Language Technologies Institute

Carnegie Mellon University

Multimodal Machine Learning

Lecture 7.2: Generative Models

Louis-Philippe Morency

* Original version co-developed with Tadas Baltrusaitis

Outline

- Probabilistic graphical models
  - Joint probabilistic distribution
  - Example: creating a graphical model
- Bayesian networks
  - Conditional probability distribution
  - Dynamic Bayesian Network
- Variational Auto Encoder
- Generative Adversarial Network
  - cGAN, infoGAN, cycleGAN

Probabilistic Graphical Models

Probabilistic Graphical Model

Definition: A probabilistic graphical model (PGM) is a graph formalism for compactly modeling joint probability distributions and dependence structures over a set of random variables.

- Random variables: $X_1,\ldots,X_n$
- P is a joint distribution over $X_1,\ldots,X_n$

Why do we want to learn the joint distribution?

Inference for Known Joint Probability Distribution

When we know the joint probability distribution :

$$P(A, B, C, D, E)$$

→ If A, B C, D and E are discrete variables, then P(A,B,C,D, E) will be a 5-D tensor (matrix)

Two main forms of inference:

1. Joint probability for a particular assignment

$$P(A = 1, B = 'car', C = 2, D = 'banana', E = 10)$$

→ A specific entry in the 5-D tensor

Inference for Known Joint Probability Distribution

2. Probability of a subset of variables (query) given known assignments of other variables (evidences)

$P(A, D|C = 3)$ → Use the product rule to *marginalize* the other variables B and E

$$P(A, D|C = 3) = \sum_{\forall b \in B, e \in E} P(A, D, b, e|C = 3)$$

→ Use the inverse of product rule $P(X|Y) = P(X, Y)/P(Y)$

$$P(A, D|C = 3) = \frac{1}{P(C)} \sum_{\forall b \in B, e \in E} P(A, D, b, e|C = 3)$$

Inference for Known Joint Probability Distribution

2. Probability of a subset of variables (query) given known assignments of other variables (evidences)

$$P(x|y) = \alpha \sum_{\forall z \in Z} P(x, y, z)$$

where x is the subset of query variables

y is the subset of evidence assignments

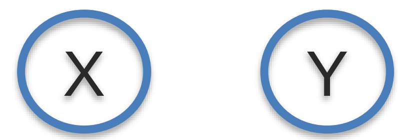
Z is the set of all other variables (not in x or y)

Can we represent P more compactly?

- Key: Exploit independence properties

Independent Random Variables

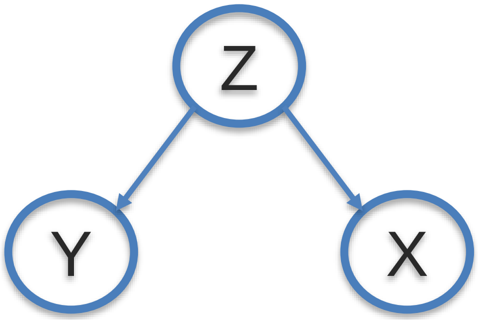
- Two variables X and Y are independent if
  - $P(X=x|Y=y) = P(X=x)$ for all values x,y
  - Equivalently, knowing Y does not change predictions of X
- If X and Y are independent then:
  - $P(X, Y) = P(X|Y)P(Y) = P(X)P(Y)$



- If $X_1,\ldots,X_n$ are independent then:
  - $P(X_1,\ldots,X_n) = P(X_1)\ldots P(X_n)$

Conditional Independence

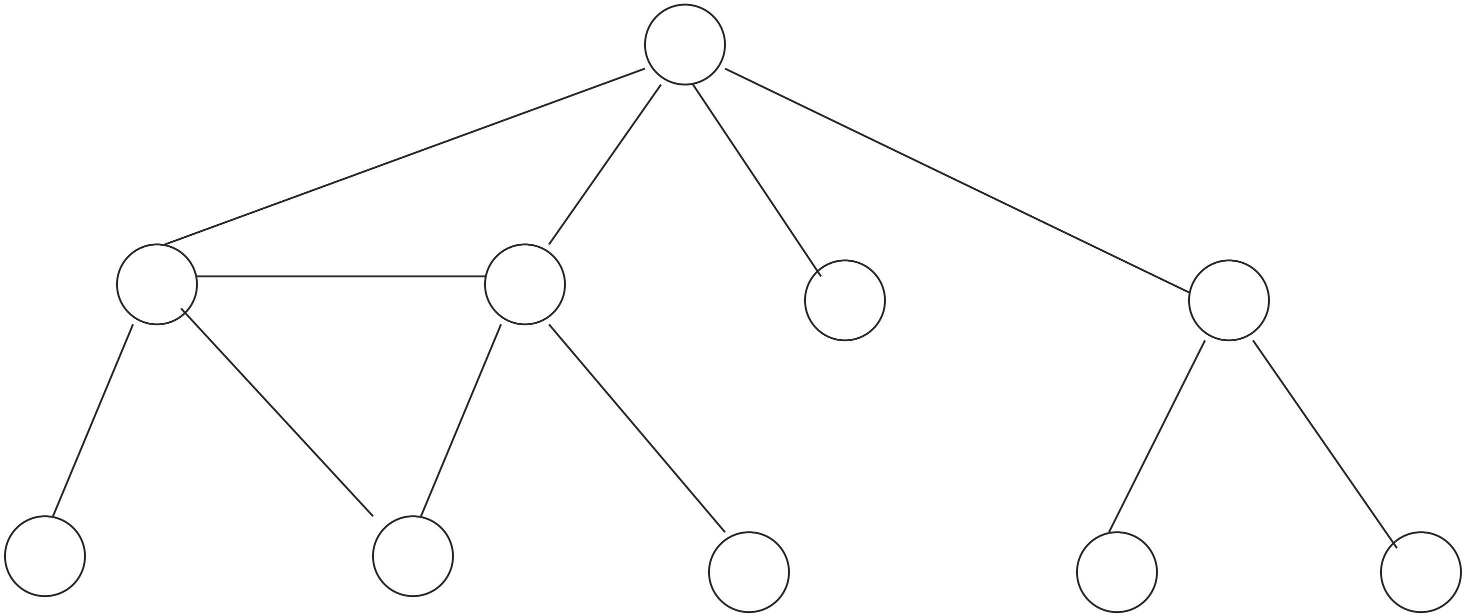
- X and Y are conditionally independent given Z if
  - P(X=x|Y=y, Z=z) = P(X=x|Z=z) for all values x, y, z
  - Equivalently, if we know Z, then knowing Y does not change predictions of X



Graphical Model

- A tool that visually illustrate conditional independence among variables in a given problem.
- Consisting of nodes (Random variables or States) and edges (Connecting two nodes, directed or undirected).
- The lack of edge represents conditional independence between variables.

Graphical Model

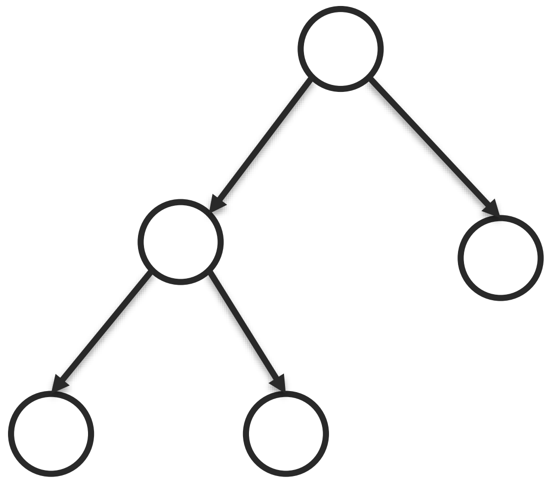


Different types of graphical models:

- Chain, Path, Cycle, Directed Acyclic Graph (DAG), Parents and Children

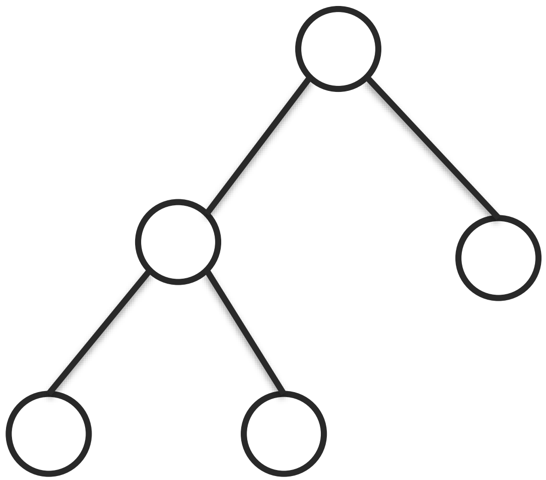
Two Main Types of Graphical Models

Bayesian networks



- Directed acyclic graph
- Conditional dependencies

Markov Models (in 2 weeks)

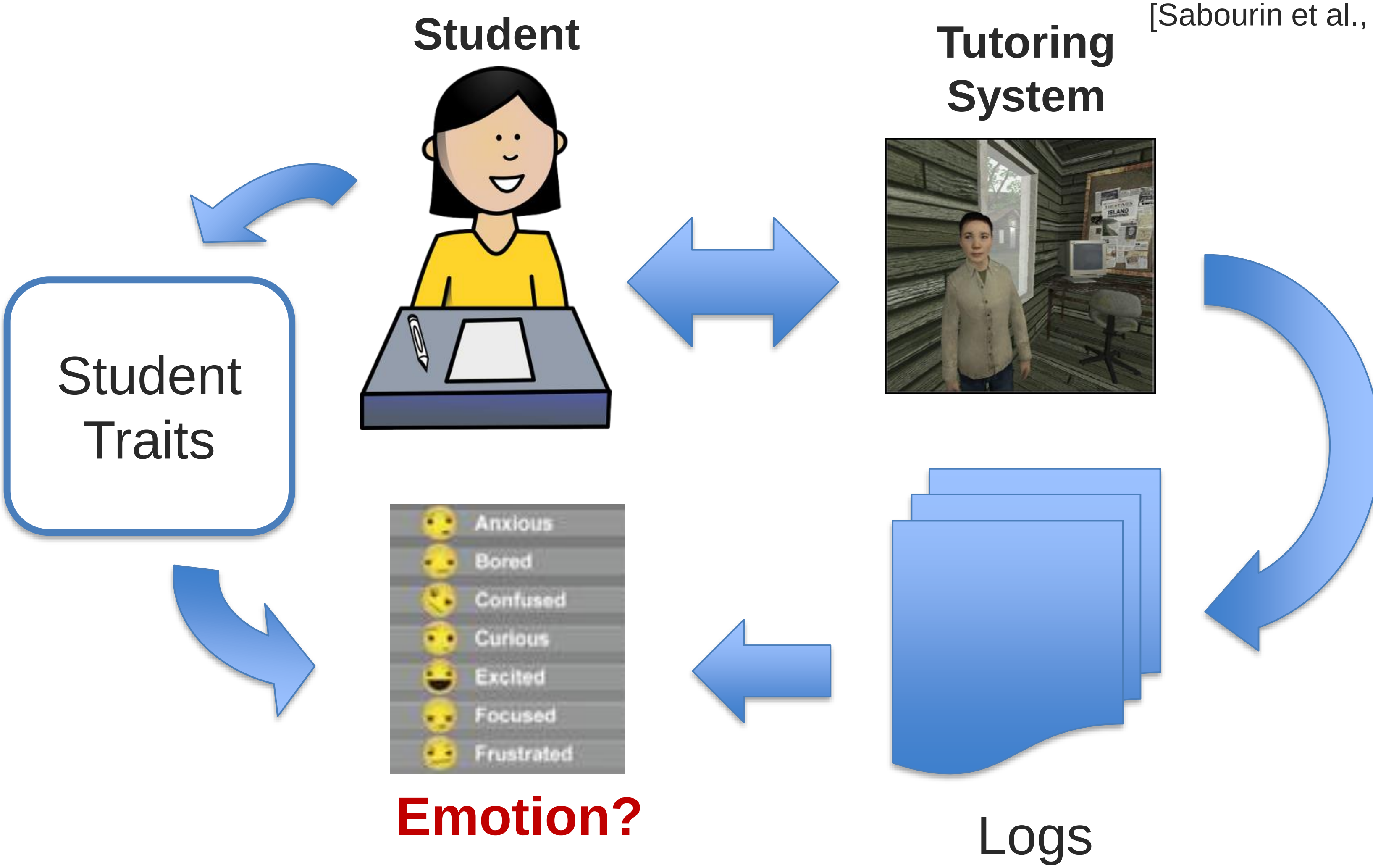


- Undirected graphical model
- Cyclic dependencies

Creating a Graphical Model

Example: Inferring Emotion from Interaction Logs

[Sabourin et al., 2011]



Example: Bayesian Network Representation

[Sabourin et al., 2011]

Outcome (non-observable)

Emotion

Evidences (observable)

