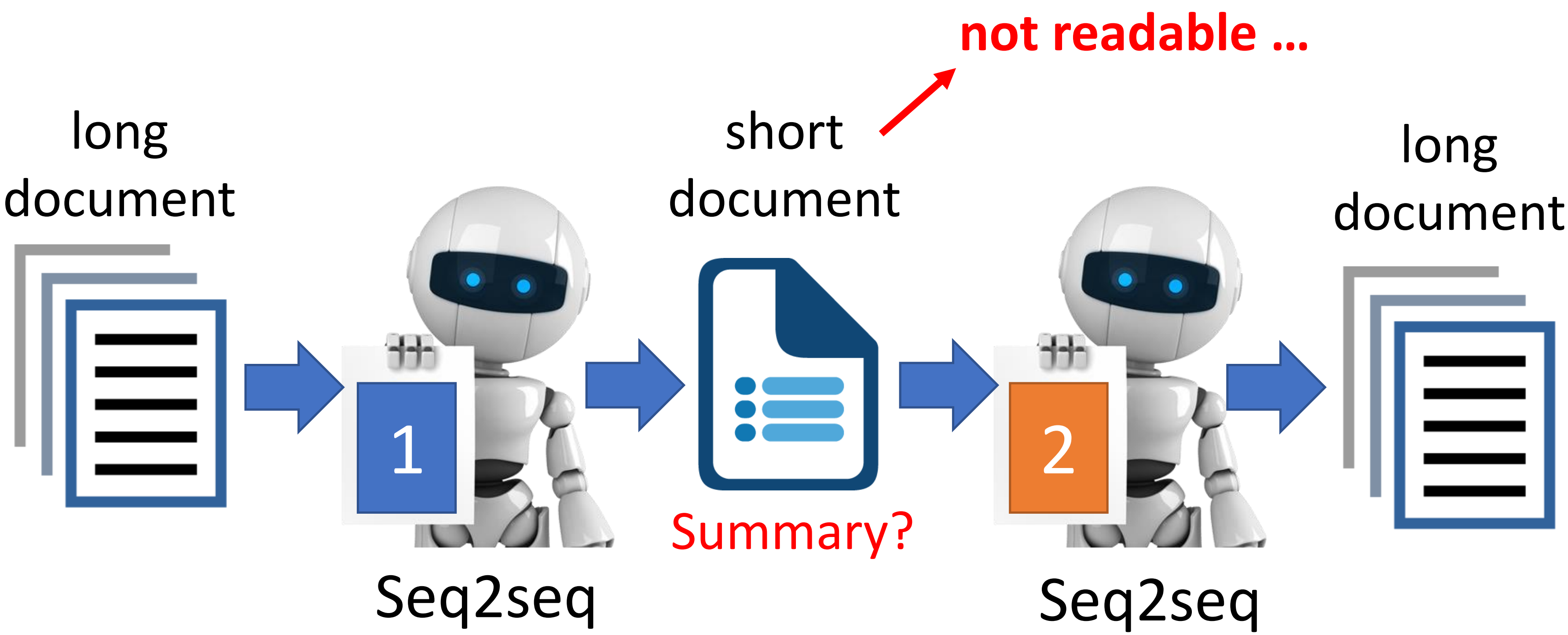


Unsupervised Abstractive Summarization

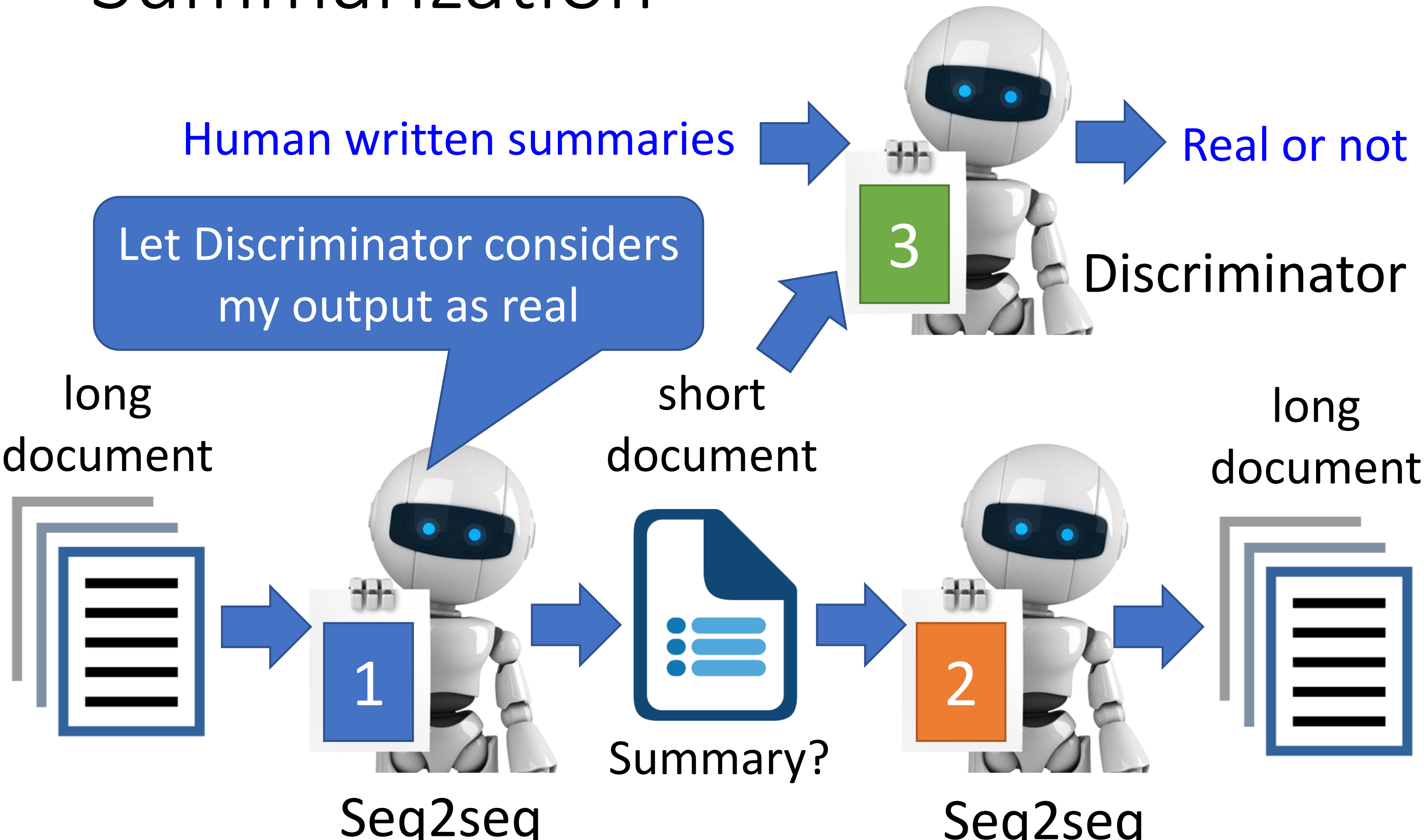
This is a ***seq2seq2seq auto-encoder***.

Using a sequence of words as latent representation.

