Disentangling Text

1. Towards Controlled Generation of Text by Hu et al. (https://arxiv.org/abs/1703.00955)
2. Style Transfer from Non-Parallel Text by Cross Alignment by Shen et al. (https://arxiv.org/abs/1705.09655)

Presented by Kelly Zhang

What does “disentanglement” mean for text?

- This is still a pretty open question.
- But generally, it is separating the meaning of a piece of text from it’s style (the way it’s presented)
- Examples:
  - Sentiment
  - Tense
  - Language
  - Word choice (ex: colloquialism vs. formal)

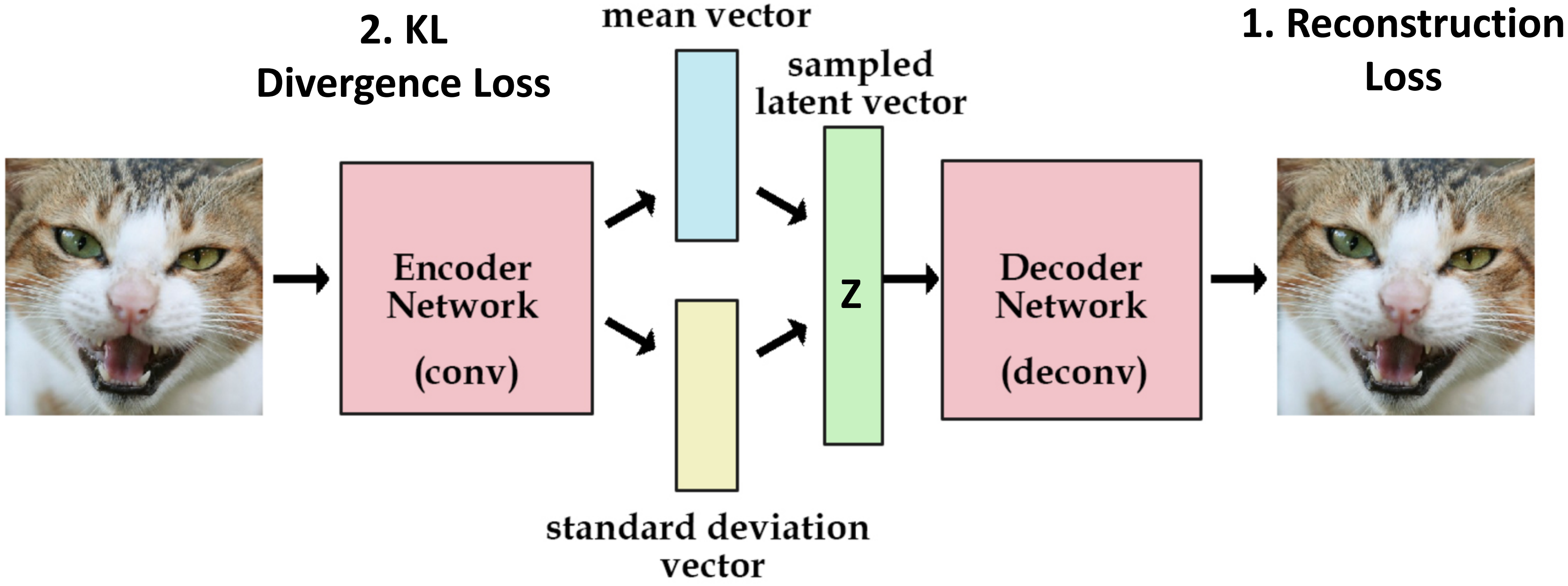
That was a wonderful movie! → That was a horrible movie!

It is hard to imagine a better tribute to this victory of survival than Nolan's spare, stunning, extraordinarily ambitious film. → ?

How is “disentanglement” measured?