Observable environment variables

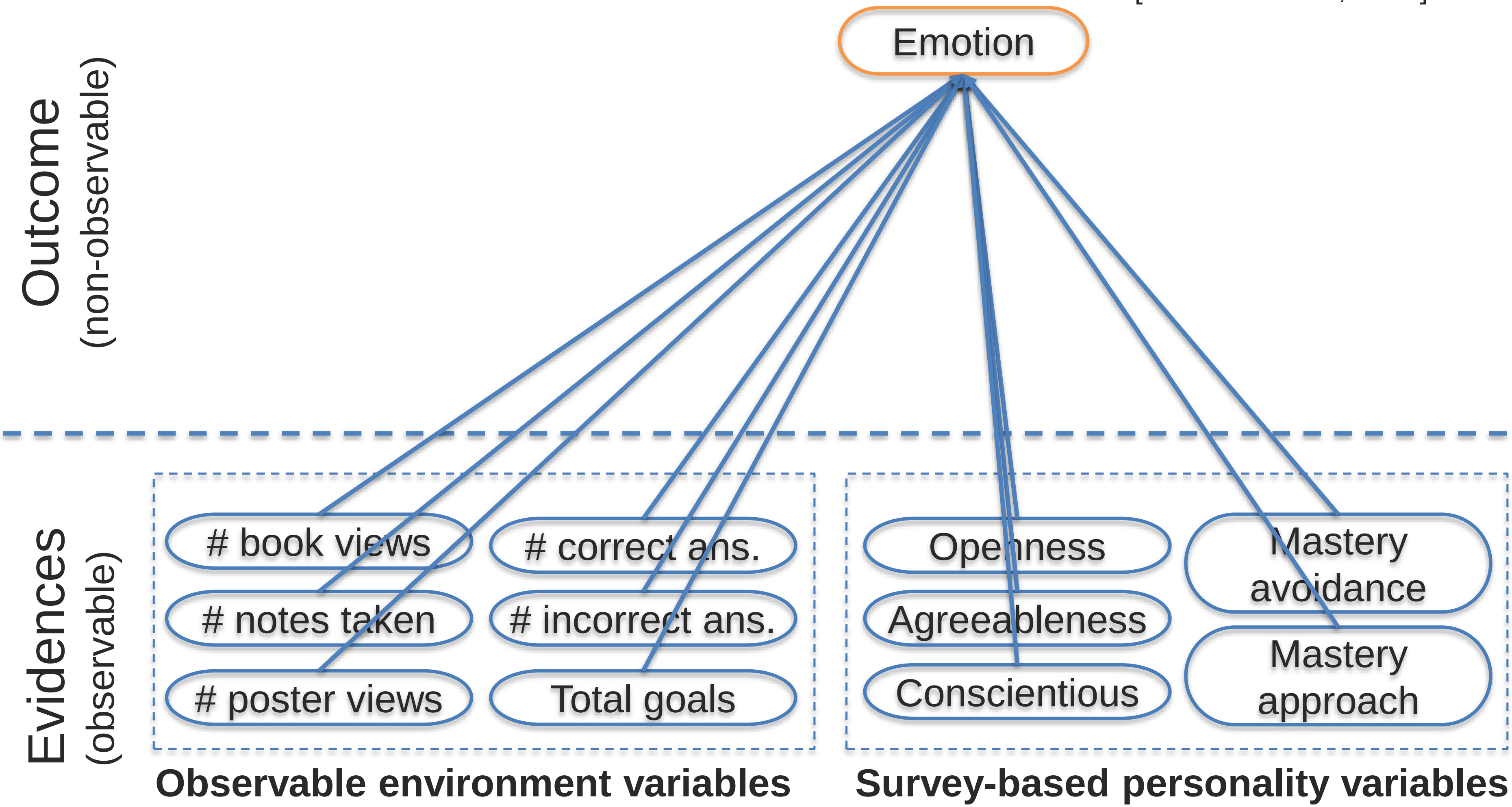
- # book views
- # correct ans.
- # notes taken
- # incorrect ans.
- # poster views
- Total goals

Survey-based personality variables

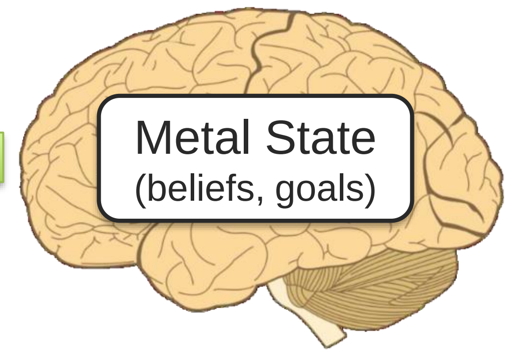
- Openness
- Agreeableness
- Conscientious
- Mastery avoidance
- Mastery approach

Example: Naïve Bayes Approach

[Sabourin et al., 2011]



Appraisal Theory of Emotion



Argues for importance of three interrelated concepts

- World events
- Mental state
- Emotional Response

Body

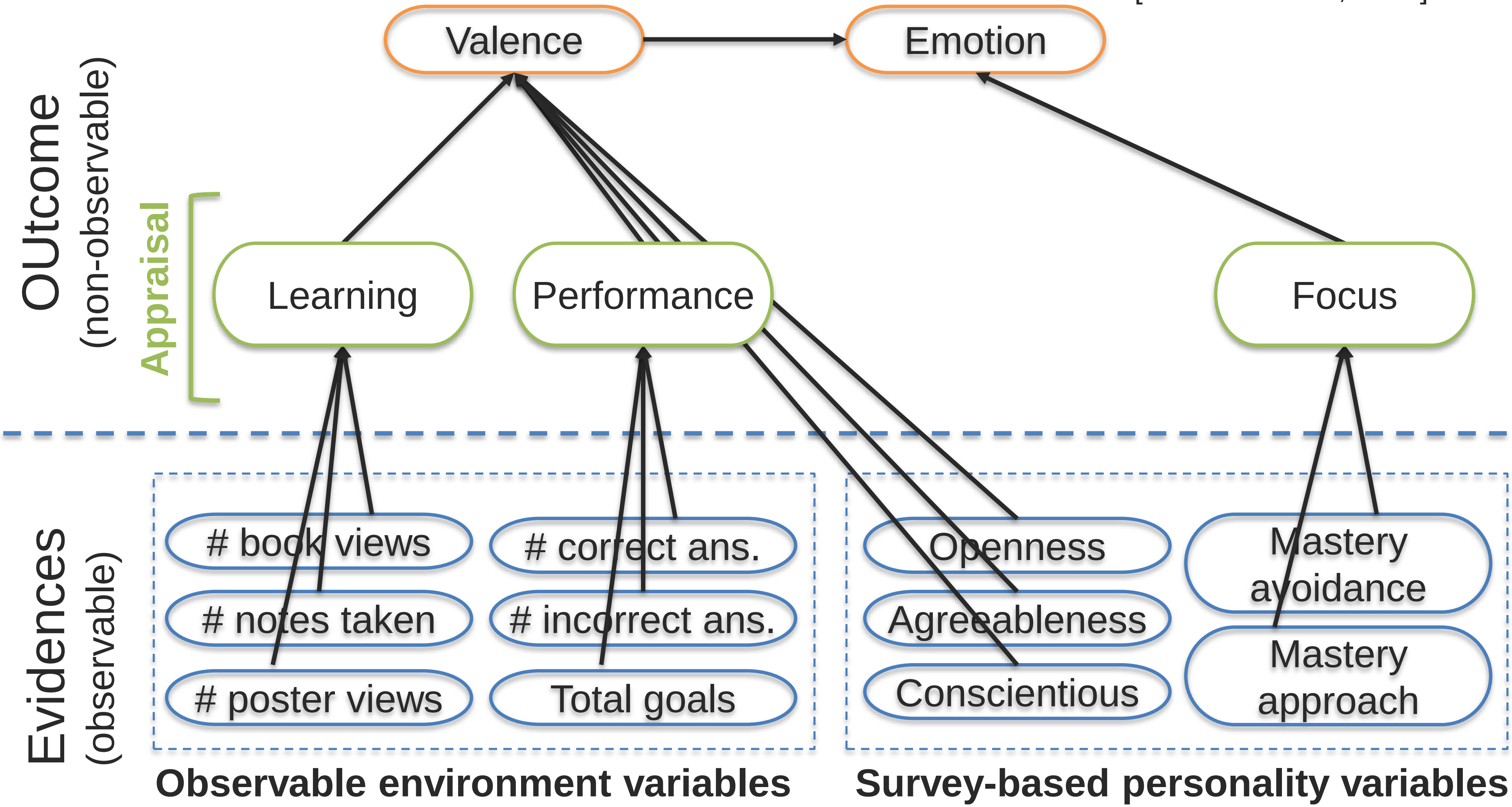
- Expression
- Action tendency
- Physiological response

If we know two of these variables, we can make predictions about the third

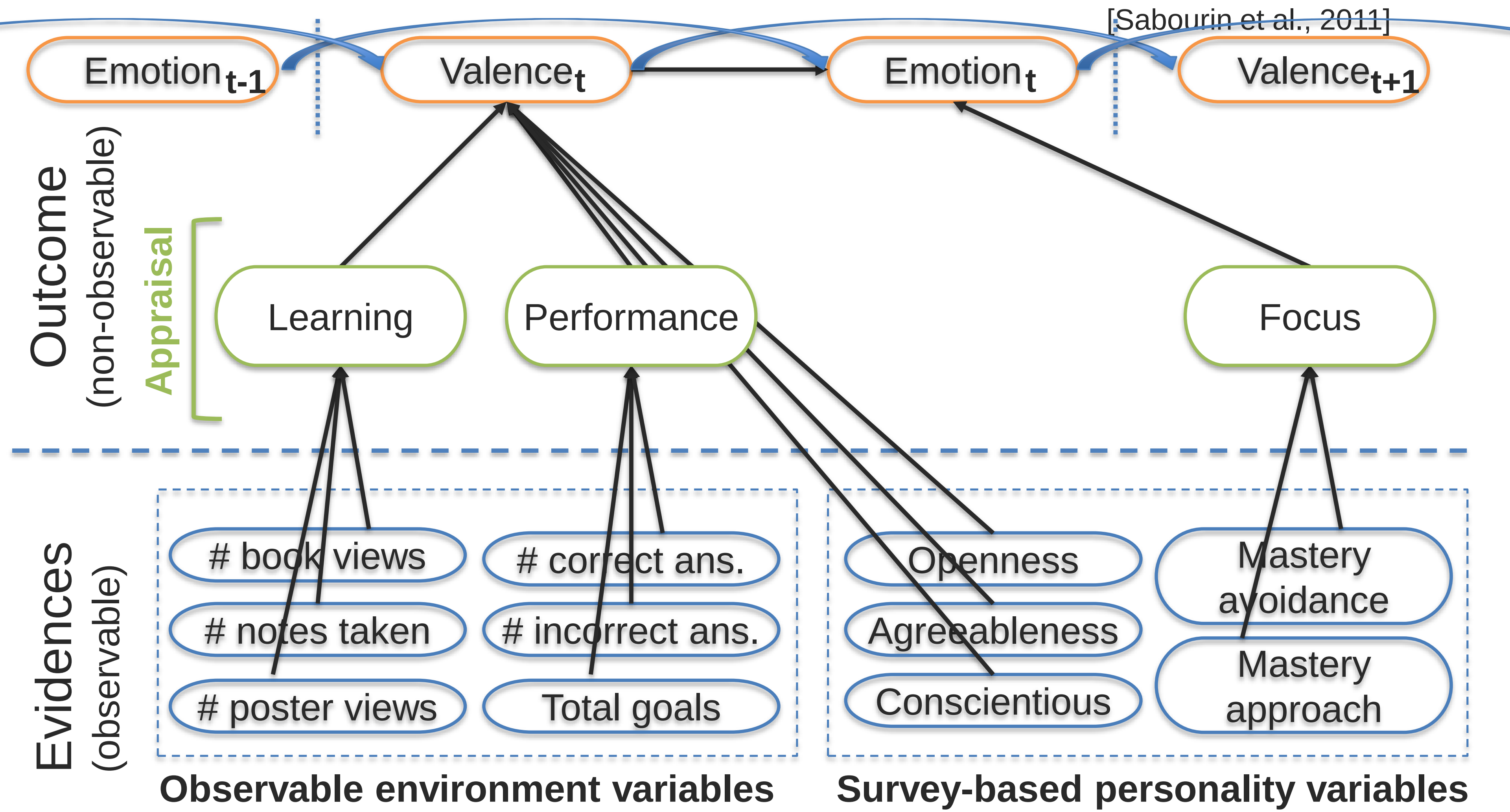
Response= f(Env., Mind)

Example: Bayesian Network Approach

[Sabourin et al., 2011]

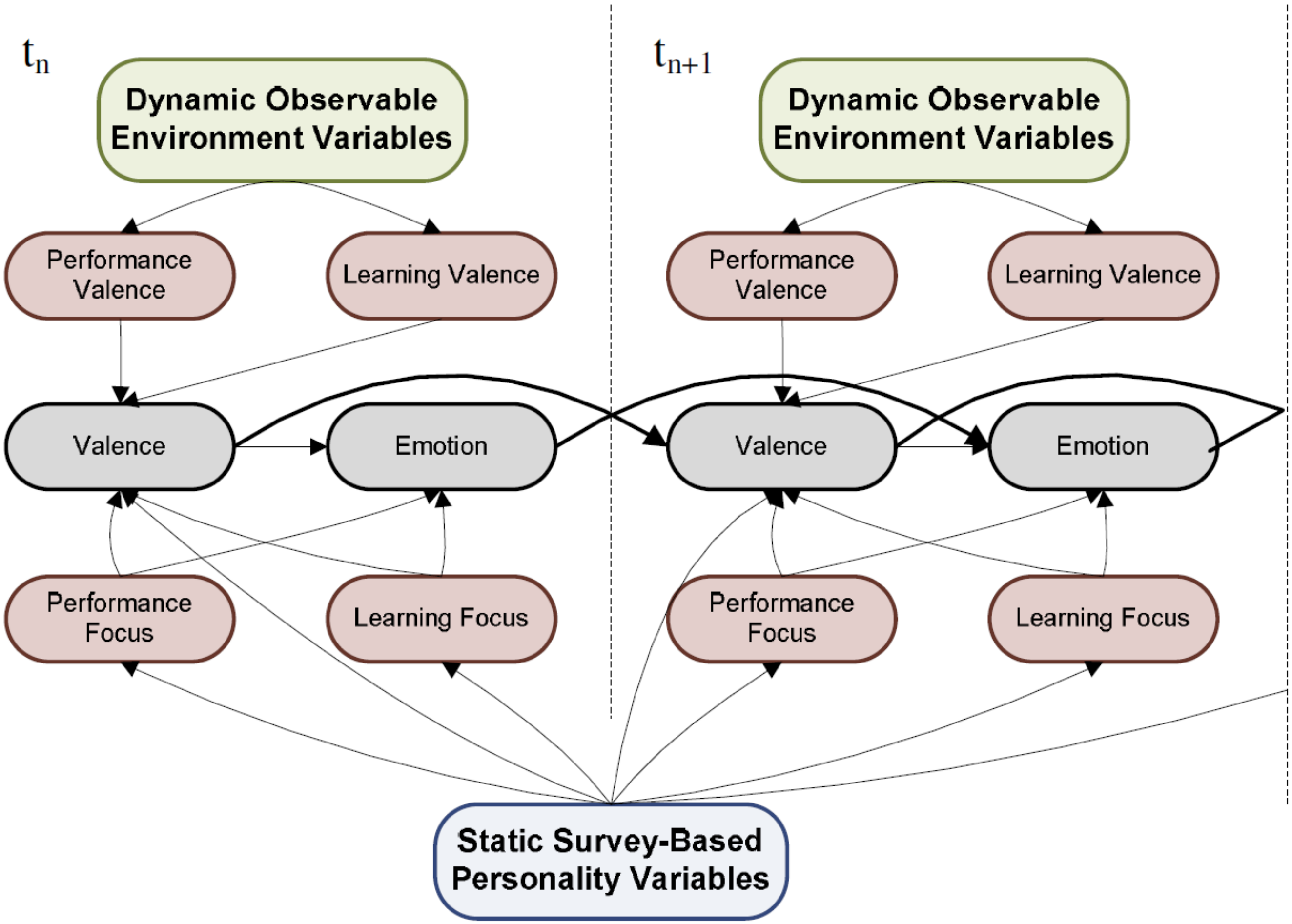


Example: Dynamic Bayesian Network Approach



Example: Dynamic Bayesian Network Approach

[Sabourin et al., 2011]