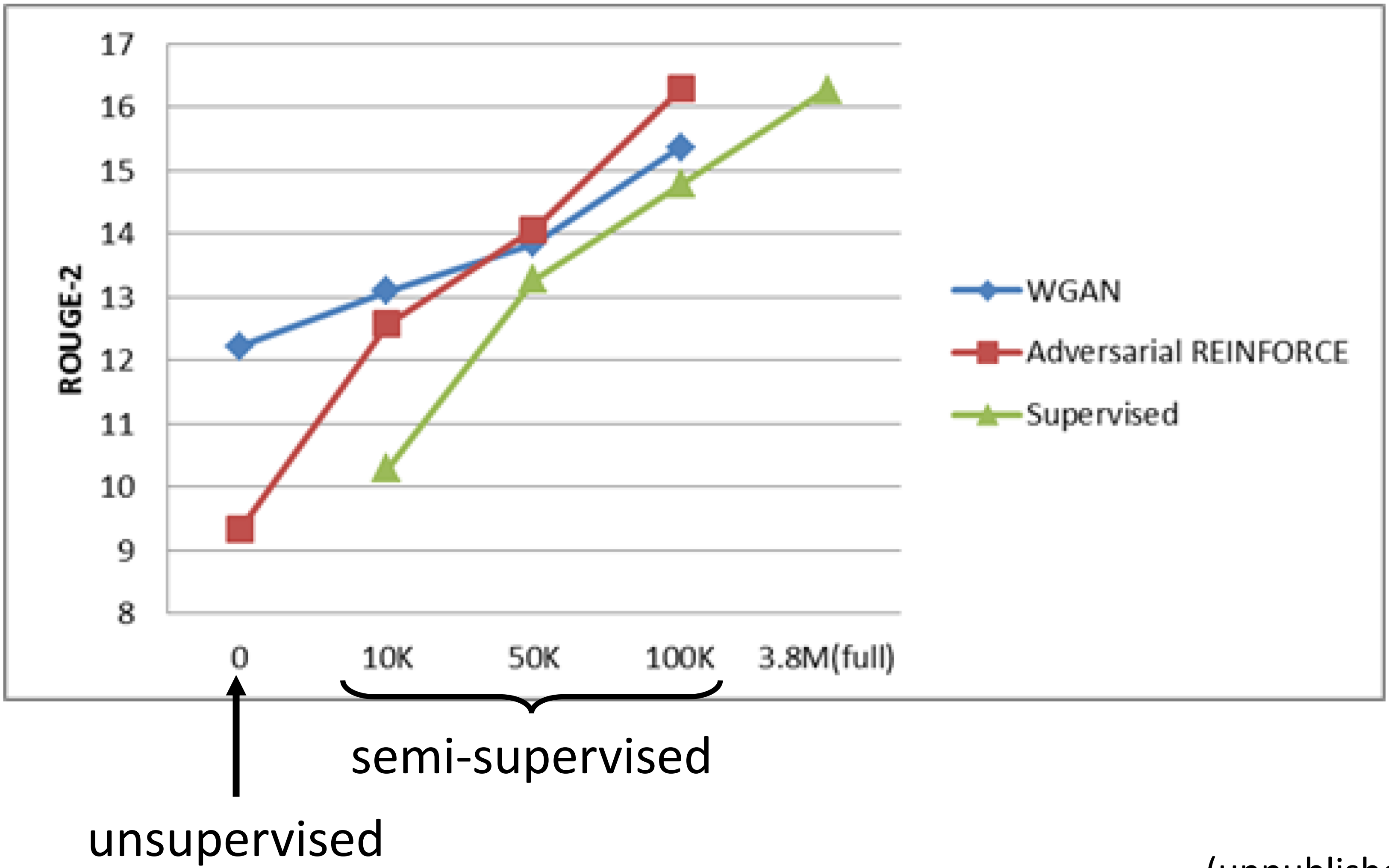
Policy gradient is used.



Unsupervised Abstractive Summarization



Semi-supervised Learning



(unpublished)

Outline of Part III

Improving Supervised Seq-to-seq Model

- RL (human feedback)
- GAN (discriminator feedback)

Unsupervised Seq-to-seq Model

- Text Style Transfer
- Unsupervised Abstractive Summarization
- Unsupervised Translation

