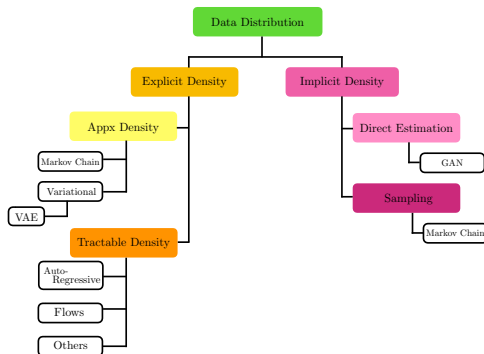


11-695: AI Engineering

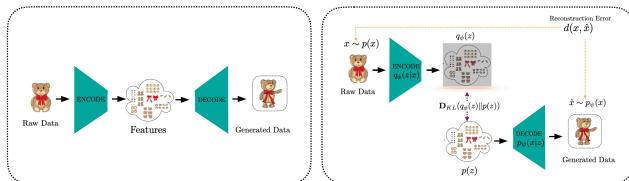
Conditional Generation

Spring 2019

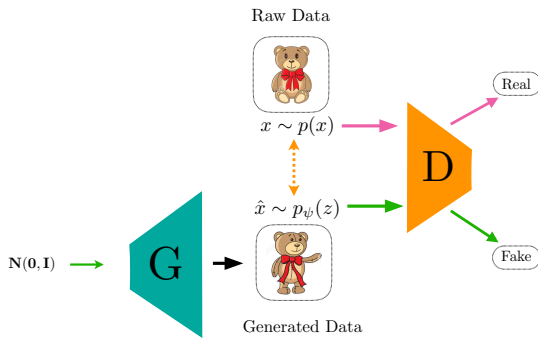
- 1 Generative Models: Review
- 2 Conditional Generation
- 3 Conditional Generative Models
- 4 Conditional Generation as Image/Video Style Transfer
- 5 Conditional Generation as Text Style Transfer
- 6 Cross-Domain Transfer/Translation



- Note: this is not an exhaustive diagram
- We have not covered Markov Chain and Tractable Density methods
 - Tractable: NICE, Glow, MADE, PixelRNN, WaveNet, Masked AutoRegressive Flow, Inverse Autoregressive Flow ..inspired from Goodfellow 2016



- Assume the model is well-trained. For example in MNIST case:
- If we want to generate digit 1, feed image of 1 into Encoder and use Decoder to generate *itself* ???
- Same thing as with other digits: Perfect results are perfect mimics of inputs.
- We can use feature vectors to control the generation but still based on the same "background", e.g. same digit 1



- We have even less control on the generation
- Generator can generate any mode based on sampling
- Upside: it can generate completely new, cool images
- But still how can we learn the mapping of styles or domains?



- The only way to control outputs is to feed ground truths (GT)
- Real case: want to generate new samples from $\mathbf{N}(\mathbf{0}, \mathbf{I})$, we *cannot* control output
- This case: VAE models 10 modes from input distribution, it returns a random mode



- The only way to control outputs is to feed ground truths (GT)
- Real case: want to generate new samples from $\mathbf{N}(\mathbf{0}, \mathbf{I})$, we *cannot* control output
- This case: VAE models 10 modes from input distribution, it returns a random mode
- Why don't we condition on the label (scalar) to generate exactly the mode that we want?

- 1 Generative Models: Review
- 2 Conditional Generation**
- 3 Conditional Generative Models
- 4 Conditional Generation as Image/Video Style Transfer
- 5 Conditional Generation as Text Style Transfer
- 6 Cross-Domain Transfer/Translation

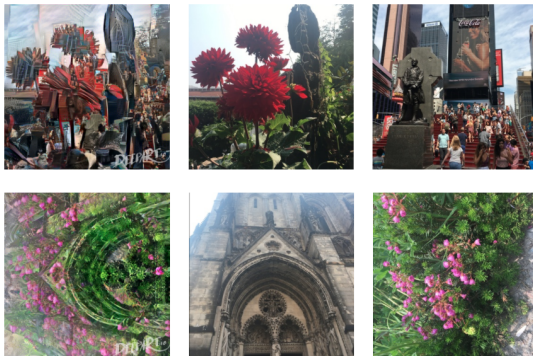
We Want More for Outputs



- Recall from last slide: Can we generate images of digits from other domain, *e.g.* from keyboard input?
- Can we adopt, say Van Gogh's style into images w/o formal drawing training?

Image credit: Gene Kogan

We Want More for Outputs

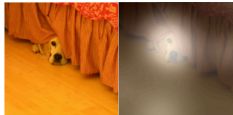


- Can we change the style of input based on an random template?

Different Input and Output Domains?



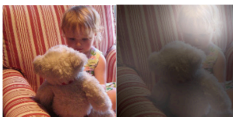
A woman is throwing a frisbee in a park.



A dog is standing on a hardwood floor.



A stop sign is on a road with a mountain in the background.



A little girl sitting on a bed with a teddy bear.



A group of people sitting on a boat in the water.