Example: Inferring Emotion from Interaction Logs

[Sabourin et al., 2011]

Student

Tutoring System

	Emotion Accuracy	Valence Accuracy
Baseline	22.4%	54.5%
Naïve Bayes	18.1%	51.2%
Bayes Net	25.5%	66.8%
Dynamic BN	32.6%	72.6%

Bayesian Networks

Bayesian networks

- A simple, graphical notation for conditional independence assertions and hence for compact specification of full joint distributions

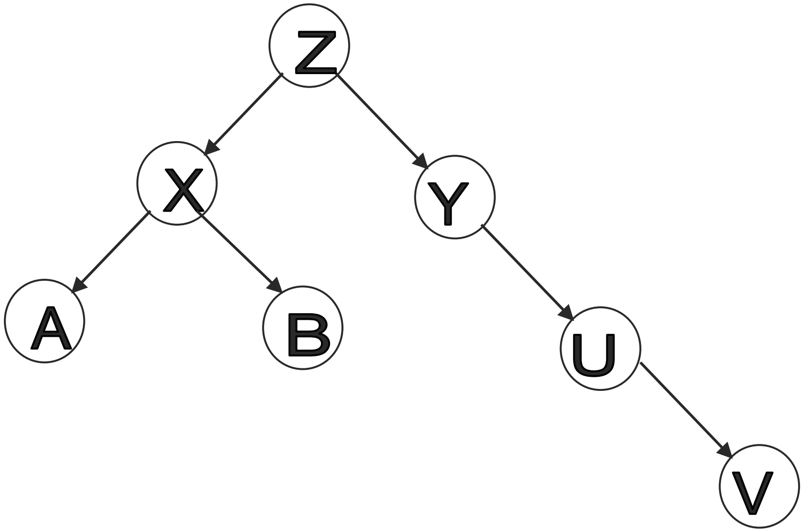
- Syntax:
  - a set of nodes, one per variable
  - a directed, acyclic graph (link ≈ "directly influences")
  - a conditional distribution for each node given its parents:

$$\mathbf{P}(X_i \mid Parents(X_i))$$

- In the simplest case, conditional distribution represented as a conditional probability distribution (CPD) giving the distribution over X_i for each combination of parent values

Bayesian Network (BN)

- A specific type of graphical model that is represented as a Directed Acyclic Graph.

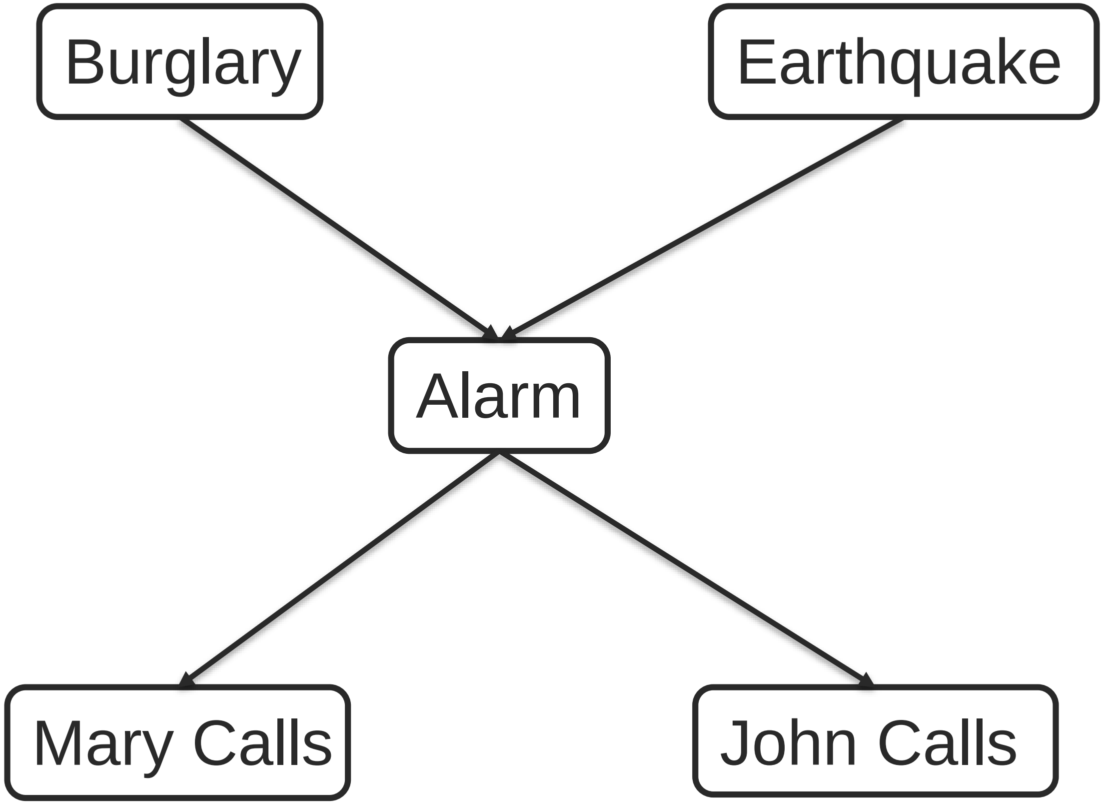


Example

“I'm at work, neighbor John calls to say my alarm is ringing, but neighbor Mary doesn't call. Sometimes it's set off by minor earthquakes. Is there a burglar?”

- Variables?
  - *Burglary*, *Earthquake*, *Alarm*, *JohnCalls*, *MaryCalls*
- “Causal" knowledge?
  - A burglar can set the alarm off
  - An earthquake can set the alarm off
  - The alarm can cause Mary to call
  - The alarm can cause John to call

Example – Network Topology



Joint Probability in Graphical Models

With chain-rule, the joint probability can be restated:

$$P(A,B,C,D,E) = P(A|B,C,D,E)P(B,C,D,E)$$

$$= P(A|B,C,D,E)P(B|C,D,E)P(C|D,E)$$

$$= P(A|B,C,D,E)P(B|C,D,E)P(C,D,E)$$

$$= P(A|B,C,D,E)P(B|C,D,E)P(C|D,E)P(D,E)$$

$$= P(A|B,C,D,E)P(B|C,D,E)P(C|D,E)P(D|E)P(E)$$

➡ The order in applying the chain-rule is arbitrary.

How can we simplify the joint probability even more, given the graphical model?

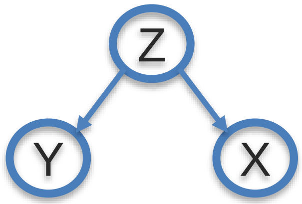
Joint Probability in Graphical Models

With chain-rule, the joint probability can be reshaped:

$$P(A,B,C,D,E) = P(A|B,C,D,E)P(B|C,D,E)P(C|D,E)P(D|E)P(E)$$

- Remember these concepts:

Independent variables



conditionally independent

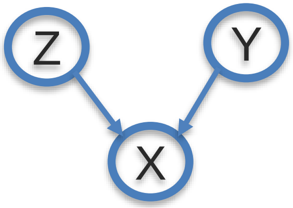
- In a Bayesian network, each conditional probability for a specific variable X only depends on its parents:

$$P(X|\ all\ variables) = \underline{P(X|parents(X))}$$

Conditional Probability Distribution (CPD)

Conditional Probability Distribution (CPD)

Given a variable X and its parents (Y and Z):



$$P(X|parents(X)) = P(X|Y,Z)$$

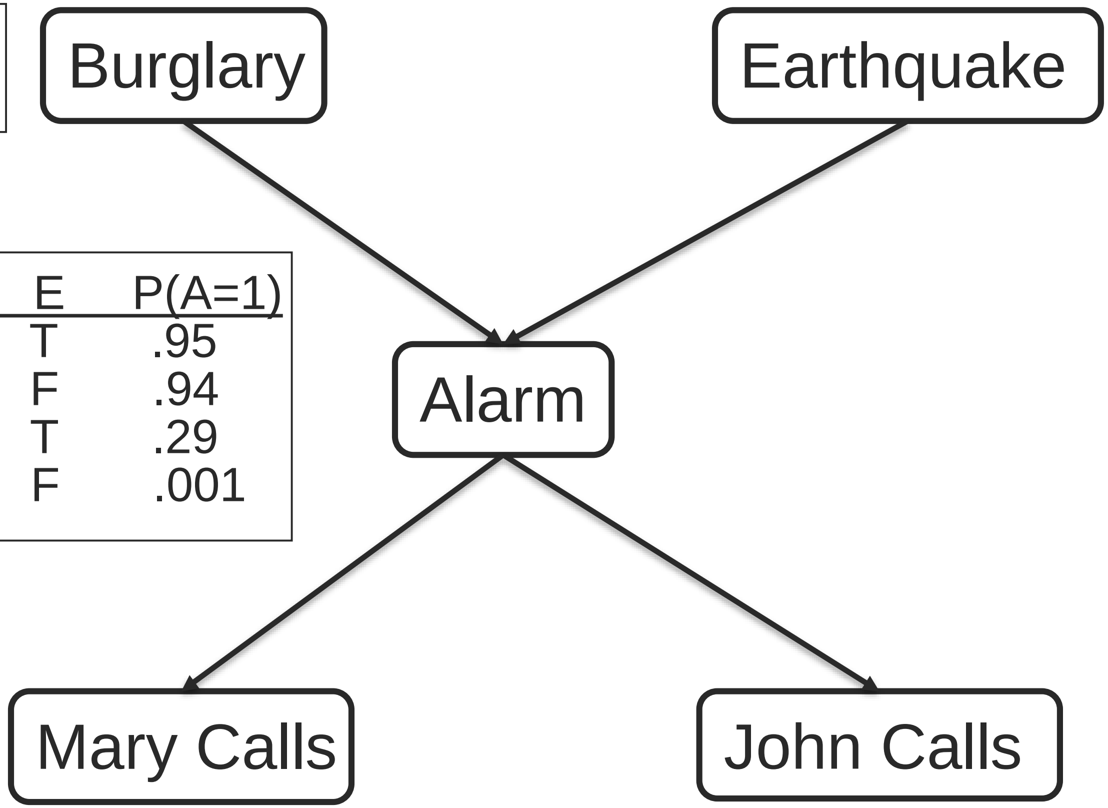
Definition: probability distribution of X when the assignment of it parents is known (Y and Z)

- For **categorical variable**: expressed as a conditional probability table

	Y=0	Y=1
P(X=0|Y)	4/6	1/3
P(X=1|Y)	2/6	2/3

- For **continuous variable**: expressed as a conditional density function
  - For example, multivariate normal density function or Gaussian linear regression (used by Bayes RegressionLinear Model)

Example – Conditional Probability Distributions



P(B=1)
.001

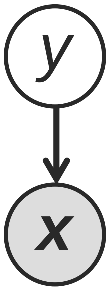
P(E=1)
.002

B	E	P(A=1)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(M=1)
T	.70
F	.01

A	P(J=1)
T	.90
F	.05

Generative Model: Naïve Bayes Classifier



Label : {0:`Dominant`, 1:`Not-dominant`}
(outcome)

Observation vector: [gaze, turn-taking,speech-energy]
(evidence)

Score function: $P(y = a|\boldsymbol{x_i})$

Likelihood

Prior

Chain rule

Bayes' theorem: $P(y|\boldsymbol{x}) = \dfrac{P(\boldsymbol{x}|y)P(y)}{P(\boldsymbol{x})} \approx \overline{P(\boldsymbol{x}|y)P(y)} = P(\boldsymbol{x}, y)$

Posterior

Marginal likelihood
(partition)

$P(\boldsymbol{x}) = \sum_y P(\boldsymbol{x}|y)P(y)$

Dynamic Bayesian Network

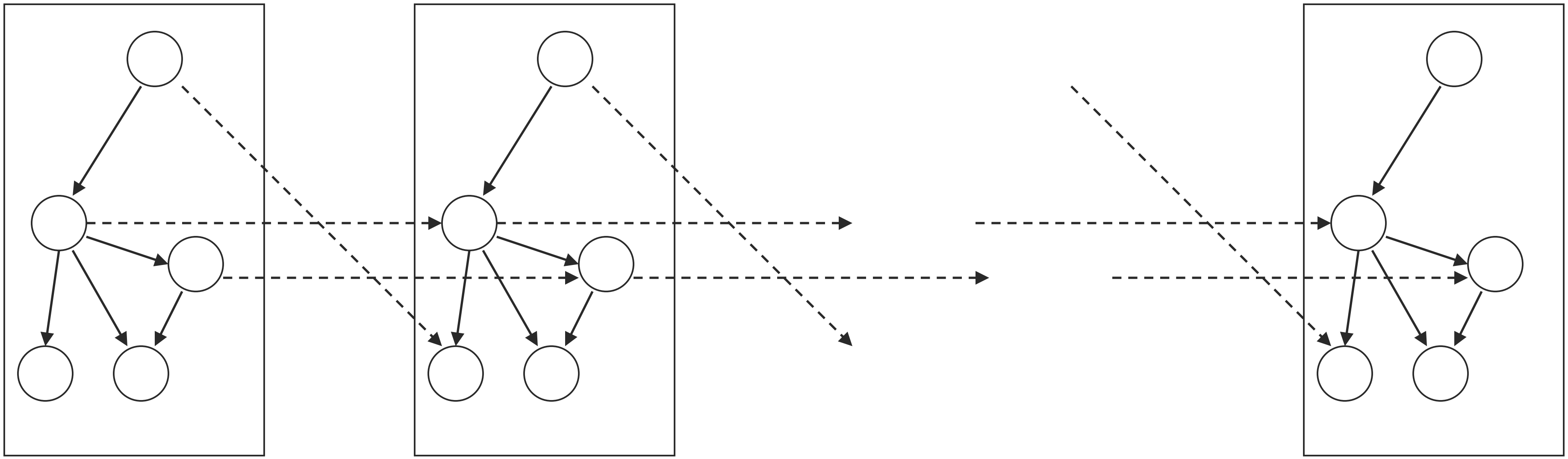
Dynamic Bayesian Network (DBN)

- Bayesian network with time-series to represent temporal dependencies.
- Dynamically changing or evolving over time.
- Directed graphical model of stochastic processes.
- Especially aiming at time series modeling.