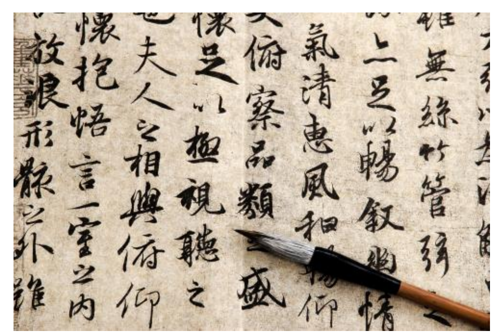
Unsupervised Machine Translation

Domain X

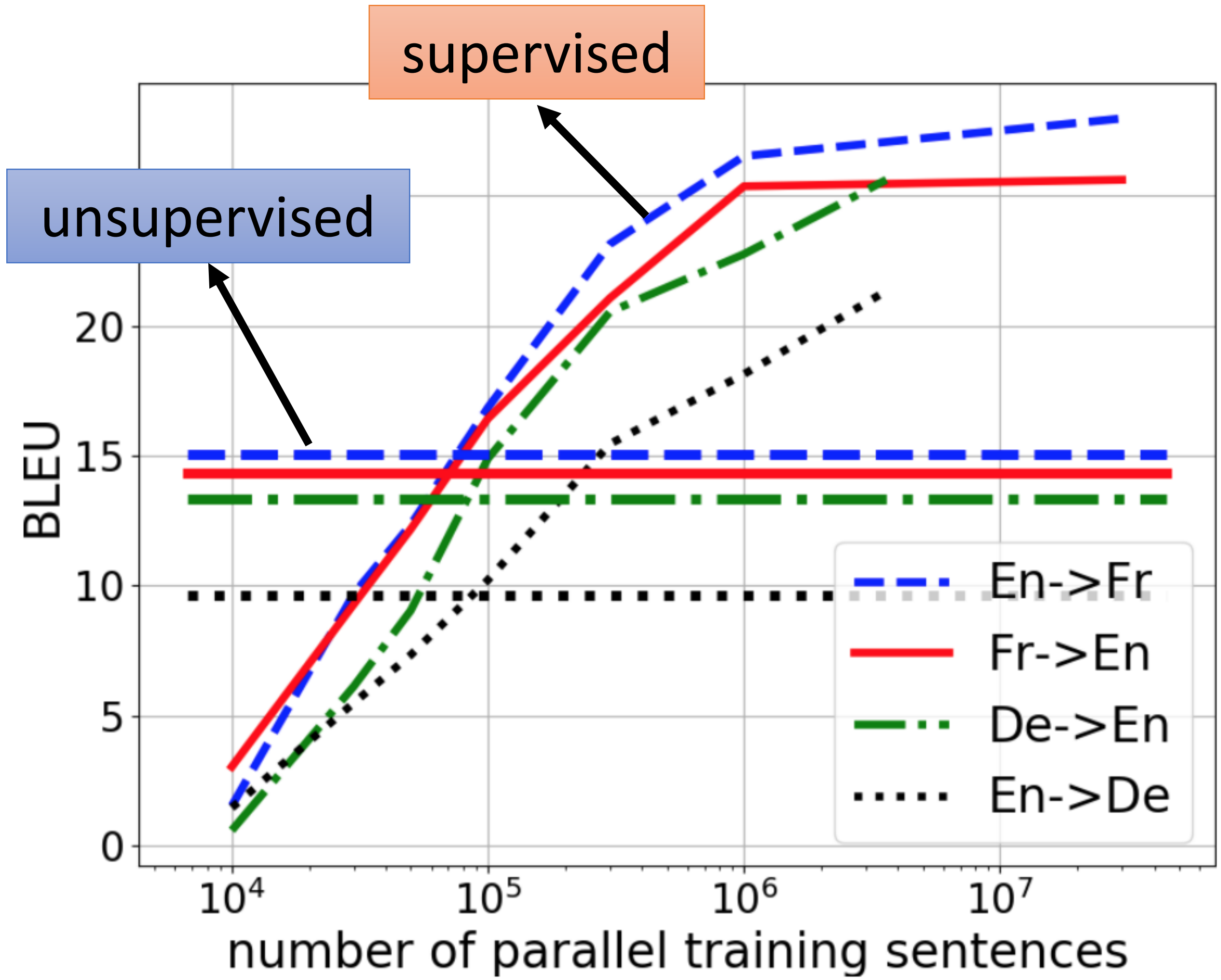
Domain Y

male

female

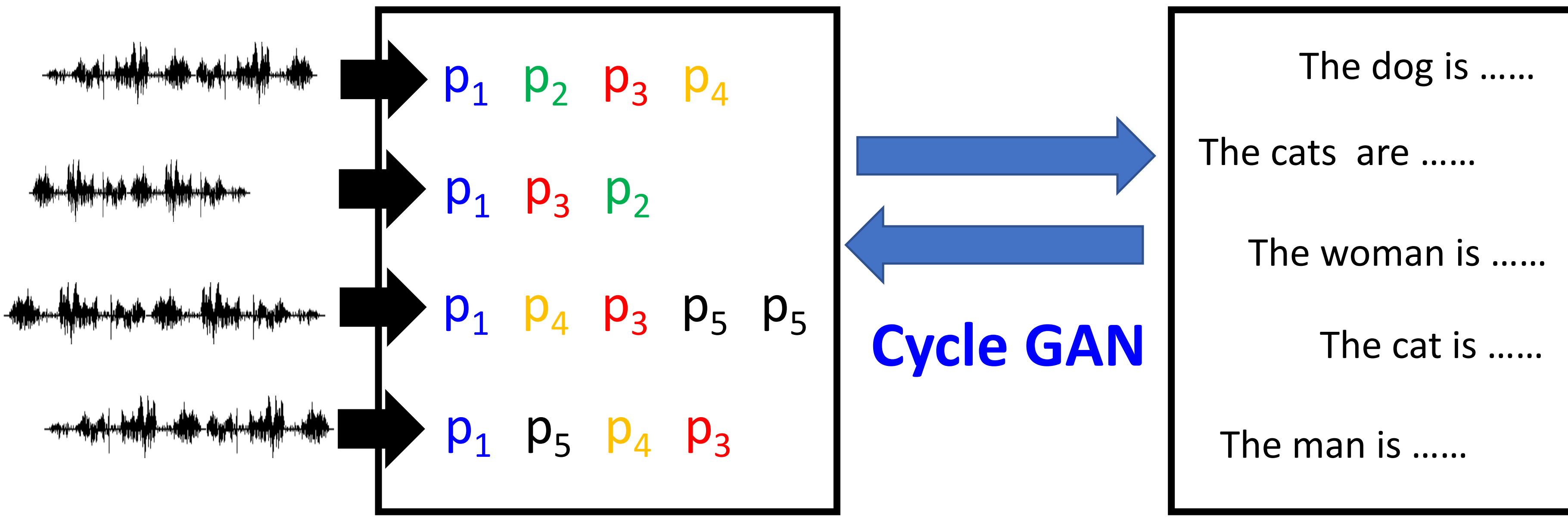


[Alexis Conneau, et al., ICLR, 2018]
[Guillaume Lample, et al., ICLR, 2018]