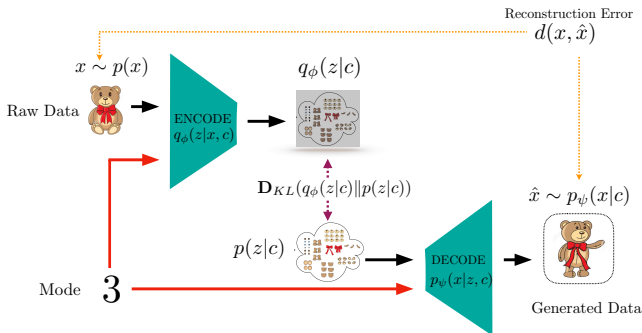


A giraffe standing in a forest with trees in the background.

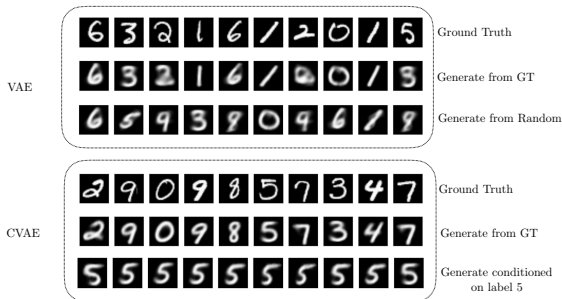
- Can we generate description based on images?
- Or vice versa?

- So far: we generate from a latent representation $\hat{x} = p(x|z)$
- Now, we condition on an attribute a *s.t.* $\hat{x} = p(x|z, a)$
- The latent distribution is now changed from $p(z)$ to $p(z|c)$
- Means that whatever input you have, you add one more input c ,
- for both Encoder and Decoder (Generator).

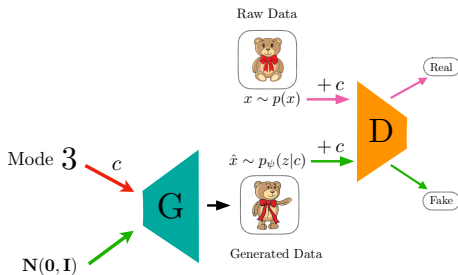
- 1 Generative Models: Review
- 2 Conditional Generation
- 3 Conditional Generative Models**
- 4 Conditional Generation as Image/Video Style Transfer
- 5 Conditional Generation as Text Style Transfer
- 6 Cross-Domain Transfer/Translation



- Recall VAE objective: $\text{ELBO} = \log p(x) - \mathbf{D}_{KL}(q(z) || p(z|x))$
- Now CVAE objective: $\text{ELBO} = \log p(x|c) - \mathbf{D}_{KL}(q(z|c) || p(z|x, c))$



- How to edit VAE to make CVAE:
 - Fuse c and input, *e.g.* as simple as concatenation
 - Embed c into some dimensional space and fuse with input
 - Inputs to Encoder and Decoder change only!
- Results is sharper (we implicitly make use of labels)
- And we can now control the mode of generation

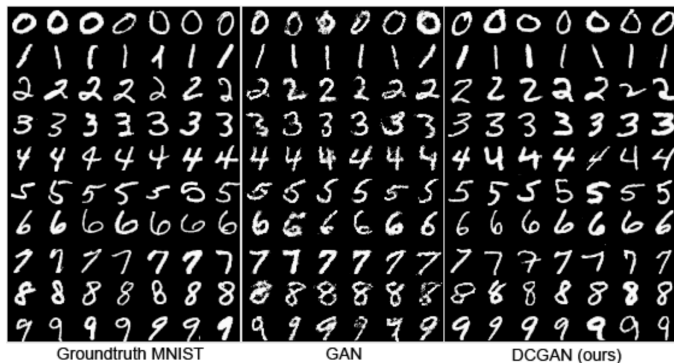


- Recall GAN objective:

$$\min_{\psi} \max_{\theta} [\mathbb{E}_{x \sim p(x)} \log D_{\theta}(x) + (1 - \mathbb{E}_{z \sim p(z)} \log D_{\theta}(G_{\psi}(z)))]$$

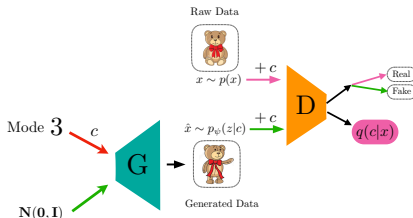
- Now CGAN objective:

$$\min_{\psi} \max_{\theta} [\mathbb{E}_{x \sim p(x)} \log D_{\theta}(x|c) + (1 - \mathbb{E}_{z \sim p(z)} \log D_{\theta}(G_{\psi}(z, c)))]$$



- How to edit VAE to make CGAN: similar to CVAE
- And also, we can now control the mode of generation