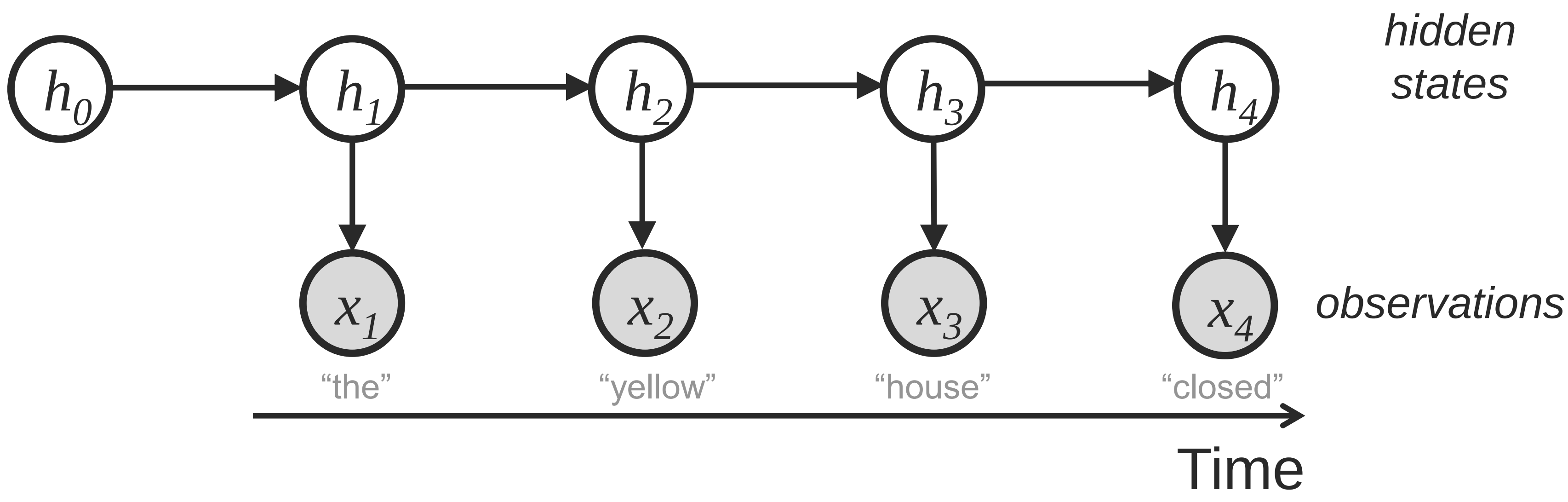
- Satisfying the Markovian condition:

  The state of a system at time t depends only on its immediate past state at time t-1.

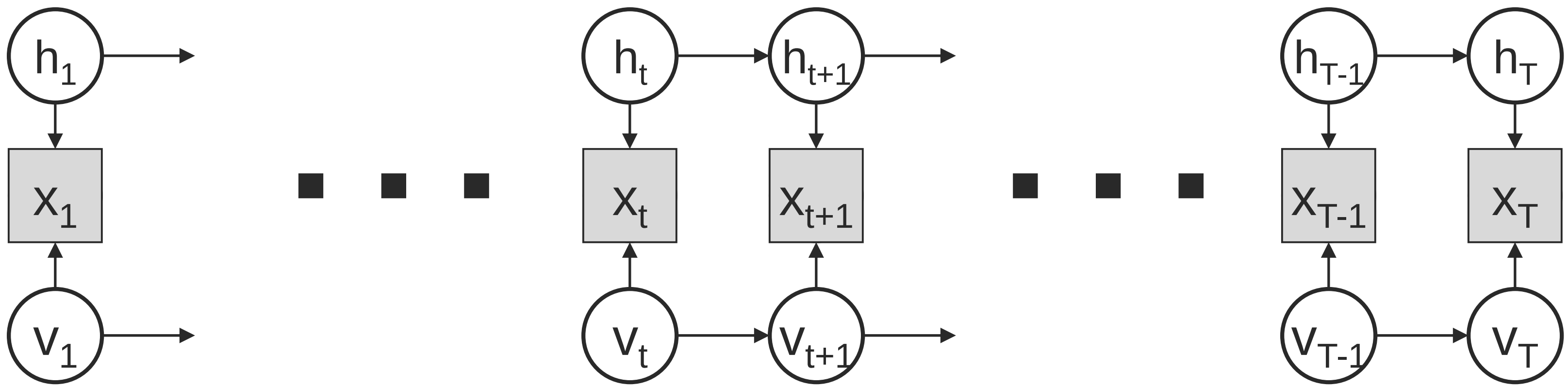
Dynamic Bayesian Network (DBN)



Hidden Markov Models

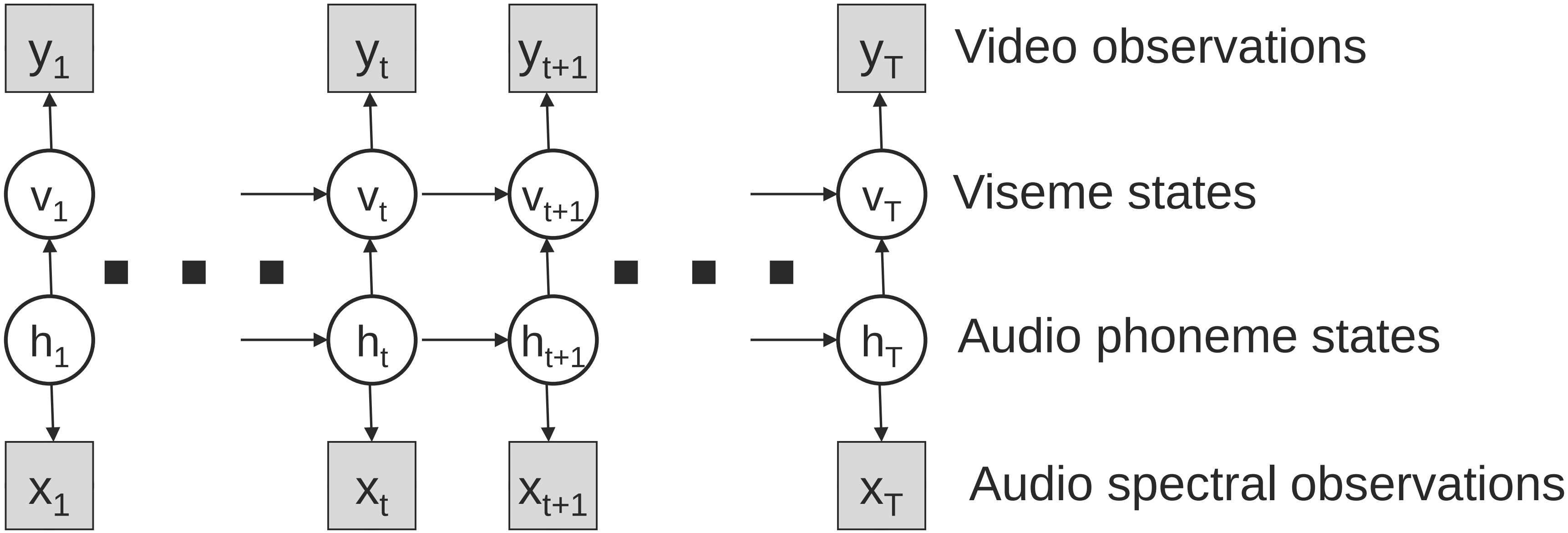


Factorial HMM



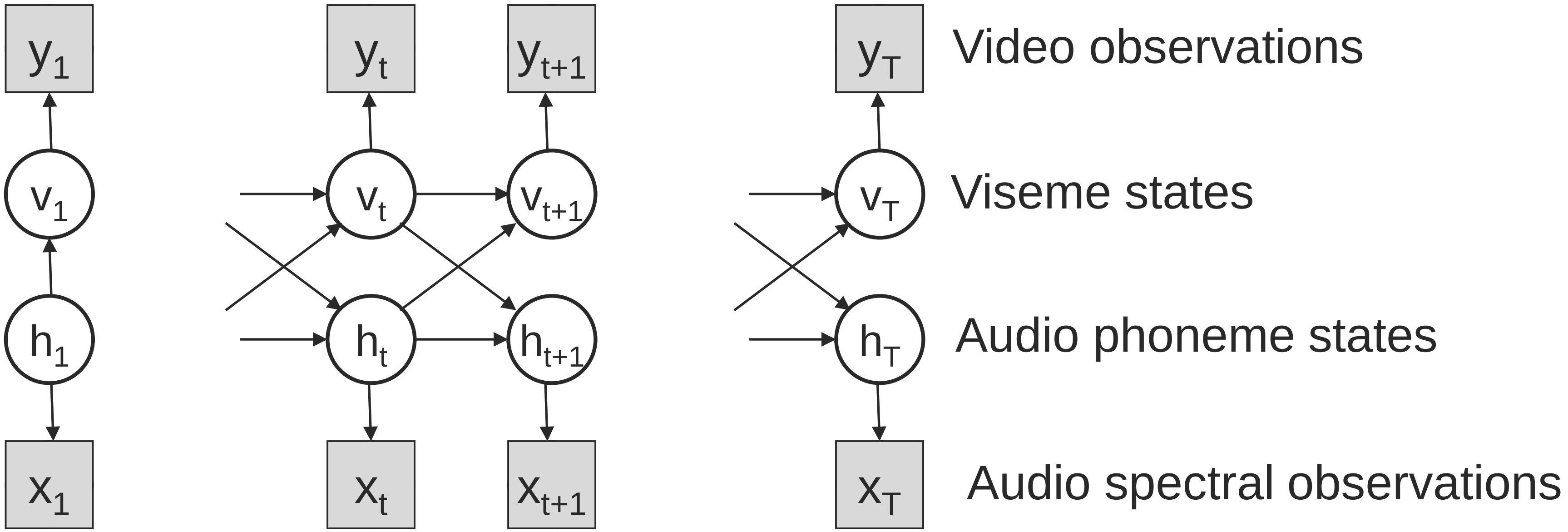
- Factorial HMM:
  - h_t and v_t represent two different types of background information, each with its own history
  - Observations x_t depend on both hidden processes
- Model parameters:
  - $p(v_{t+1}|v_t)$, $p(h_{t+1}|h_t)$, $p(x_t|h_t,v_t)$

The Boltzmann Zipper



- Both streams have a "memory" (h_t and v_t)
- Model parameters:
  - $p(h_{t+1}|h_t)$, $p(x_t|h_t)$
  - $p(v_{t+1}|v_t,h_{t+1})$, $p(y_t|h_t)$

The Coupled HMM



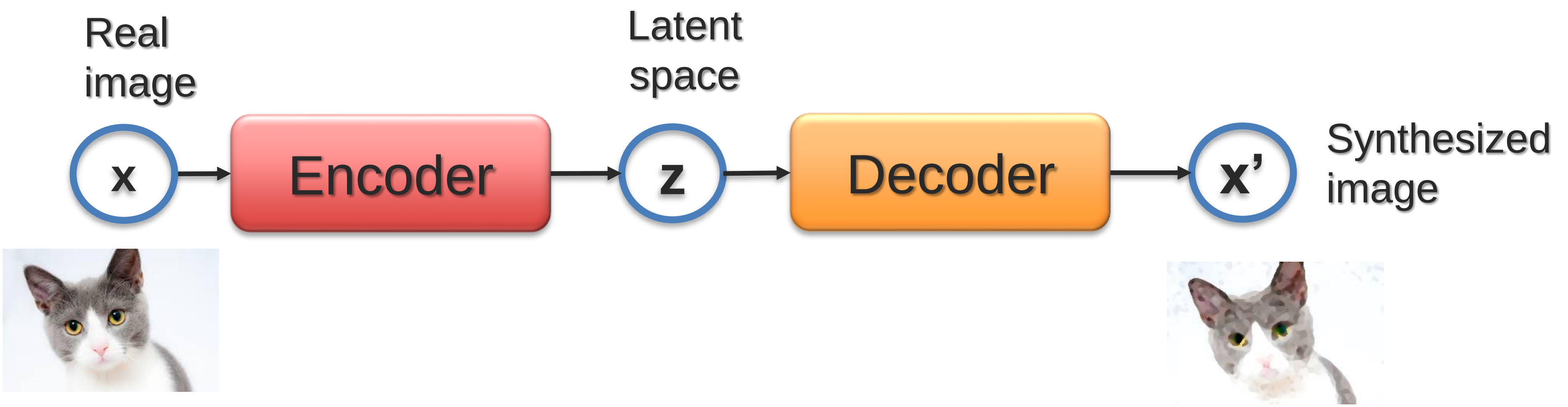
- Advantage over Boltzmann Zipper: More flexible, because neither vision nor sound is “privileged” over the other.
  - $p(h_{t+1}|v_t,h_t)$, $p(x_t|h_t)$
  - $p(v_{t+1}|v_t,h_t)$, $p(y_t|h_t)$

Learning (Dynamic) Bayesian Networks

- Multiple techniques exist to learn the model parameters based on data
  - Maximum likelihood estimator
  - Bayesian estimator, which allows to include prior information
- Python libraries:
  - http://pgmpy.org/
  - http://www.bayespy.org
  - https://pomegranate.readthedocs.io/en/latest/