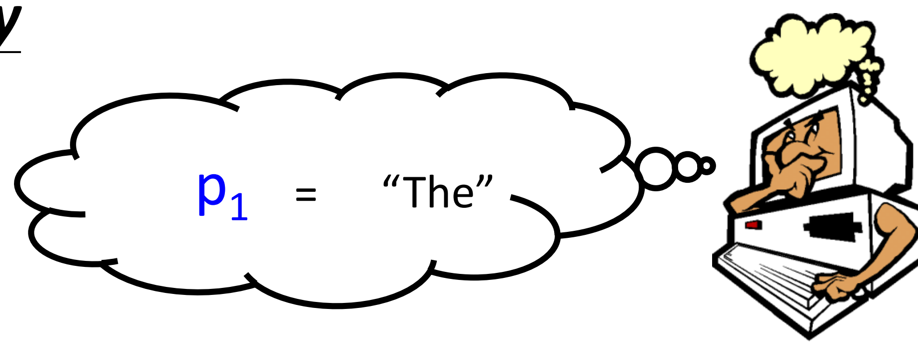
Unsupervised learning with 10M sentences = **Supervised** learning with 100K sentence pairs

Unsupervised Speech Recognition



Acoustic Pattern Discovery

Can we achieve unsupervised speech recognition?