Image credit: Alec Radford *et al.* 2016



- D also models distribution $q(c|x)$, a variational appx for $p(c|x)$
- InfoGAN objective: $\min_{\psi} \max_{\theta} [\mathcal{L}_{CGAN} - \lambda I(c, G(x, c))]$ where $\mathcal{L}_{CGAN} = \mathbb{E}_{x \sim p(x)} \log D_{\theta}(x|c) + (1 - \mathbb{E}_{z \sim p(z)} \log D_{\theta}(G_{\psi}(z, c)))$
- Variational Mutual Information:

$$I(c, G(z, c)) = H(c) - H(c|G(z, c)) \quad (1)$$

$$= \mathbb{E}_{x \sim G(z, c)} [\mathbb{E}_{c' \sim P(c|x)} [\log P(c'|x)]] + H(c) \quad (2)$$

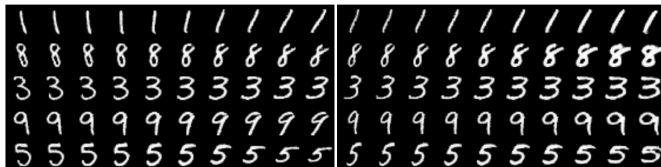
$$\geq \mathbb{E}_{x \sim G(z, c)} [\mathbb{E}_{c' \sim P(c|x)} [\log Q(c'|x)]] + H(c) \quad (3)$$

Model: Xi Chen *et al.* 2016



(a) Varying c_1 on InfoGAN (Digit type)

(b) Varying c_1 on regular GAN (No clear meaning)



(c) Varying c_2 from -2 to 2 on InfoGAN (Rotation)

(d) Varying c_3 from -2 to 2 on InfoGAN (Width)

- Latent Representation can be disentangled
- $c_1 \sim \text{Cat}(K = 10, p = 0.1)$, and $c_2, c_3 \sim U[-1, 1]$

Image credit: Xi Chen *et al.* 2016

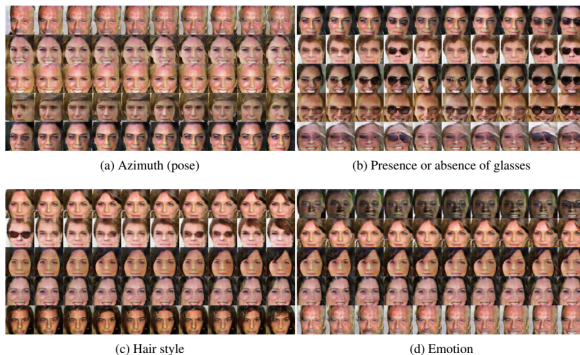
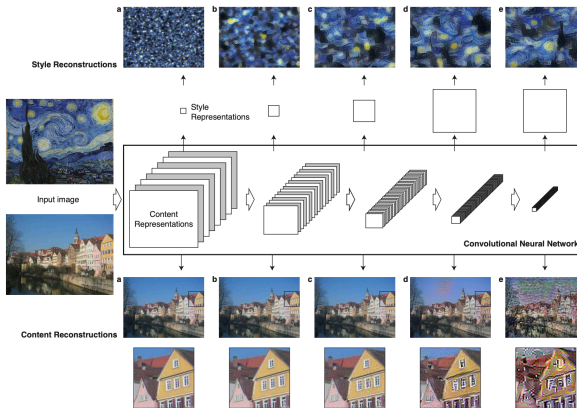
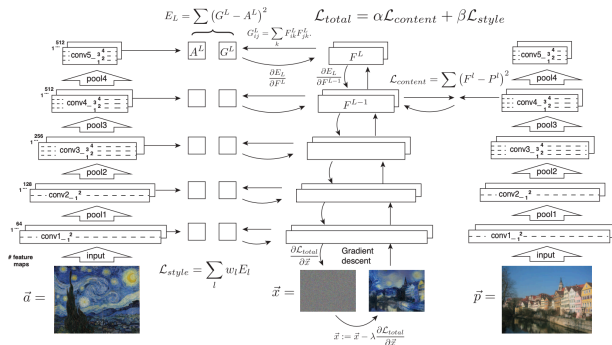


Figure 6: **Manipulating latent codes on CelebA:** (a) shows that a categorical code can capture the azimuth of face by discretizing this variation of continuous nature; in (b) a subset of the categorical code is devoted to signal the presence of glasses; (c) shows variation in hair style, roughly ordered from less hair to more hair; (d) shows change in emotion, roughly ordered from stern to happy.

- 1 Generative Models: Review
- 2 Conditional Generation
- 3 Conditional Generative Models
- 4 Conditional Generation as Image/Video Style Transfer
- 5 Conditional Generation as Text Style Transfer
- 6 Cross-Domain Transfer/Translation



- Style Image and Input Image have features extracted at each CNN layer of VGG

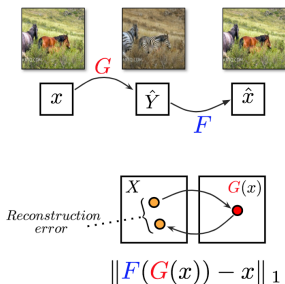


- White noise x is modeled into 2 components: content and style
- Each will have the respective loss
- Both losses are optimized together

Image credit: Leon A. Gatys *et al.* 2016 (Image Style Transfer Using Convolutional Neural Networks)