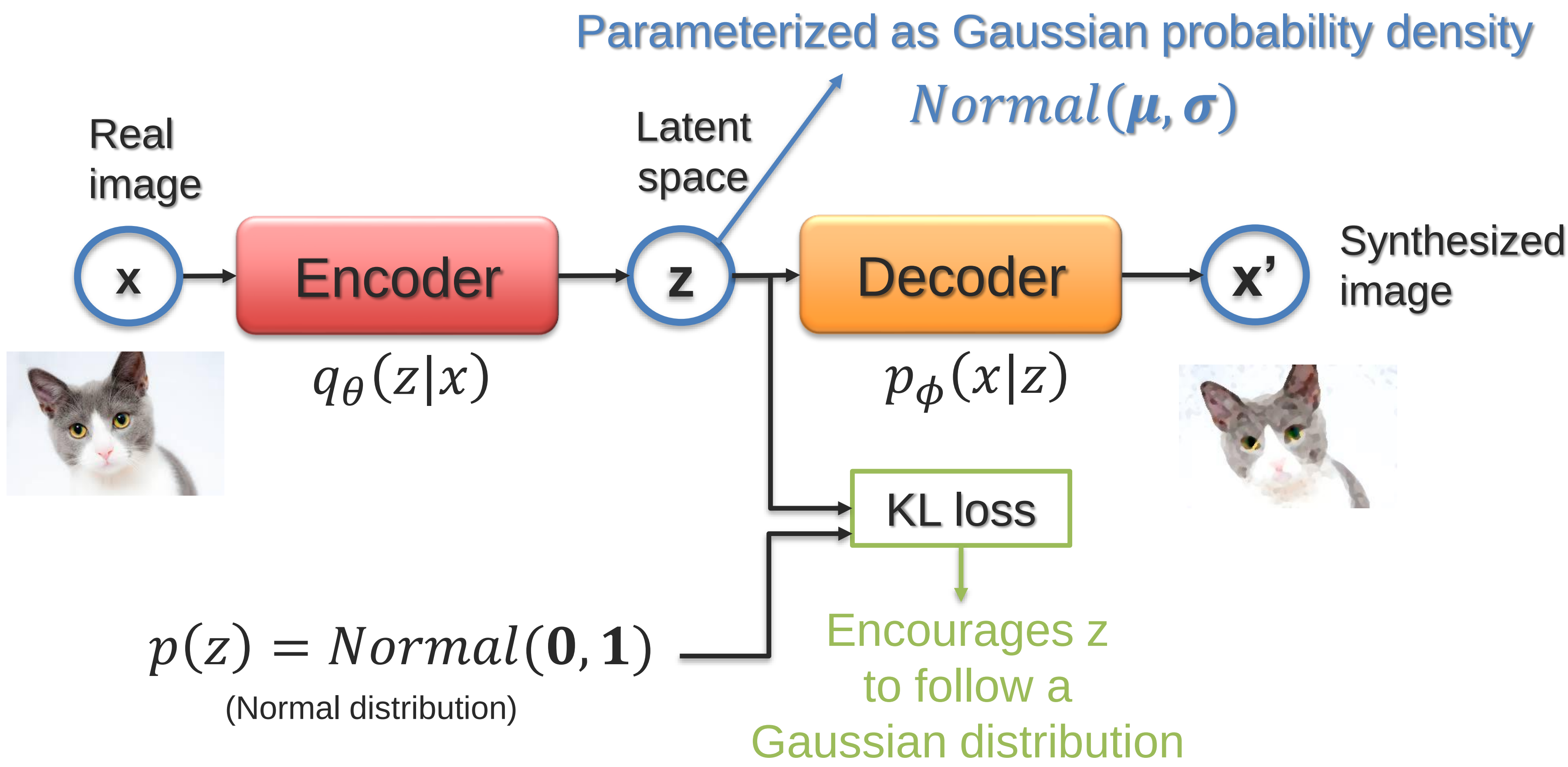
Variational AutoEncoder

Auto-encoder



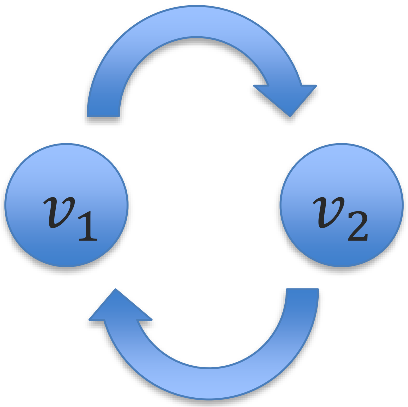
After learning this autoencoder,
can I input any z vector in the decoder?

Variational Autoencoder



Variational Inference

- When inference is not possible
  - Either relax the problem
  - Or use variational methods
- Variational inference:
  - Unroll through time (MCMC, Gibbs) – RBM
  - Mean-field Approximation (Fully Connected CRF)



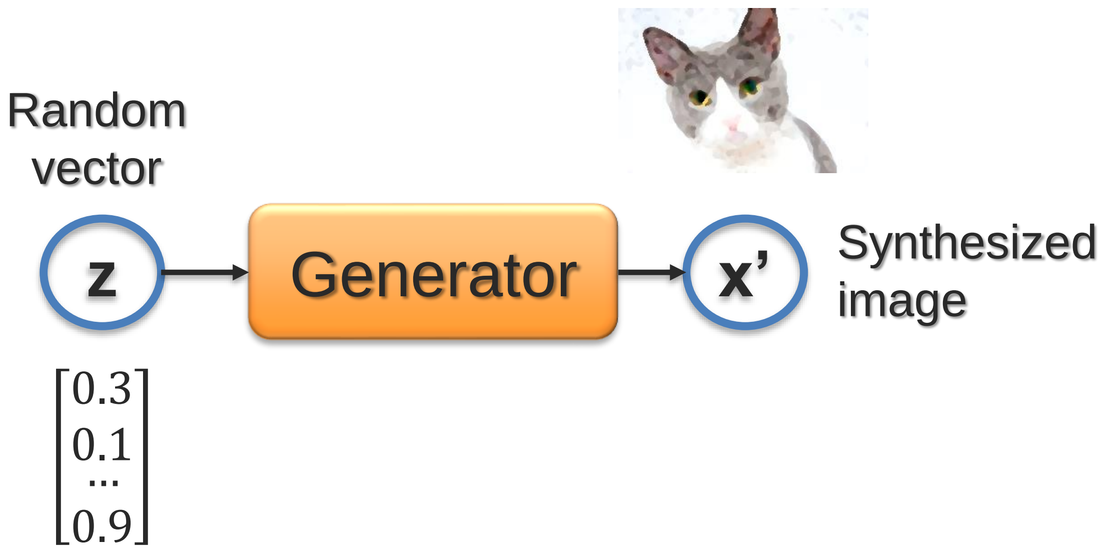
Variational Auto-encoder

- The normal distribution has nice properties.

But these images are not as realistic looking…

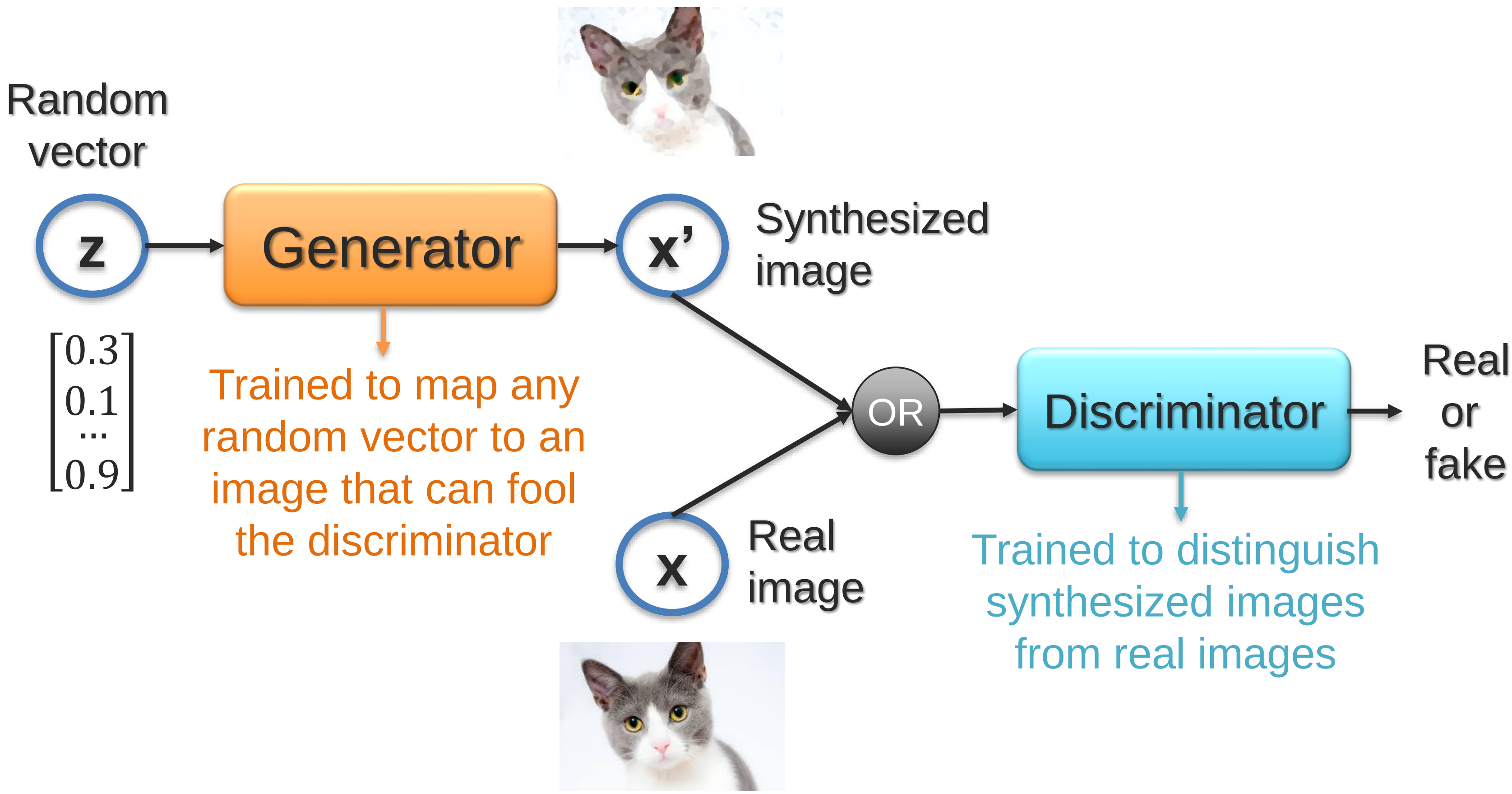
Generative Adversarial Networks

Generative Network



How to train the generator to synthesize realistic images?

Generative Adversarial Network (GAN)



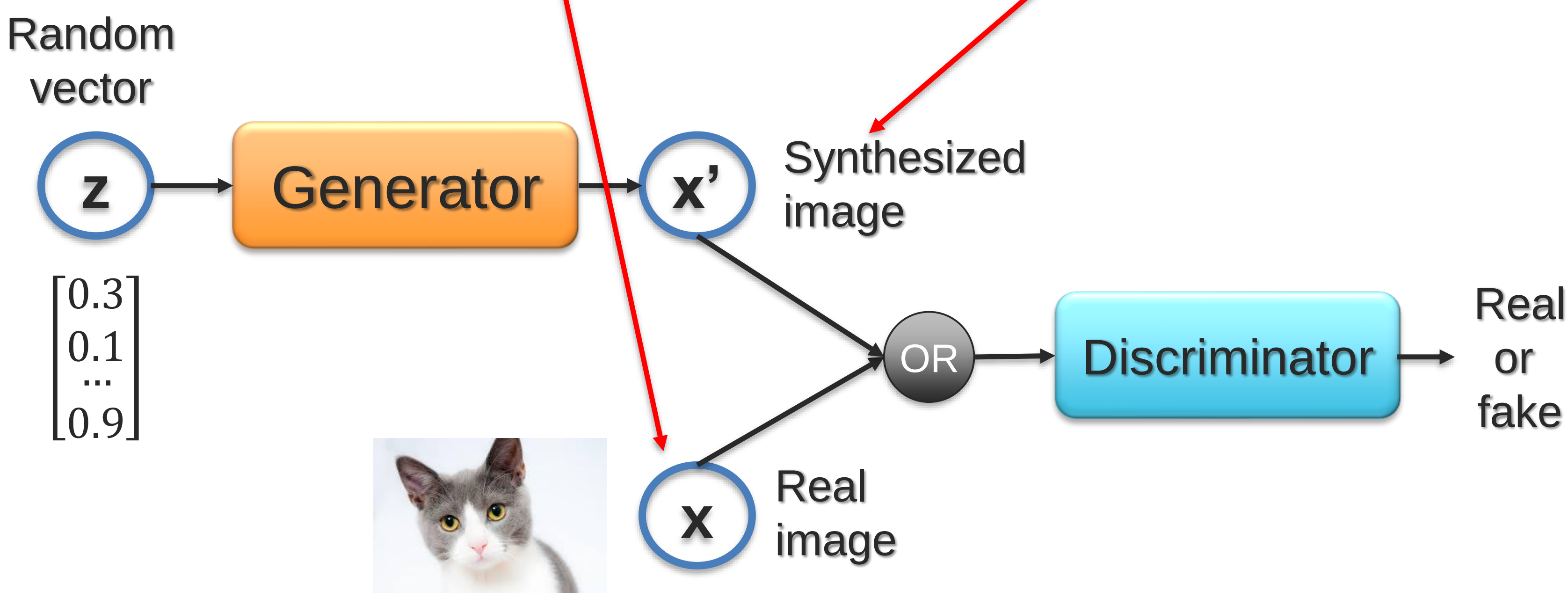
How to train both the generator and the discriminator?

GAN Training

$$\max_{\mathcal{D}} \min_{\mathcal{G}} V(\mathcal{G}, \mathcal{D})$$

How do we optimize this objective function?

$$V(\mathcal{G}, \mathcal{D}) = \mathbb{E}_{p_{data}(\mathbf{x})} \log \mathcal{D}(\mathbf{x}) + \mathbb{E}_{p_g(\mathbf{x})} \log(1 - \mathcal{D}(\mathbf{x}))$$