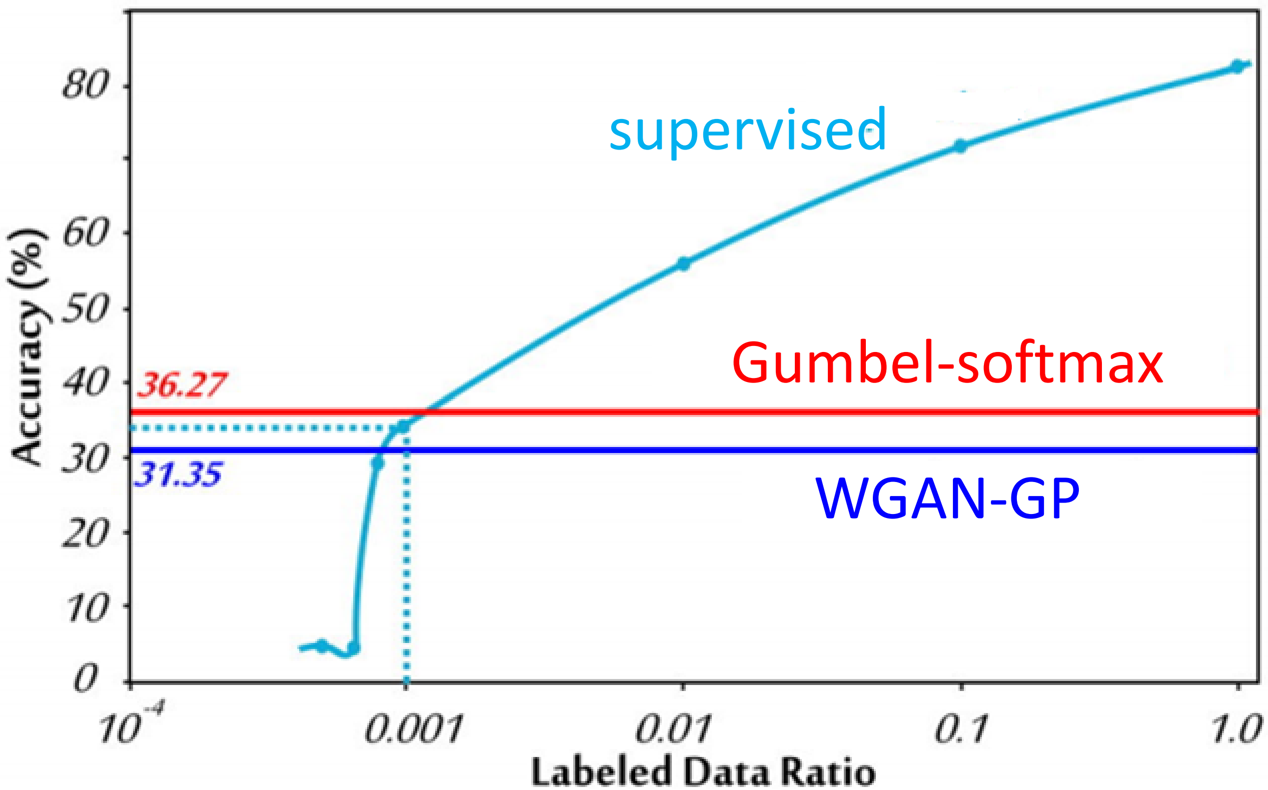


[Liu, et al., arXiv, 2018] [Chen, et al., arXiv, 2018]

Unsupervised Speech Recognition

- Phoneme recognition

Audio: TIMIT
Text: WMT



Concluding Remarks

Conditional Sequence Generation

- RL (human feedback)
- GAN (discriminator feedback)

Unsupervised Conditional Sequence Generation

- Text Style Transfer
- Unsupervised Abstractive Summarization
- Unsupervised Translation

To Learn More …

You can learn more from the YouTube Channel

https://www.youtube.com/playlist?list=PLJV_el3uVTsMd2G9ZjcpJn1YfnM9wVOBf

(in Mandarin)

Reference

- **Conditional Sequence Generation**
  - Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, Dan Jurafsky, Deep Reinforcement Learning for Dialogue Generation, EMNLP, 2016
  - Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, Dan Jurafsky, Adversarial Learning for Neural Dialogue Generation, EMNLP, 2017
  - Matt J. Kusner, José Miguel Hernández-Lobato, GANS for Sequences of Discrete Elements with the Gumbel-softmax Distribution, arXiv 2016
  - Tong Che, Yanran Li, Ruixiang Zhang, R Devon Hjelm, Wenjie Li, Yangqiu Song, Yoshua Bengio, Maximum-Likelihood Augmented Discrete Generative Adversarial Networks, arXiv 2017
  - Lantao Yu, Weinan Zhang, Jun Wang, Yong Yu, SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient, AAAI 2017