- Training converges when x matches content and style

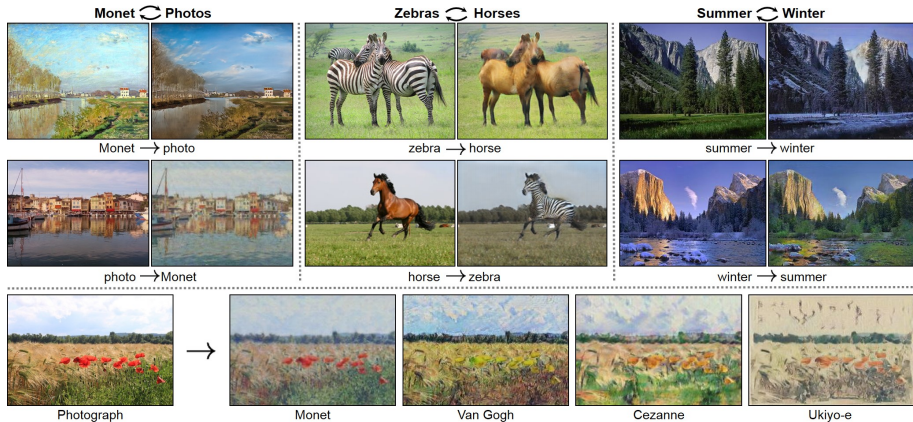


- Cycle loss is based on 2 phases: forward and back transfers
- Provides symmetric alignment intuition



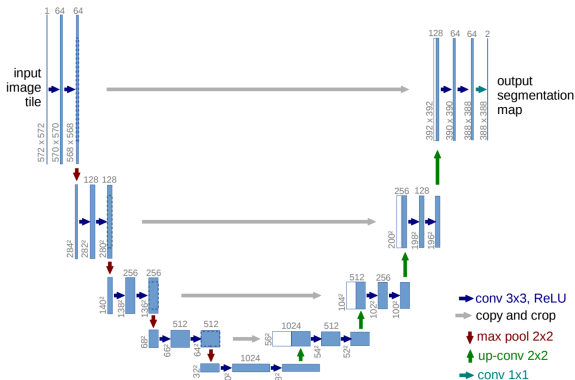
- 1 Generator,
- 2 Discriminators (1 per each style).
- Augmented by cycle loss

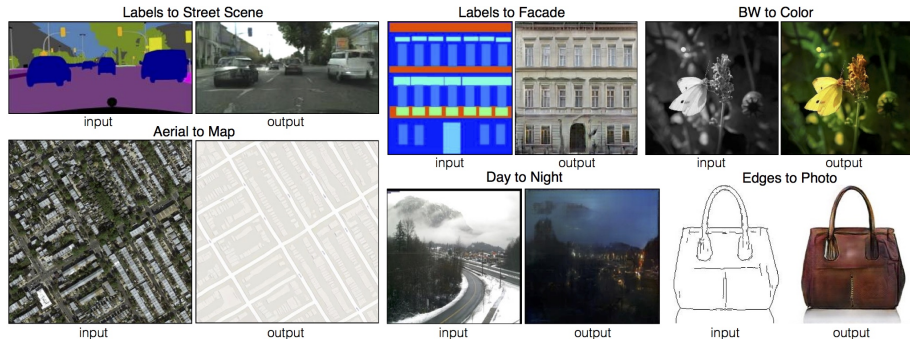
Image credit: Jun-Yan Zhu *et al.* 2017 (Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks)



- Is known as unpaired image-to-image translation

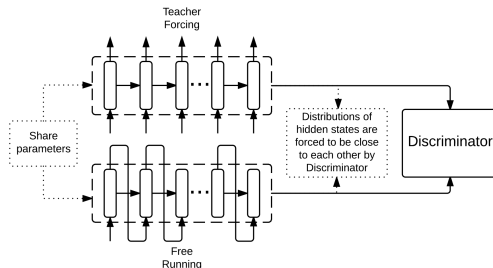
Image credit: Jun-Yan Zhu *et al.* 2017 (Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks)





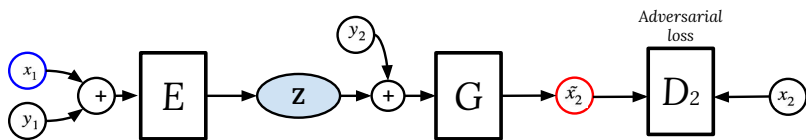
- Use CGAN to transfer styles
- Supervised by labels

- 1 Generative Models: Review
- 2 Conditional Generation
- 3 Conditional Generative Models
- 4 Conditional Generation as Image/Video Style Transfer
- 5 Conditional Generation as Text Style Transfer
- 6 Cross-Domain Transfer/Translation



- Can not use GAN directly for text domain for discreteness
- RNN is a go-to solution when dealing with this domain
- Still want to apply adversarial training:
 - **G**: RNN as usual
 - **D**: classify *hidden states output* between free-run and teacher-forcing modes