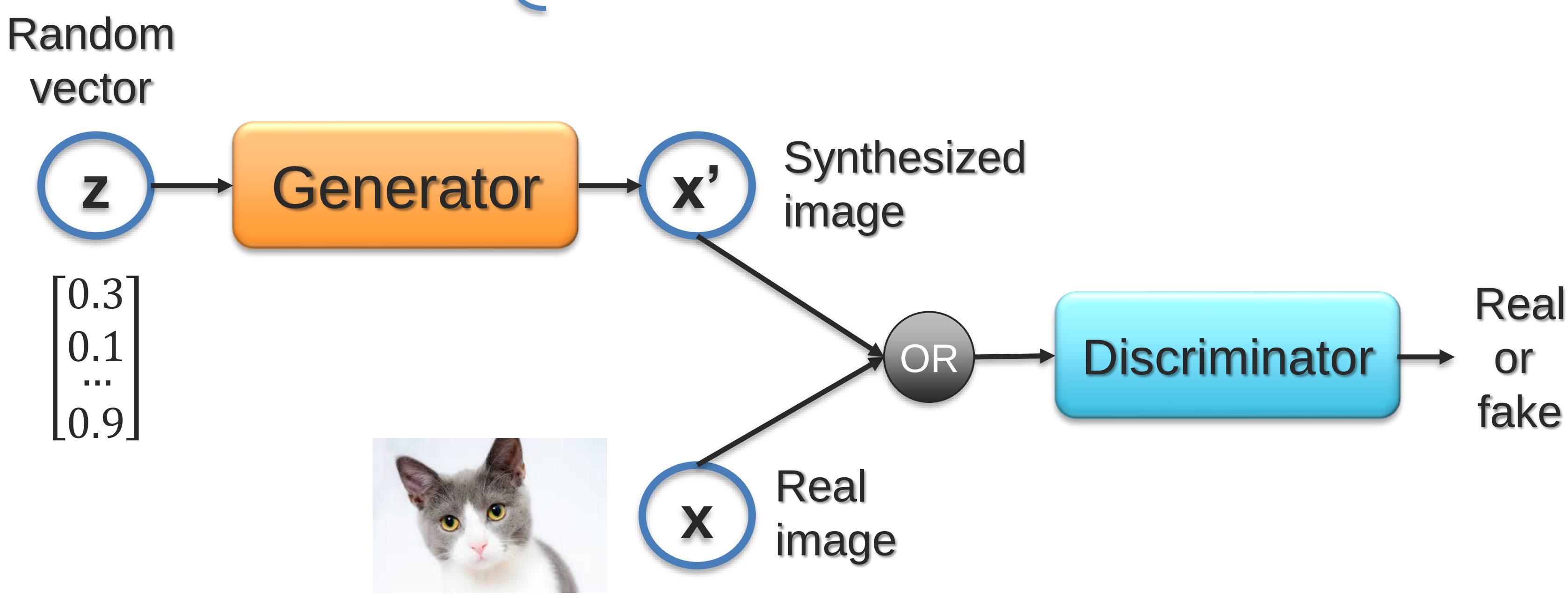


GAN Training

$$\max_{\mathcal{D}} \min_{\mathcal{G}} V(\mathcal{G}, \mathcal{D})$$

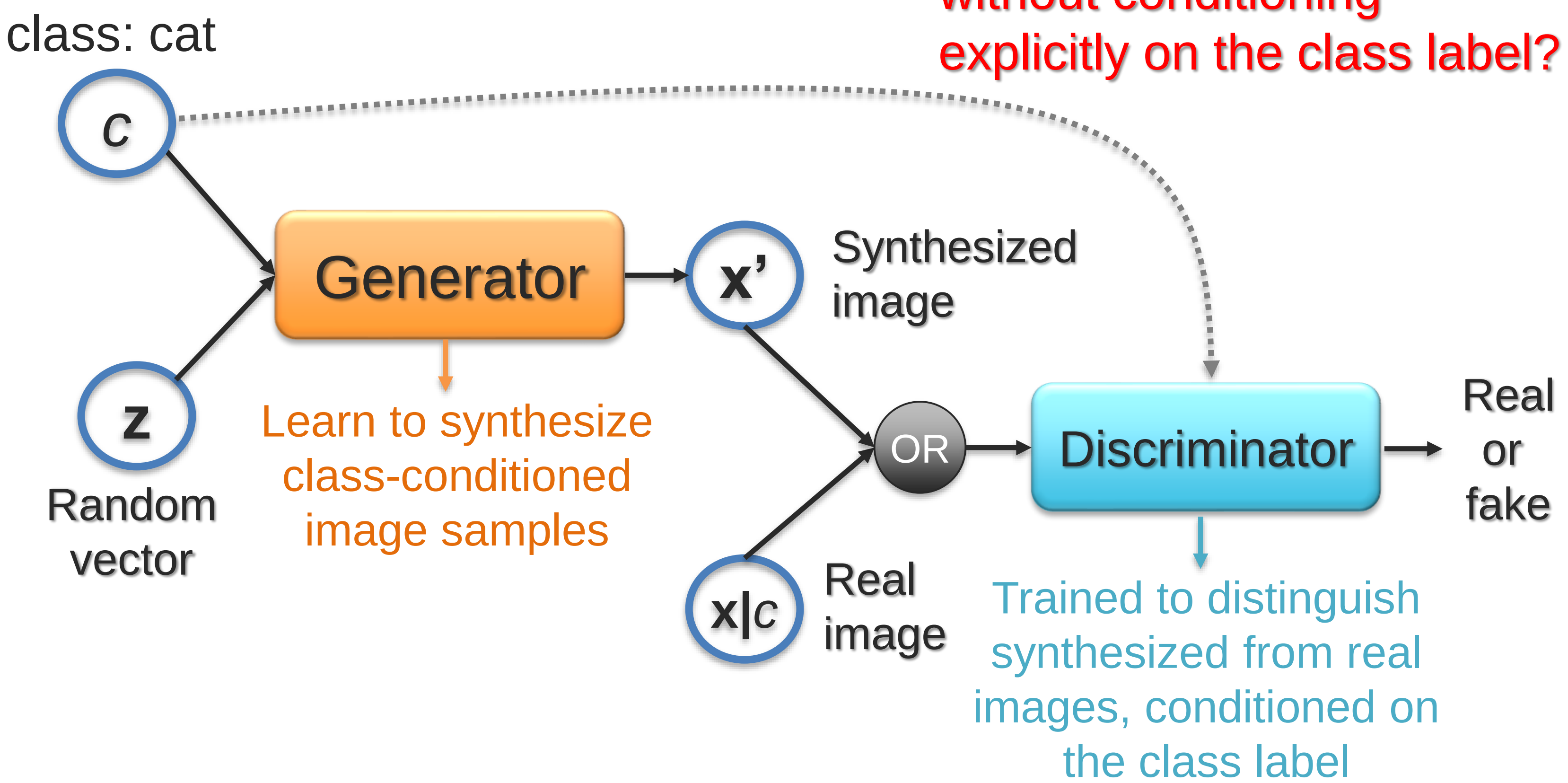
Optimization:

1. Fix generator, and update discriminator
2. Fix discriminator, and update generator

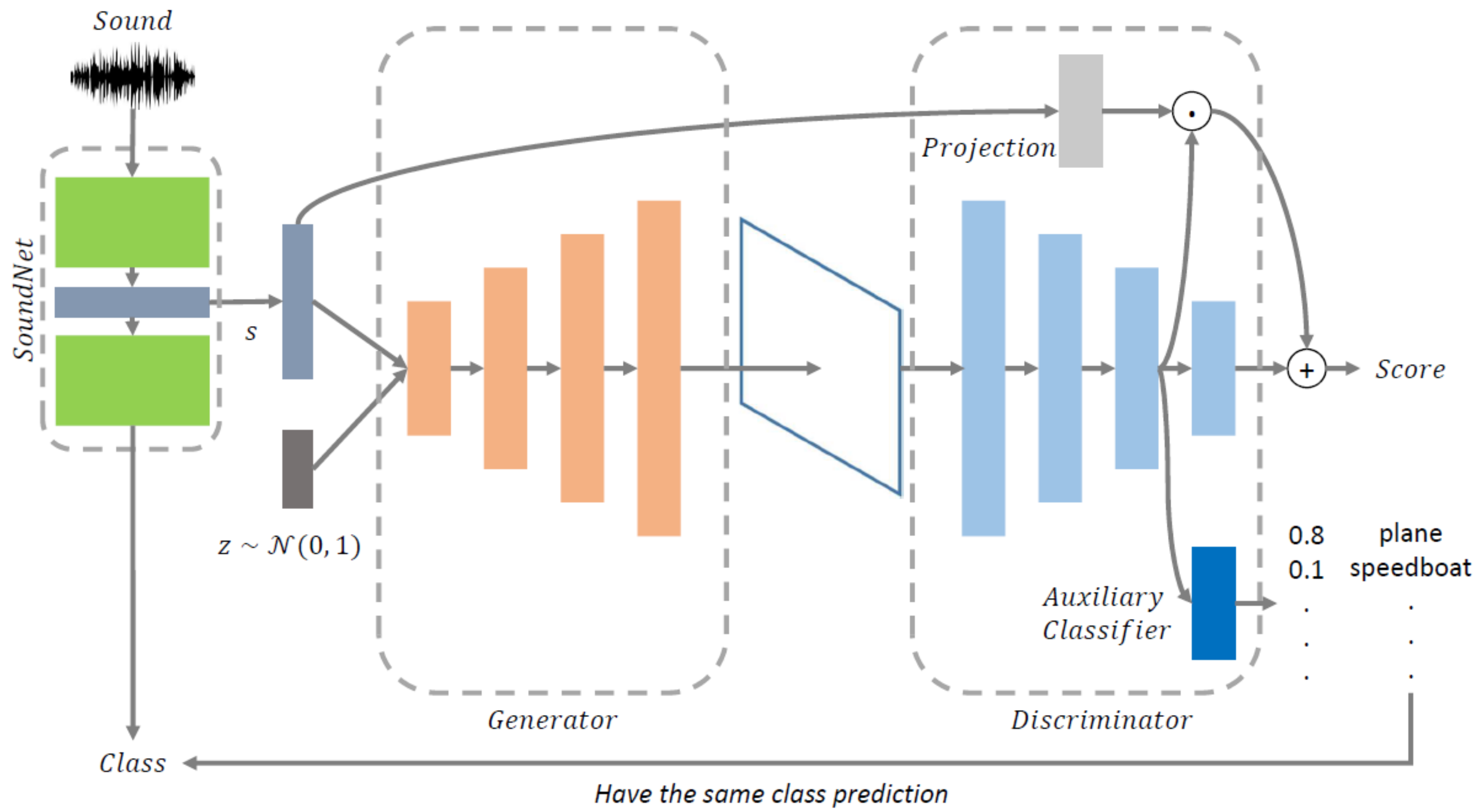


Conditional GAN

How to train discriminator
without conditioning
explicitly on the class label?

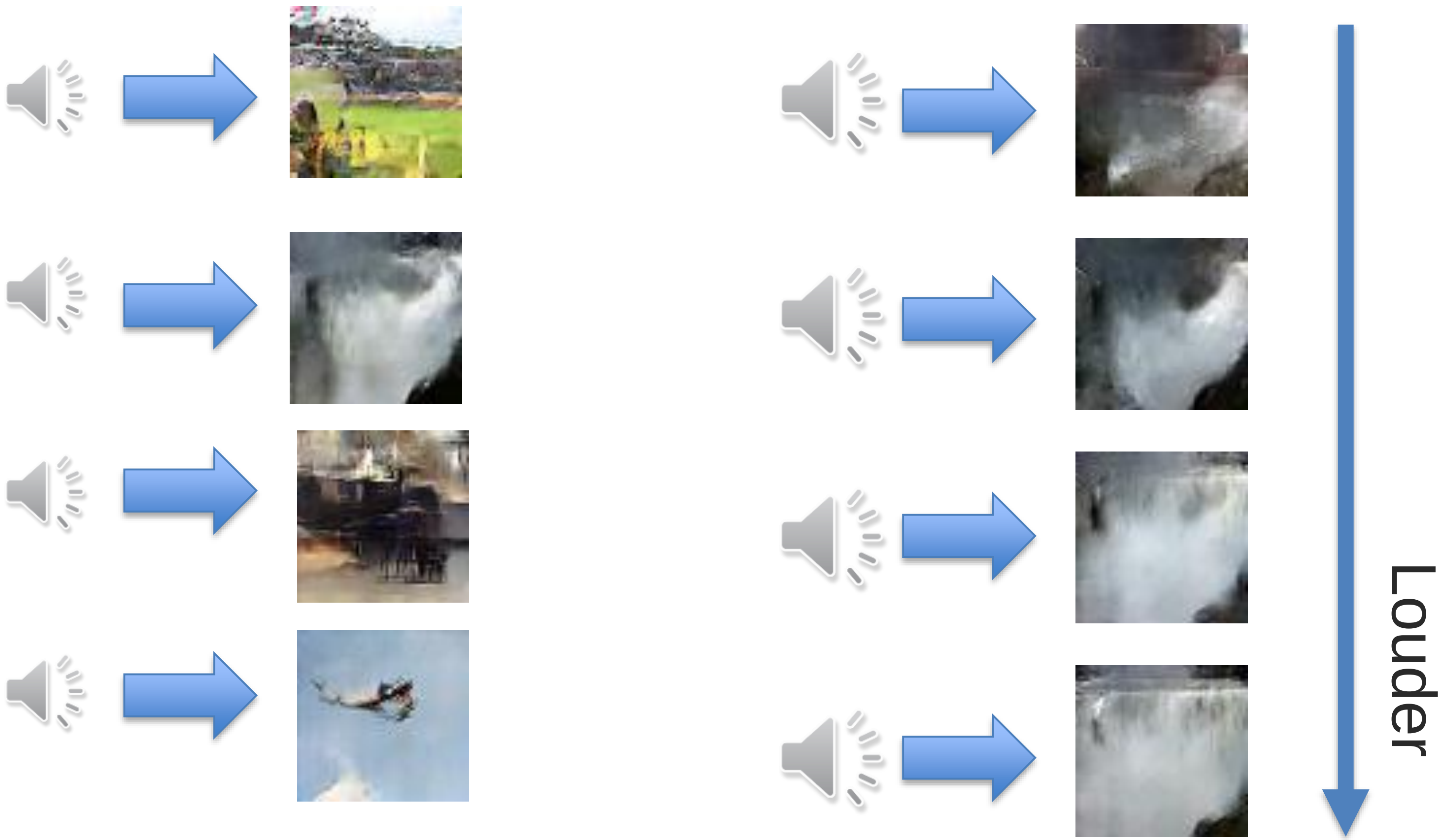


Audio to Scene



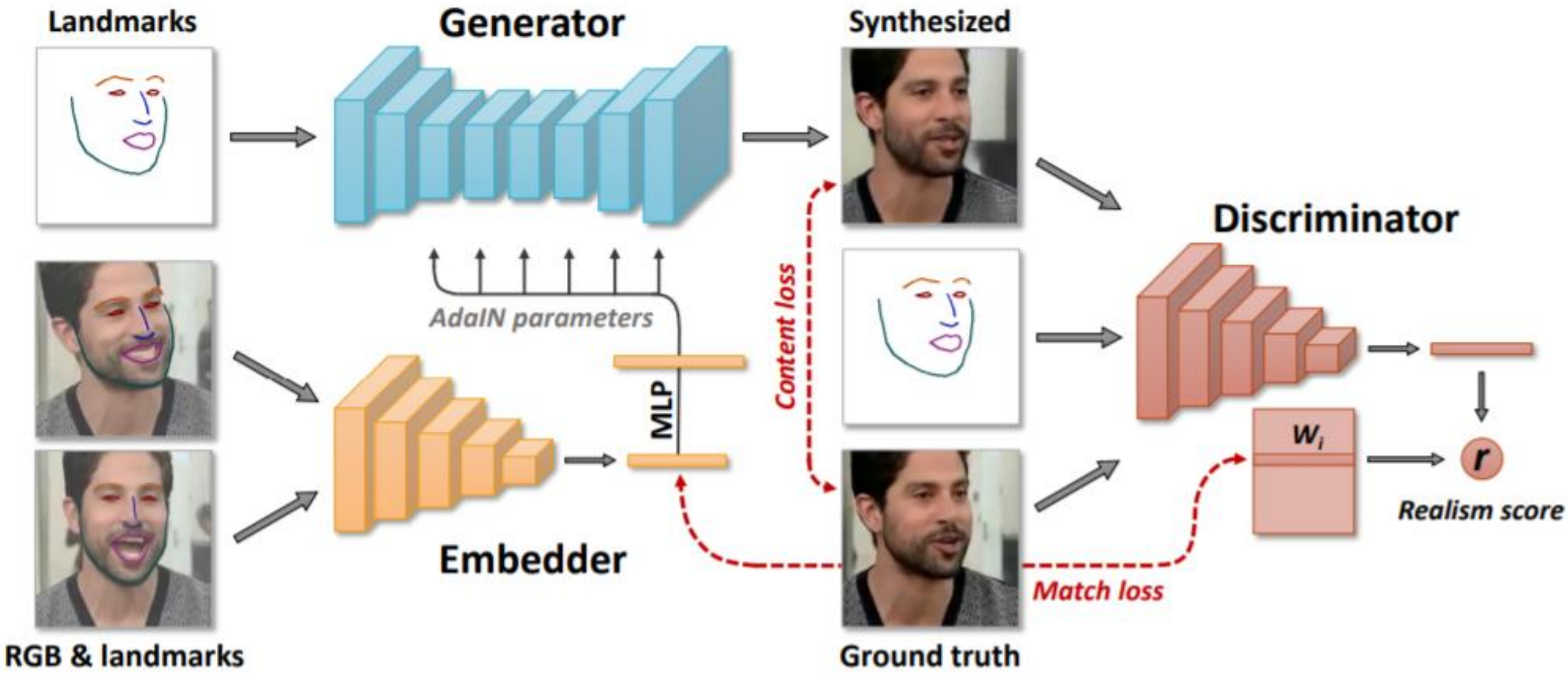
https://wjohn1483.github.io/audio_to_scene/index.html