Reference

- **Conditional Sequence Generation**

  - Sai Rajeswar, Sandeep Subramanian, Francis Dutil, Christopher Pal, Aaron Courville, Adversarial Generation of Natural Language, arXiv, 2017
  - Ofir Press, Amir Bar, Ben Bogin, Jonathan Berant, Lior Wolf, Language Generation with Recurrent Generative Adversarial Networks without Pre-training, ICML workshop, 2017
  - Zhen Xu, Bingquan Liu, Baoxun Wang, Chengjie Sun, Xiaolong Wang, Zhuoran Wang, Chao Qi , Neural Response Generation via GAN with an Approximate Embedding Layer, EMNLP, 2017
  - Alex Lamb, Anirudh Goyal, Ying Zhang, Saizheng Zhang, Aaron Courville, Yoshua Bengio, Professor Forcing: A New Algorithm for Training Recurrent Networks, NIPS, 2016
  - Yizhe Zhang, Zhe Gan, Kai Fan, Zhi Chen, Ricardo Henao, Dinghan Shen, Lawrence Carin, Adversarial Feature Matching for Text Generation, ICML, 2017
  - Jiaxian Guo, Sidi Lu, Han Cai, Weinan Zhang, Yong Yu, Jun Wang, Long Text Generation via Adversarial Training with Leaked Information, AAAI, 2018
  - Kevin Lin, Dianqi Li, Xiaodong He, Zhengyou Zhang, Ming-Ting Sun, Adversarial Ranking for Language Generation, NIPS, 2017
  - William Fedus, Ian Goodfellow, Andrew M. Dai, MaskGAN: Better Text Generation via Filling in the______, ICLR, 2018

Reference

- **Conditional Sequence Generation**

  - Sidi Lu, Yaoming Zhu, Weinan Zhang, Jun Wang, Yong Yu, Neural Text Generation: Past, Present and Beyond, arXiv, 2018
  - Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, Yong Yu, Texygen: A Benchmarking Platform for Text Generation Models, arXiv, 2018
  - Zhen Yang, Wei Chen, Feng Wang, Bo Xu, Improving Neural Machine Translation with Conditional Sequence Generative Adversarial Nets, NAACL, 2018
  - Lijun Wu, Yingce Xia, Li Zhao, Fei Tian, Tao Qin, Jianhuang Lai, Tie-Yan Liu, Adversarial Neural Machine Translation, arXiv 2017
  - Linqing Liu, Yao Lu, Min Yang, Qiang Qu, Jia Zhu, Hongyan Li, Generative Adversarial Network for Abstractive Text Summarization, AAAI 2018
  - Rakshith Shetty, Marcus Rohrbach, Lisa Anne Hendricks, Mario Fritz, Bernt Schiele, Speaking the Same Language: Matching Machine to Human Captions by Adversarial Training, ICCV 2017
  - Xiaodan Liang, Zhiting Hu, Hao Zhang, Chuang Gan, Eric P. Xing, Recurrent Topic-Transition GAN for Visual Paragraph Generation, arXiv 2017

Reference

- **Text Style Transfer**
  - Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, Rui Yan, Style Transfer in Text: Exploration and Evaluation, AAAI, 2018
  - Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola, Style Transfer from Non-Parallel Text by Cross-Alignment, NIPS 2017
  - Chih-Wei Lee, Yau-Shian Wang, Tsung-Yuan Hsu, Kuan-Yu Chen, Hung-Yi Lee, Lin-shan Lee, Scalable Sentiment for Sequence-to-sequence Chatbot Response with Performance Analysis, ICASSP, 2018
  - Junbo (Jake) Zhao, Yoon Kim, Kelly Zhang, Alexander M. Rush, Yann LeCun, Adversarially Regularized Autoencoders, arxiv, 2017

Reference

- **Unsupervised Machine Translation**
  - Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, Hervé Jégou, Word Translation Without Parallel Data, ICRL 2018
  - Guillaume Lample, Ludovic Denoyer, Marc'Aurelio Ranzato, Unsupervised Machine Translation Using Monolingual Corpora Only, ICRL 2018
- **Unsupervised Speech Recognition**
  - Da-Rong Liu, Kuan-Yu Chen, Hung-Yi Lee, Lin-shan Lee, Completely Unsupervised Phoneme Recognition by Adversarially Learning Mapping Relationships from Audio Embeddings, arXiv, 2018
  - Yi-Chen Chen, Chia-Hao Shen, Sung-Feng Huang, Hung-yi Lee, Towards Unsupervised Automatic Speech Recognition Trained by Unaligned Speech and Text only, arXiv, 2018