Main Contributions

- Raise the non-identifiability issues in bidirectional adversarial learning
- Propose ALICE algorithm: a conditional entropy framework to remedy the issues
- Unify ALI/BiGAN, CycleGAN/DiscoGAN/DualGAN and Conditional GAN as joint distribution matching

Non-identifiability issues

Generative Adversarial Networks (GANs)

Marginal distribution matching: \( p(x) = q(x) \)

Adversarially Learned Inference (ALI)

Joint distribution matching: \( p(x, z) = q(x, z) \)

Importantly, universal distribution approximators for the sampling procedure of conditions \( x \sim q_p(x|z) \) and \( z \sim q_z(x|z) \) are carried out as:

\[
\hat{x} = g_{\theta}(z, e), \ z = p(z), \ e \sim \mathcal{N}(0,1), \ \text{and} \ \hat{z} = g_{\phi}(x, \zeta), \ x \sim q(x), \ \zeta \sim \mathcal{N}(0,1).
\]

Issues: The correlation between \( x \) and \( z \) is not specified.

Problem illustration

- In (a), for \( 0 < \delta < 1 \), we can generate "realistic" \( x \) from any sample of \( p(z) \), but with poor reconstruction.
- In (b) \( \delta = 1 \) or (c) \( \delta = 0 \), only one of the solutions will be meaningful in supervised learning.

Many applications require meaningful mappings.

- In unsupervised learning, the inferred latent code can reconstruct its \( x \) itself with high probability \( \delta \to 1 \) or \( \delta \to 0 \).
- In supervised learning, the task-specific correspondence between samples imposes restrictions on the mappings.

### ALICE Algorithms

Adversarially Learned Inference with Conditional Entropy (ALICE)

\[
\min_{\theta, \phi} \max_{\omega, \eta} \mathcal{L}_{ALI}(\theta, \phi, \omega) = \mathcal{L}_{D}(\theta, \phi, \omega) + \mathcal{L}_{C}(\theta, \phi).
\]  

(1)

In unsupervised learning, cycle-consistency is considered to be upperbound CE:

- **Explicit cycle-consistency**
  Prescribed the distribution forms, e.g. \( \ell_1 \)-norm
- **Implicit cycle-consistency**
  Adversarially learned "perfect" reconstruction

\[
\min_{\theta, \phi} \max_{\omega, \eta} \mathcal{L}_{C}(\theta, \phi, \omega) = \mathcal{E}_{\phi}(\theta, \omega) + \mathcal{E}_{\omega}(\theta, \phi) + \mathcal{E}_{\eta}(\theta, \phi) + \mathcal{E}_{\delta}(\theta, \phi).
\]  

(2)

In semi-supervised learning, the pairwise information is leveraged to approximate CE:

- **Explicit mapping**
  Prescribed the forms, \( \ell_1 \)-norm or standard supervised losses
- **Implicit mapping**
  Implicit mapping via conditional GAN

\[
\min_{\theta, \phi} \max_{\omega, \eta} \mathcal{L}_{C}(\theta, \phi, \omega) = \mathcal{E}_{\phi}(\theta, \omega) + \mathcal{E}_{\omega}(\theta, \phi) + \mathcal{E}_{\eta}(\theta, \phi) + \mathcal{E}_{\delta}(\theta, \phi).
\]  

(3)

A Unified Perspective for Bivariate GANs

- **ALI is equivalent to CycleGAN**
  CycleGAN is easier to train, as it decomposes the joint distribution matching objective (as in ALI) into four subproblems.

\[
H^\omega(x|z) = -\mathcal{E}_\phi(\theta: \omega) \log (\sigma(f_{\phi}(x, z))) + \mathcal{E}_{\omega}(\theta: \phi) \log (1 - \sigma(f_{\phi}(x, z)));
\]  

(4)

**Stochastic mapping vs. Deterministic mapping**

Deterministic mappings imply cycle-consistency in theory (as in BigAN), but have practical difficulties. When cycle-consistency is satisfied e.g. optimum of (2), (i) a deterministic mapping enforces the conditionals are matched. (ii) The matched conditionals enforce \( H^\omega(x|z) = 0 \), indicating the mapping becomes deterministic.

**Conditional GAN is doing joint distribution matching**

When the optimum in (3) is achieved, \( \pi(x|z) = q_p(x|z) = q_p(x, z) \).

One can leverage the empirically-defined distributions \( \pi(x|z) \) implied by paired data, to resolve the ambiguity issues in unsupervised bivariate GANs.

### Experiments

**I. Toy dataset**

- Grid search over a set of hyper-parameters for 576 experiments
- (a) True \( x \) and \( z \) reconstruction of \( x \)
- (b) True \( x \) and \( z \) reconstruction of \( z \)
- (c) Inception Score
  - Figure: Generation (c) and reconstruction (d) results on toy data (a,b)

**II. Alice4Alice**

ALICE for painting the cartoon ‘Alice’s Wonderland’, based on edges

- (a) ALICE
- (b) ALI
- (c) Denoising autoencoders
  - Figure: Sampling of \( z \), reconstruction of \( x \) and linear interpolation in \( z \)

ALICE: one pair in each mode is leveraged to resolve ambiguity

**Code**: https://github.com/ChunyuanLI/ALICE