Audio to Scene



https://wjohn1483.github.io/audio_to_scene/index.html

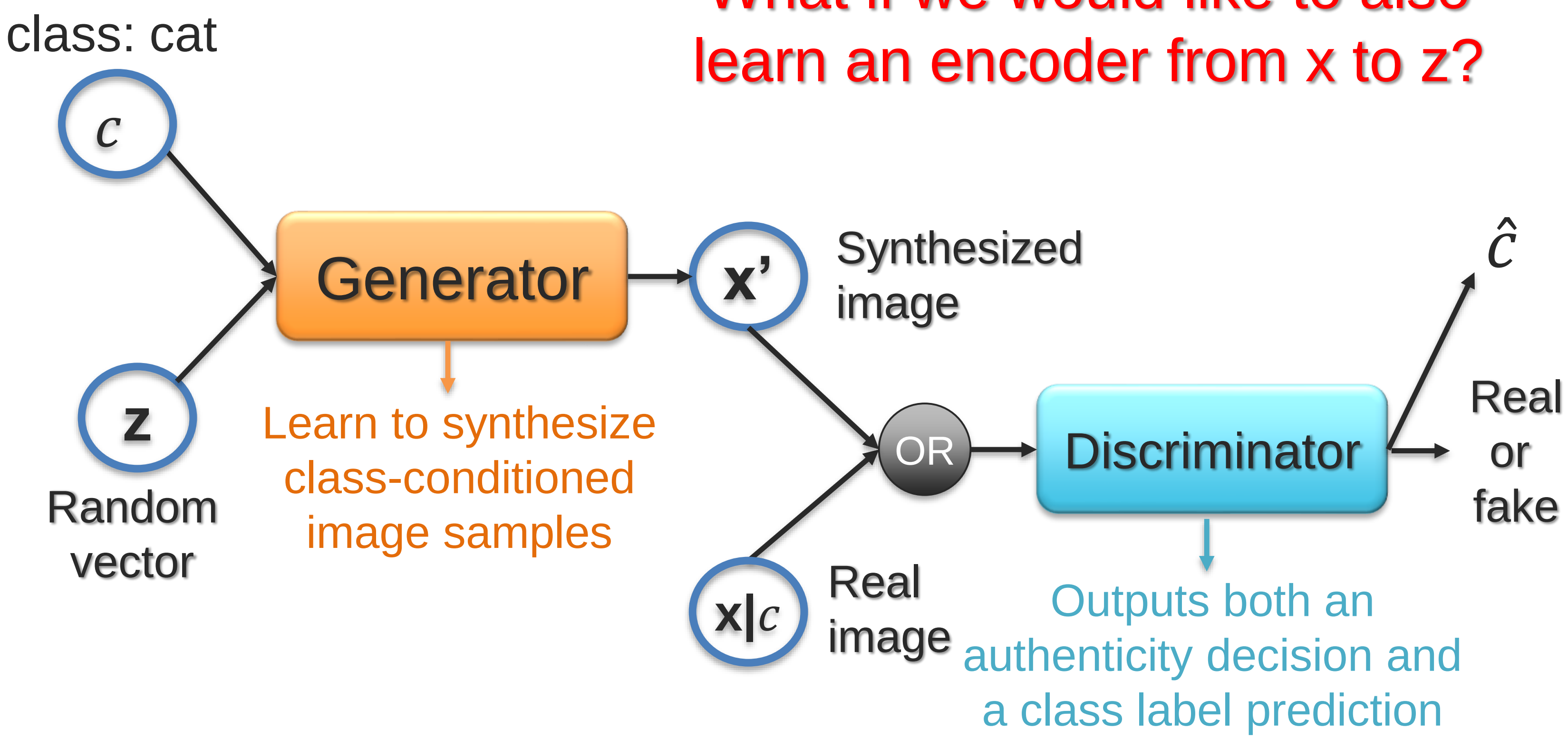
Talking Head



https://arxiv.org/abs/1905.08233

Info GAN

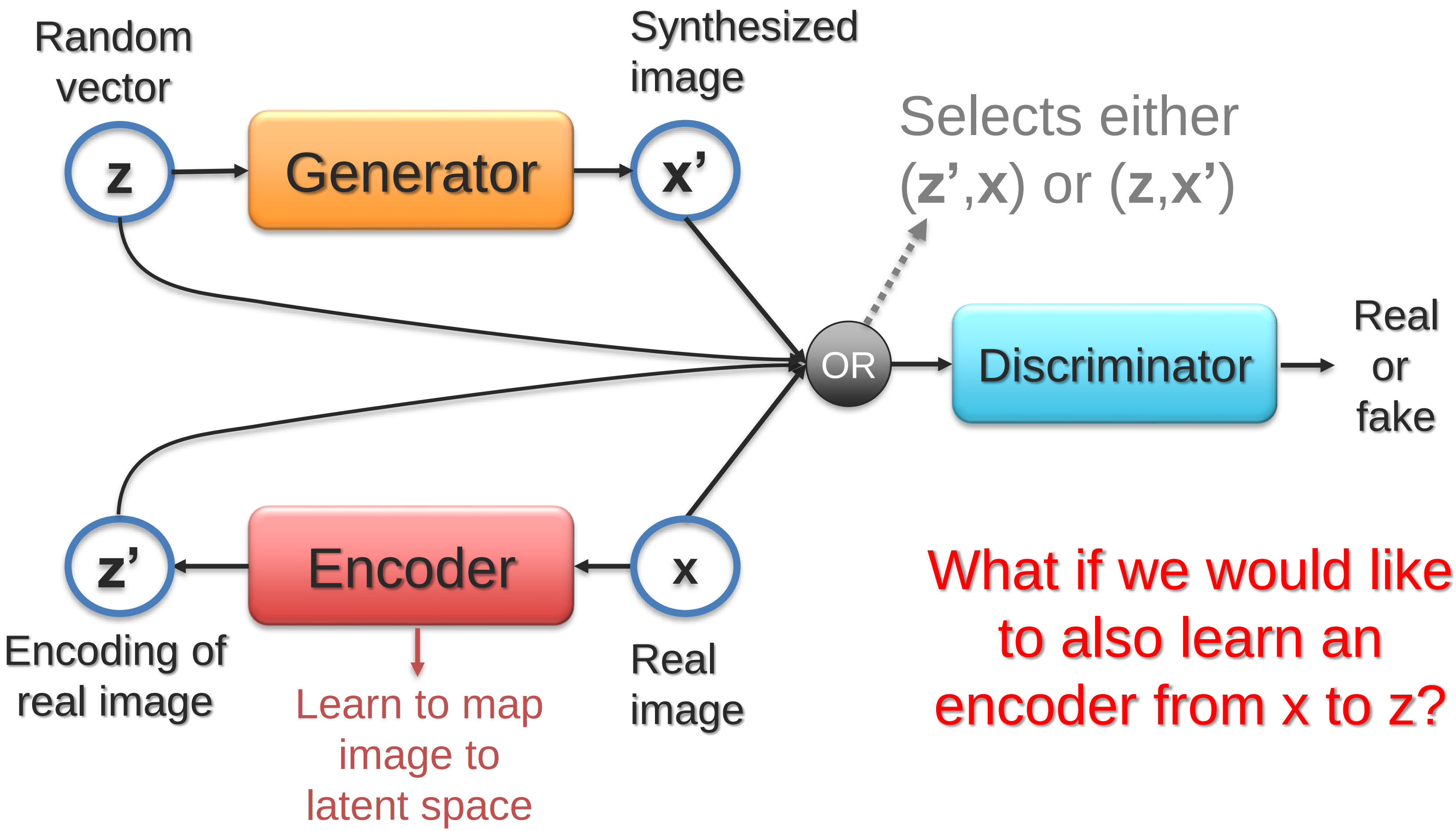
What if we would like to also learn an encoder from x to z?



Talking Head

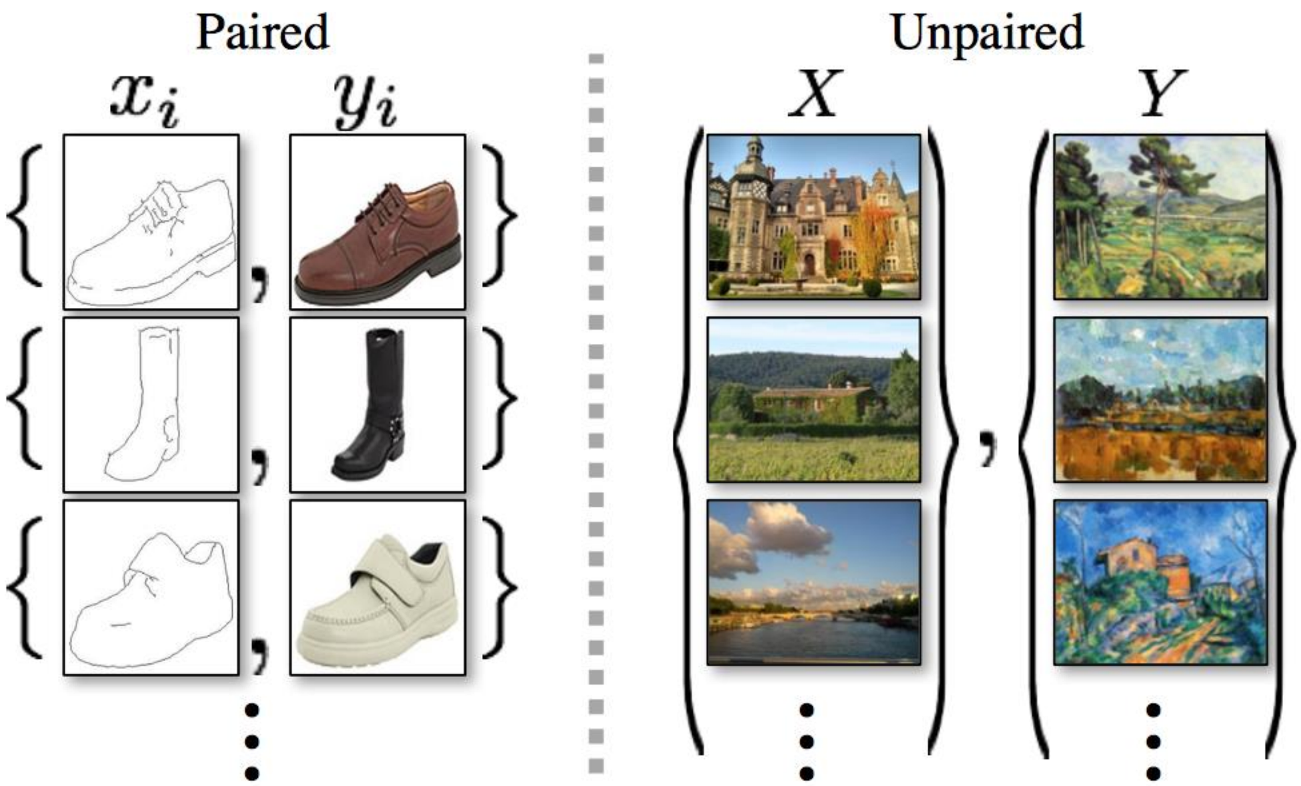
https://arxiv.org/abs/1905.08233

Bidirectional GAN

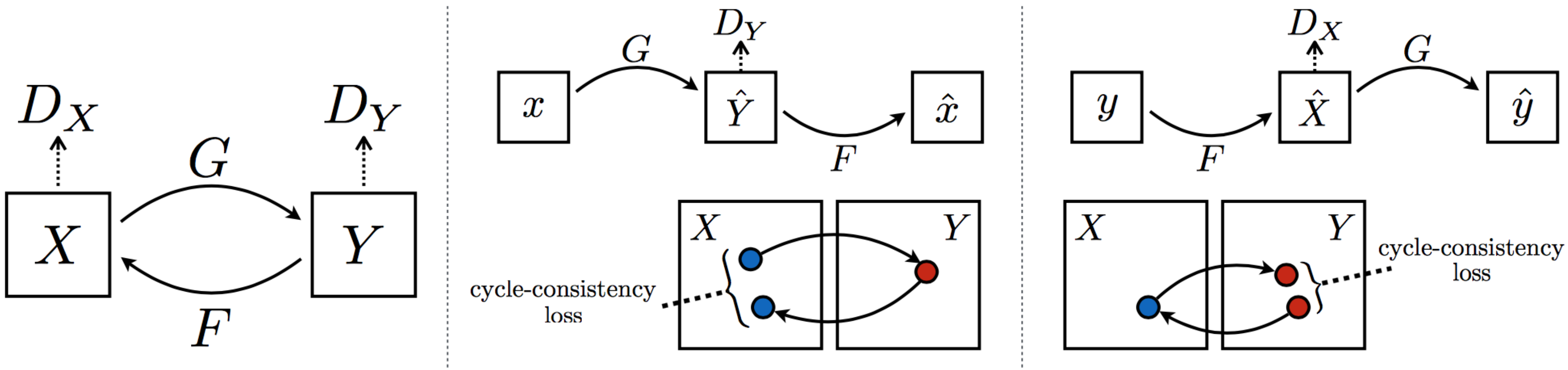


What if we would like to also learn an encoder from x to z?