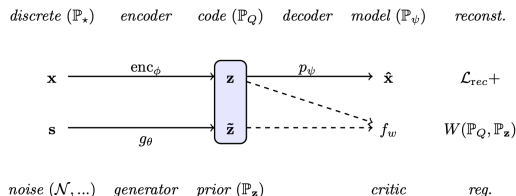
Image credit: Anirudh Goyal, Alex Lamb *et al.* 2016 (Professor Forcing: A New Algorithm for Training Recurrent Networks)



- Encoder + Generator = enc-RNN + dec-RNN = Seq2Seq
 - Encode: (text, style) = (x_i, y_i)
 - Embedded Representation: (z_{ci}, z_{si}) , drop style z_{si} ,
 - combined with new style for decoding: (z_{ci}, y_j) , $j = i$ or $j \neq i$
 - Result: $\hat{x}_j$ (we have transferred from style i to j).
- Professor-Forcing between 2 styles:
 - D_1 : distinguish $\hat{x}_1|z_{c1}, y_1$ and $\hat{x}_1|z_{c2}, y_1$
 - D_2 : distinguish $\hat{x}_2|z_{c2}, y_2$ and $\hat{x}_2|z_{c1}, y_2$

From negative to positive
consistently slow . consistently good . consistently fast .
my goodness it was so gross . my husband 's steak was phenomenal . my goodness was so awesome .
it was super dry and had a weird taste to the entire slice . it was a great meal and the tacos were very kind of good . it was super flavorful and had a nice texture of the whole side .
From positive to negative
i love the ladies here ! i avoid all the time ! i hate the doctor here !
my appetizer was also very good and unique . my bf was n't too pleased with the beans . my appetizer was also very cold and not fresh whatsoever .
came here with my wife and her grandmother ! came here with my wife and hated her ! came here with my wife and her son .

Table 3: Sentiment transfer samples. The first line is an input sentence, the second and third lines are the generated sentences after sentiment transfer by Hu et al. (2017) and our cross-aligned auto-encoder, respectively.



- EncoderRNN: $enc_\phi(z|x) = \mathbb{P}_Q$ (real), DecoderRNN: $p_\psi(\hat{x}|z)$
- Generator from noise $s \sim \mathbf{N}(\mathbf{0}, \mathbf{I})$: $g_\theta(\tilde{z}|s) = \mathbb{P}_z$ (fake)
- Discriminator: real vs. fake using Wasserstein distance W between Encoder distribution $\mathbb{P}_Q$ and prior distribution $\mathbb{P}_z$
- Objective: $\min_{\phi, \psi} [\mathcal{L}_{rec}(\phi, \psi)) + \lambda W(\mathbb{P}_Q, \mathbb{P}_z)]$

Image credit: Jake (Junbo) Zhao *et al.* 2018 (Adversarially Regularized Autoencoders)

Positive great indoor mall .
⇒ ARAE no smoking mall .
⇒ Cross-AE terrible outdoor urine .

Positive it has a great atmosphere , with wonderful service .
⇒ ARAE it has no taste , with a complete jerk .
⇒ Cross-AE it has a great horrible food and run out service .

Positive we came on the recommendation of a bell boy and the food was amazing .
⇒ ARAE we came on the recommendation and the food was a joke .
⇒ Cross-AE we went on the car of the time and the chicken was awful .

Negative hell no !
⇒ ARAE hell great !
⇒ Cross-AE incredible pork !

Negative small , smokey , dark and rude management .
⇒ ARAE small , intimate , and cozy friendly staff .
⇒ Cross-AE great , , , chips and wine .

Negative the people who ordered off the menu did n't seem to do much better .
⇒ ARAE the people who work there are super friendly and the menu is good .
⇒ Cross-AE the place , one of the office is always worth you do a business .

Science what is an event horizon with regards to black holes ?
⇒ Music what is your favorite sitcom with adam sandler ?
⇒ Politics what is an event with black people ?

Science take 1ml of hcl (concentrated) and dilute it to 50ml .
⇒ Music take em to you and shout it to me
⇒ Politics take bribes to islam and it will be punished .

Science just multiply the numerator of one fraction by that of the other .
⇒ Music just multiply the fraction of the other one that 's just like it .
⇒ Politics just multiply the same fraction of other countries .

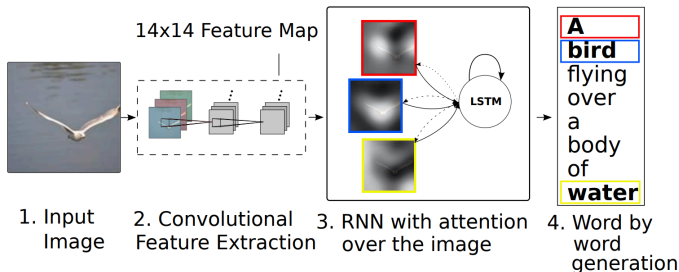
Music do you know a website that you can find people who want to join bands ?
⇒ Science do you know a website that can help me with science ?
⇒ Politics do you think that you can find a person who is in prison ?