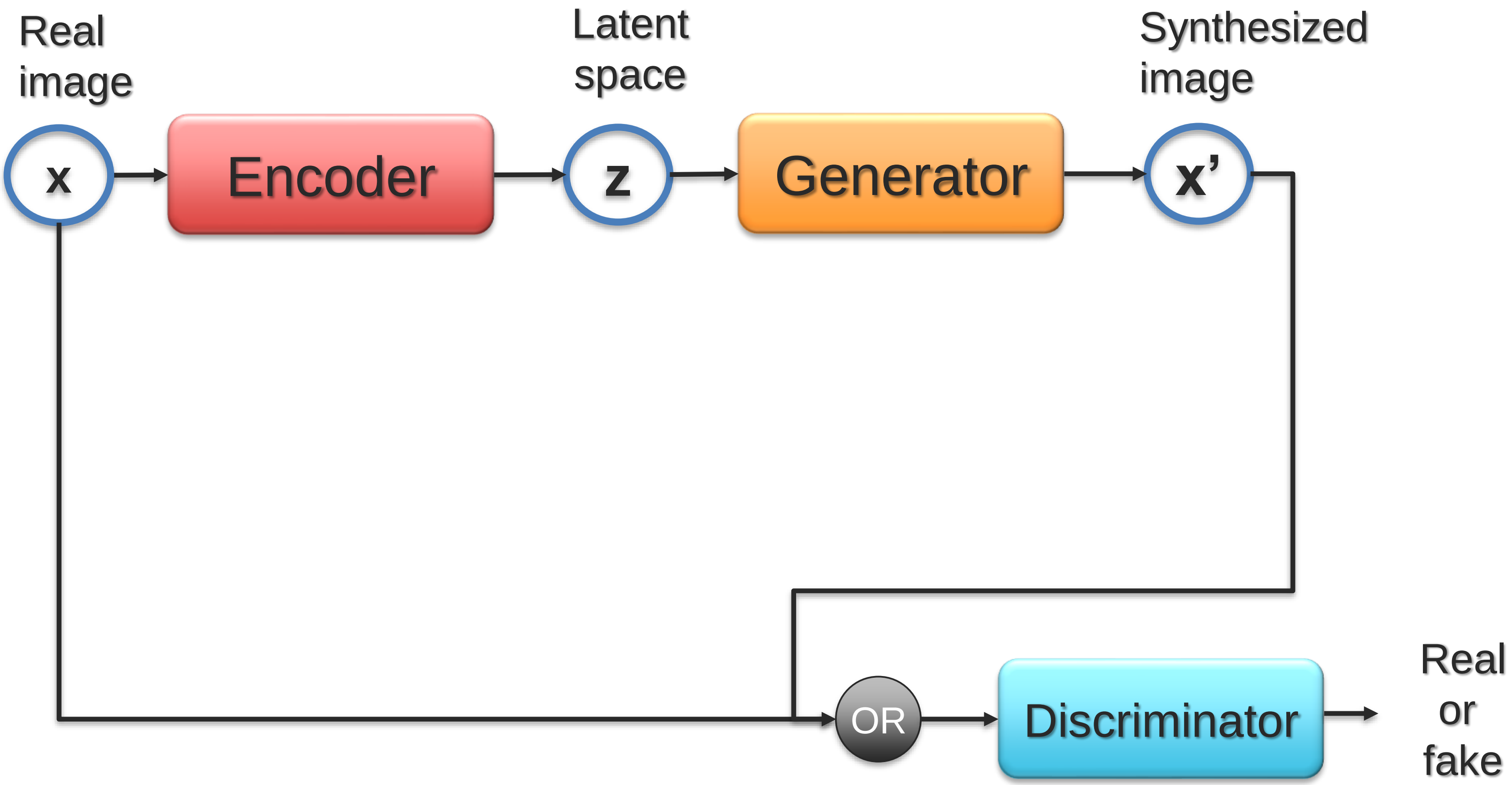
Paired and Unpaired Data



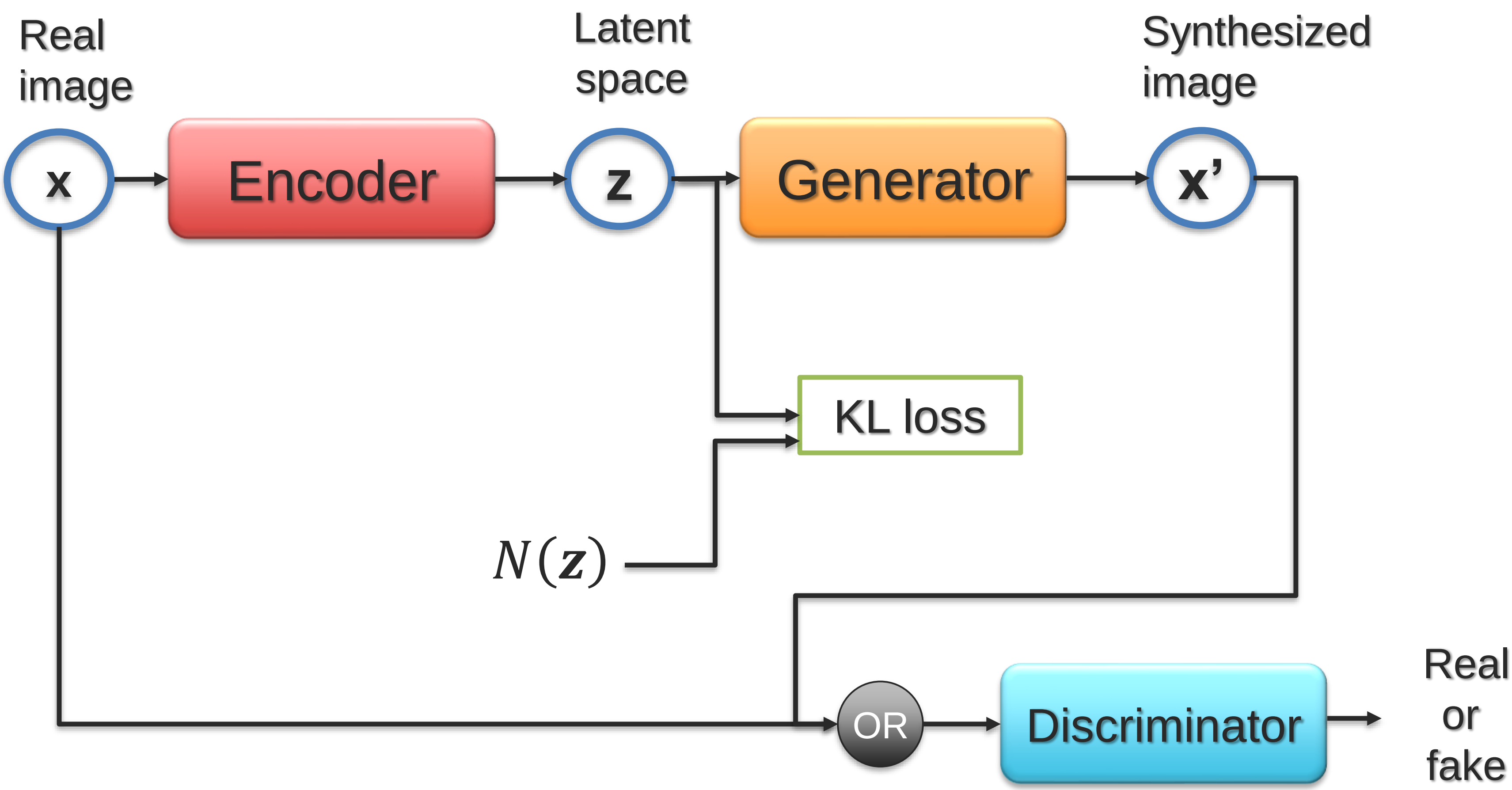
Cycle GAN


D_X
D_Y
G
X
Y
F
D_Y
G
x
$\hat{Y}$
$\hat{x}$
F
X
Y
cycle-consistency
loss
D_X
G
y
$\hat{X}$
$\hat{y}$
F
X
Y
cycle-consistency
loss

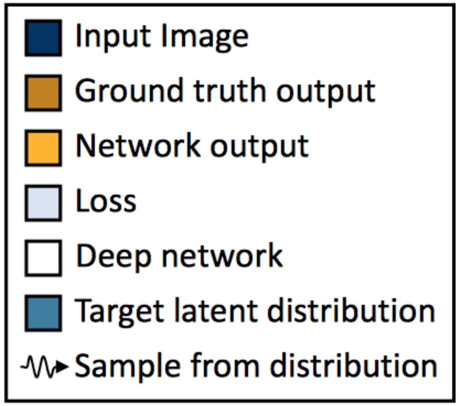
cAE-GAN


Real
image
x
Encoder
Latent
space
z
Generator
Synthesized
image
x’
OR
Discriminator
Real
or
fake

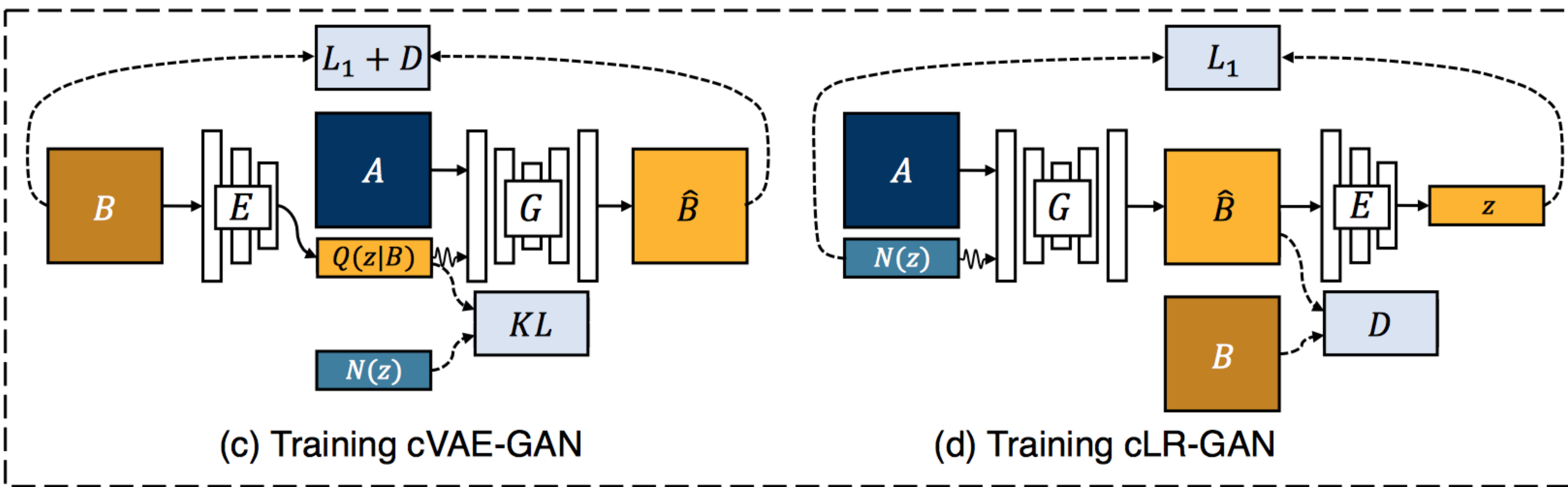
cVAE-GAN

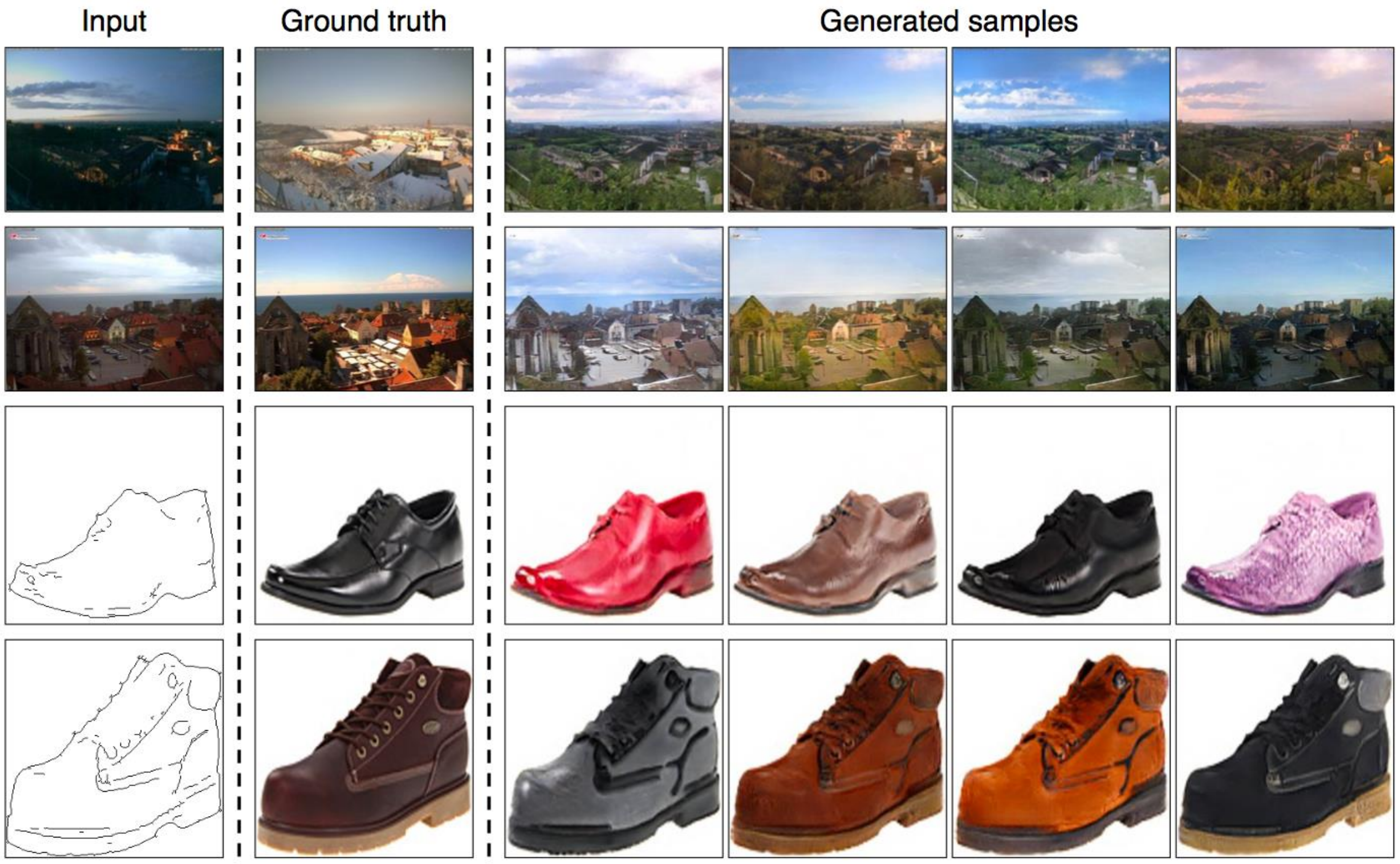

Real
image
Latent
space
Synthesized
image
x
Encoder
z
Generator
x'
KL loss
$N(z)$
OR
Discriminator
Real
or
fake

BiCycle GAN



Let's put everything in one model!!




Input
Ground truth
Generated samples