Music all three are fabulous artists , with just incredible talent ! !
⇒ Science all three are genetically bonded with water , but just as many substances ,
 are capable of producing a special case .
⇒ Politics all three are competing with the government , just as far as i can .

Music but there are so many more i can 't think of !
⇒ Science but there are so many more of the number of questions .
⇒ Politics but there are so many more of the can i think of today .

- 1 Generative Models: Review
- 2 Conditional Generation
- 3 Conditional Generative Models
- 4 Conditional Generation as Image/Video Style Transfer
- 5 Conditional Generation as Text Style Transfer
- 6 Cross-Domain Transfer/Translation

- So far: Style/Attribute transfer within the same domain: Image or Text
- Cross-Domain Transfer/Translation:
 - Image-to-Text: Image Captioning
 - Text-to-Image: Image Synthesis
 - Text-to-Audio: TTS (Text to Speech)
 - Audio-to-Text: ASR (Automatic Speech Recognition)
 - Any pair of modalities



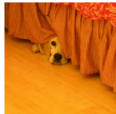
- Still Encoder-Decoder model
 - Encode images by CNN
 - Decode to text by RNN
 - Attention to align spatial to sequential space
- Translate Image to Text, or generate Text conditioned on Image

Image credit: Kevin Xu *et al.* 2016 (Show, Attend and Tell: Neural Image Caption Generation with Visual Attention)

Image Captioning: Show-Attend-Tell



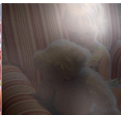
A woman is throwing a frisbee in a park.



A dog is standing on a hardwood floor.



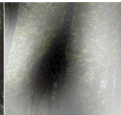
A stop sign is on a road with a mountain in the background.



A little girl sitting on a bed with a teddy bear.



A group of people sitting on a boat in the water.



A giraffe standing in a forest with trees in the background.

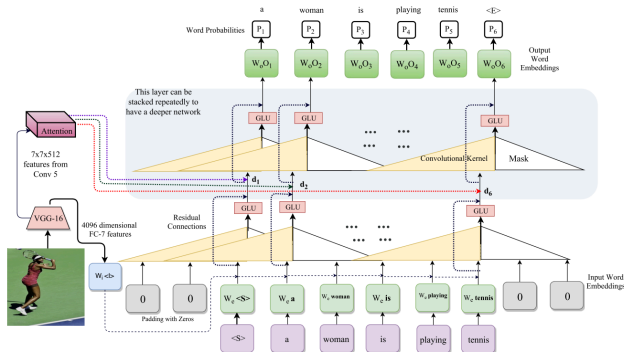


Figure 4: Our convolutional architecture for image captioning. Our architecture has four components: (i) Input embedding layer, (ii) Image embedding, (iii) Convolutional module and (iii) Output embedding layer. Details of these components can be found in Section 5.

Convolutional Image Captioning



LSTM: a man and a woman in a suit and tie

CNN: a black and white photo of a man and woman in a suit

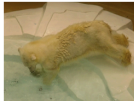
GT: A man sitting next to a woman while wearing a suit.



LSTM: a man is walking down a path in the woods

CNN: a truck is parked in the dirt near a tree

GT: a truck is parked at a campground with snow on it



LSTM: a cat is laying down on a bed

CNN: a polar bear is drinking water from a white bowl

GT: A white polar bear laying on top of a pool of water



LSTM: a bear is standing on a rock in a zoo

CNN: two bears are walking on a rock in the zoo

GT: two bears touching noses standing on rocks



LSTM: a box of donuts with a variety of toppings

CNN: a box of doughnuts with sprinkles and a sign

GT: A bunch of doughnuts with sprinkles on them



LSTM: a dog is standing in the grass near a tree

CNN: a dog is standing in the grass with a frisbee in its mouth

GT: large dog retrieving the frisbee for his owner



LSTM: a parking meter with a sign on it

CNN: a doll is sitting next to a parking meter

GT: A doll with articulated joints stares from her perch between two parking meters.



LSTM: a boat is docked in the water near a dock

CNN: a boat is docked in the water near a mountain

GT: A white boat floating on a lake under mountains



LSTM: a cat is standing on top of a refrigerator

CNN: a cat is standing in a kitchen looking at the camera

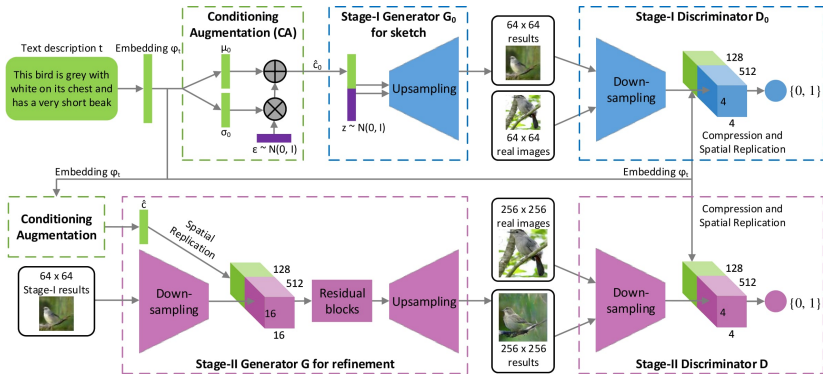
GT: A cat in a kitchen on top of a refrigerator



LSTM: a dog and a dog in a field

CNN: two cows are standing in a field of grass

GT: A dog and a horse standing near each other



- Two-phase generation (relate to Progressive GAN?)

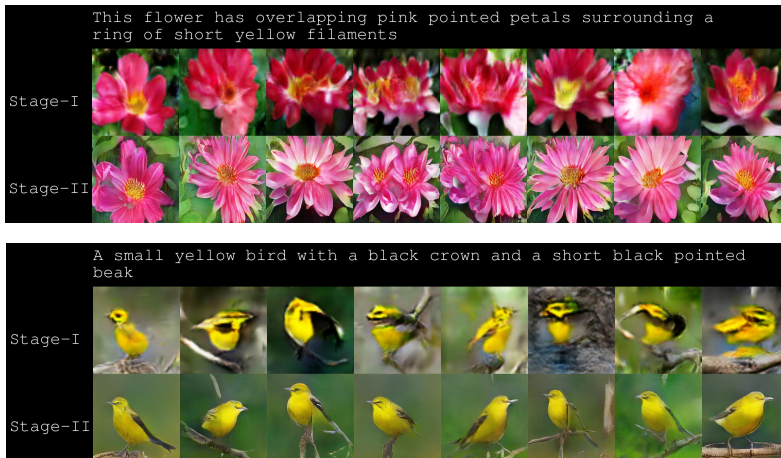


Image credit: Han Zhang *et al.* 2017 (StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks)

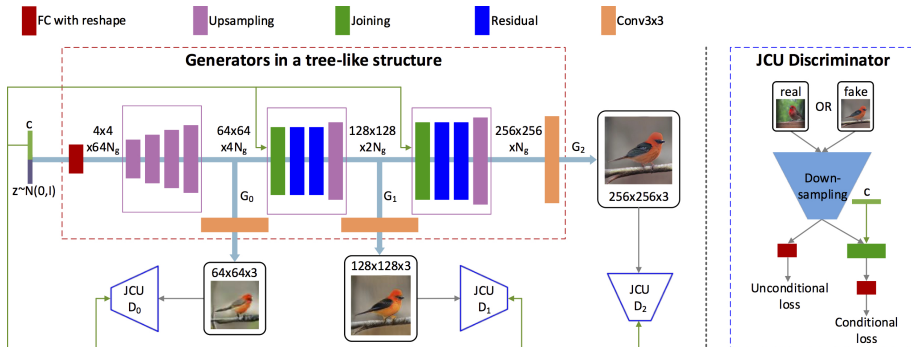


Image credit: Han Zhang *et al.* 2018 (StackGAN++: Realistic Image Synthesis with Stacked Generative Adversarial Networks)

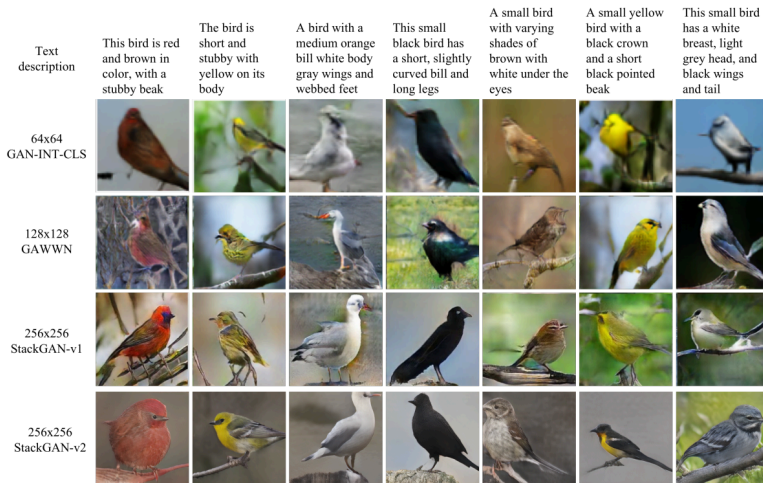


Fig. 3: Example results by our StackGANs, GAWWN [33], and GAN-INT-CLS [35] conditioned on text descriptions from CUB test set.

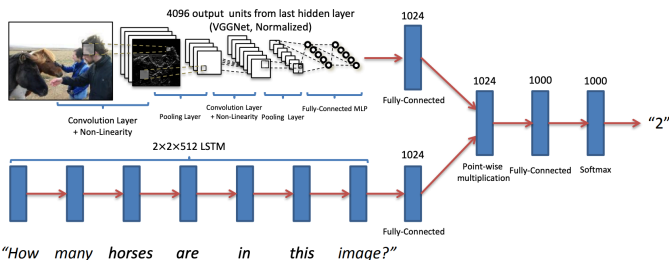


Fig. 8: Our best performing model (deeper LSTM Q + norm I). This model uses a two layer LSTM to encode the questions and the last hidden layer of VGGNet [48] to encode the images. The image features are then ℓ_2 normalized. Both the question and image features are transformed to a common space and fused via element-wise multiplication, which is then passed through a fully connected layer followed by a softmax layer to obtain a distribution over answers.

- Reference architecture for VQA
- Extracting features from both and merge/align them using point-wise multiplication

Image credit: Aishwarya Agrawal *et al.* 2016 (VQA: Visual Question Answering)

Visual Question Answering (VQA)



Is something under the sink broken?	yes	no
	yes	no
	yes	no
What number do you see?	33	5
	33	6
	33	7



Can you park here?	no	no	no
	no	no	yes
What color is the hydrant?	white and orange	red	
	white and orange	white and orange	yellow



What kind of store is this?	bakery	art supplies
	bakery	grocery
	pastry	grocery
Is the display case as full as it could be?	no	no
	no	yes
	no	yes



How many bikes are there?	2	3
	2	4
	2	12
What number is the bus?	48	4
	48	46
	48	number 6



Does this man have children?	yes	yes
	yes	yes
	yes	yes
Is this man crying?	no	no
	no	yes
	no	yes



Has the pizza been baked?	yes	yes	yes
	yes	yes	yes
	yes	yes	yes
What kind of cheese is topped on this pizza?	feta	mozzarella	
	feta	mozzarella	
	ricotta	mozzarella	



How many pickles are on the plate?	1	1
	1	1
	1	1
What is the shape of the plate?	circle	circle
	round	round
	round	round



What does the sign say?	stop	stop
	stop	stop
	stop	stop
What shape is this sign?	octagon	diamond
	octagon	octagon
	octagon	round



How many glasses are on the table?	3	2
	3	2
	3	6
What is the woman reaching for?	door handle	fruit
	glass	glass
	wine	remote



Do you think the boy on the ground has broken legs?	yes	no
	yes	no
	yes	yes
Why is the boy on the right freaking out?	his friend is hurt	ghost
	other boy fell down	lightning
	someone fell	sprayed by hose



Are the kids in the room the grandchildren of the adults?	probably	yes
	yes	yes
	yes	yes
What is on the bookshelf?	nothing	books
	nothing	books
	nothing	books



How many balls are there?	2	1
	2	2
	2	2
What side of the teeter totter is on the ground?	right	left
	right	left
	right side	right side

Fig. 2: Examples of questions (black), (a subset of the) answers given when looking at the image (green), and answers given when not looking at the image (blue) for numerous representative examples of the dataset. See the appendix for more examples.

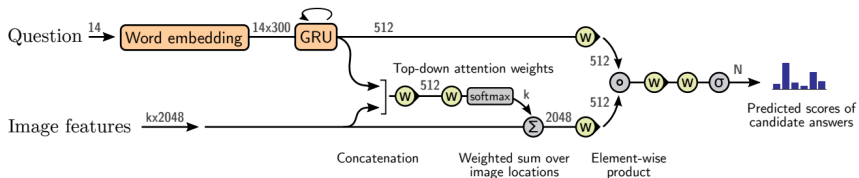


Figure 4. Overview of the proposed VQA model. A deep neural network implements a joint embedding of the question and image features $\{v_1, \dots, v_k\}$. These features can be defined as the spatial output of a CNN, or following our approach, generated using bottom-up attention. Output is generated by a multi-label classifier operating over a fixed set of candidate answers. Gray numbers indicate the dimensions of the vector representations between layers. Yellow elements use learned parameters.

- Still based on the reference model
- With up-down attention

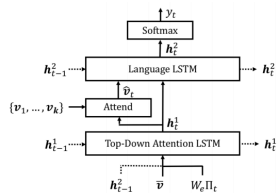
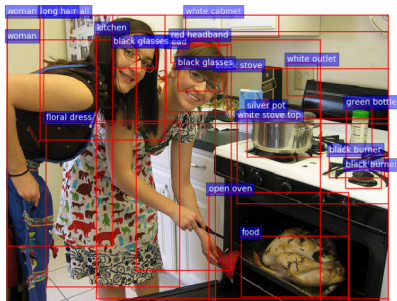
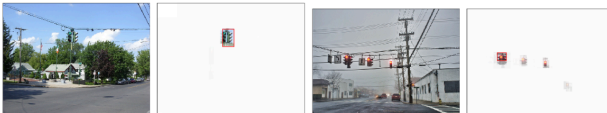


Figure 3. Overview of the proposed captioning model. Two LSTM layers are used to selectively attend to spatial image features $\{v_1, \dots, v_k\}$. These features can be defined as the spatial output of a CNN, or following our approach, generated using bottom-up attention.

- Faster RCNN to extract region vectors
- With up-down attention: using 2 LSTMs to align image-text

Question: What color is illuminated on the traffic light? Answer left: green. Answer right: red.



Question: What is the man holding? Answer left: phone. Answer right: controller.



Question: What color is his tie? Answer left: blue. Answer right: black.