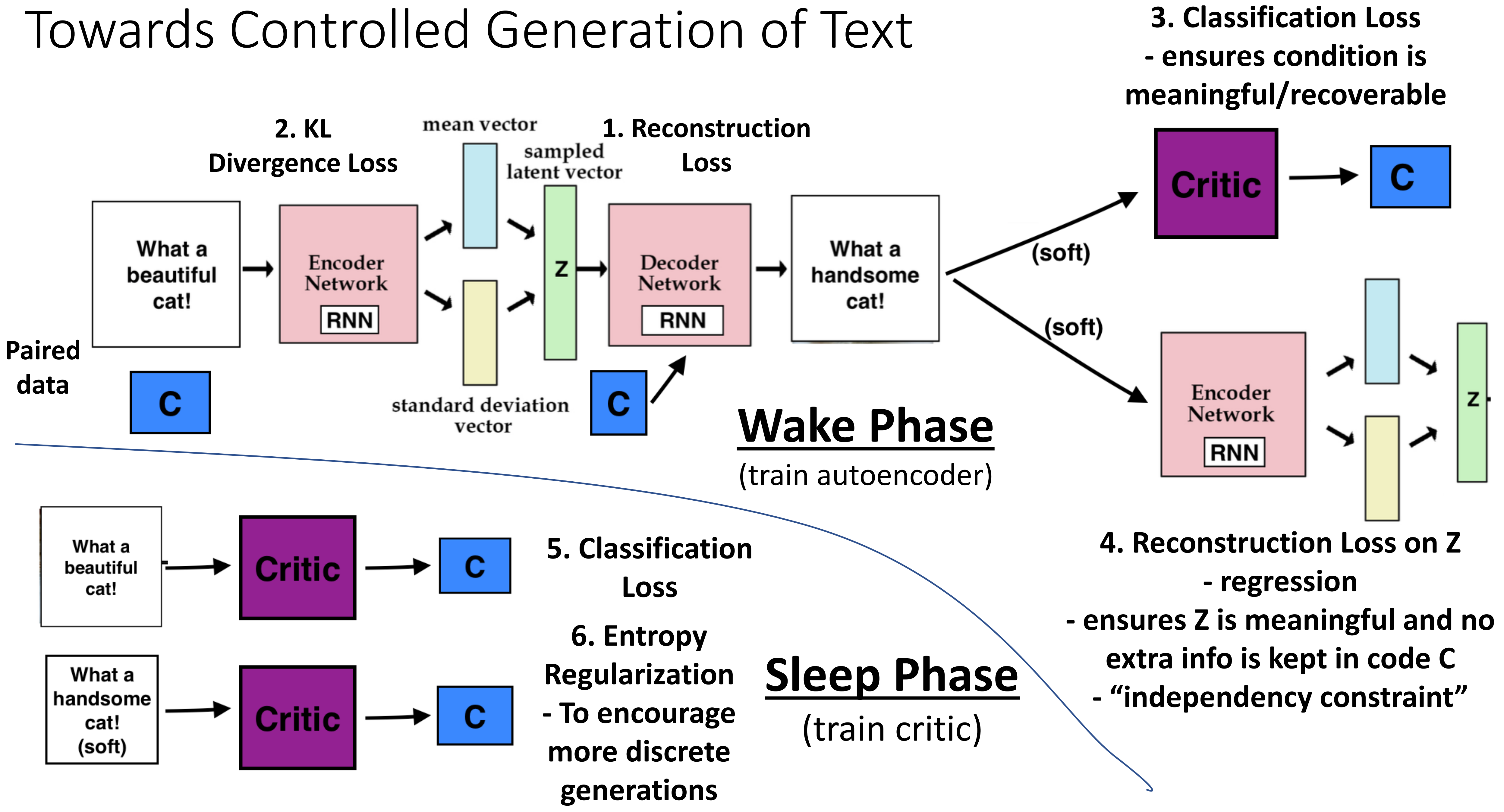
- One relatively easy way to discern that disentanglement is successful is through generation
- One can then score the generations on qualities of interest
- For example on MNIST, one can
  - a) run a pre-trained classifier on generated digits to ensure ”content” of generation is preserved
  - b) evaluate style transfer by visual similarity and nearest neighbors
    (I am not sure of an automatic way to do this)

Variational Autoencoder for Generation



- VAEs encourages the latent state (z) of the autoencoder to be Gaussian (smooth) w/ KL
- This encourages the generations to be coherent when you sample a random z vector for variable generation
- This is one way to learn a mapping between inputs and latent states (will return to this idea)

Towards Controlled Generation of Text



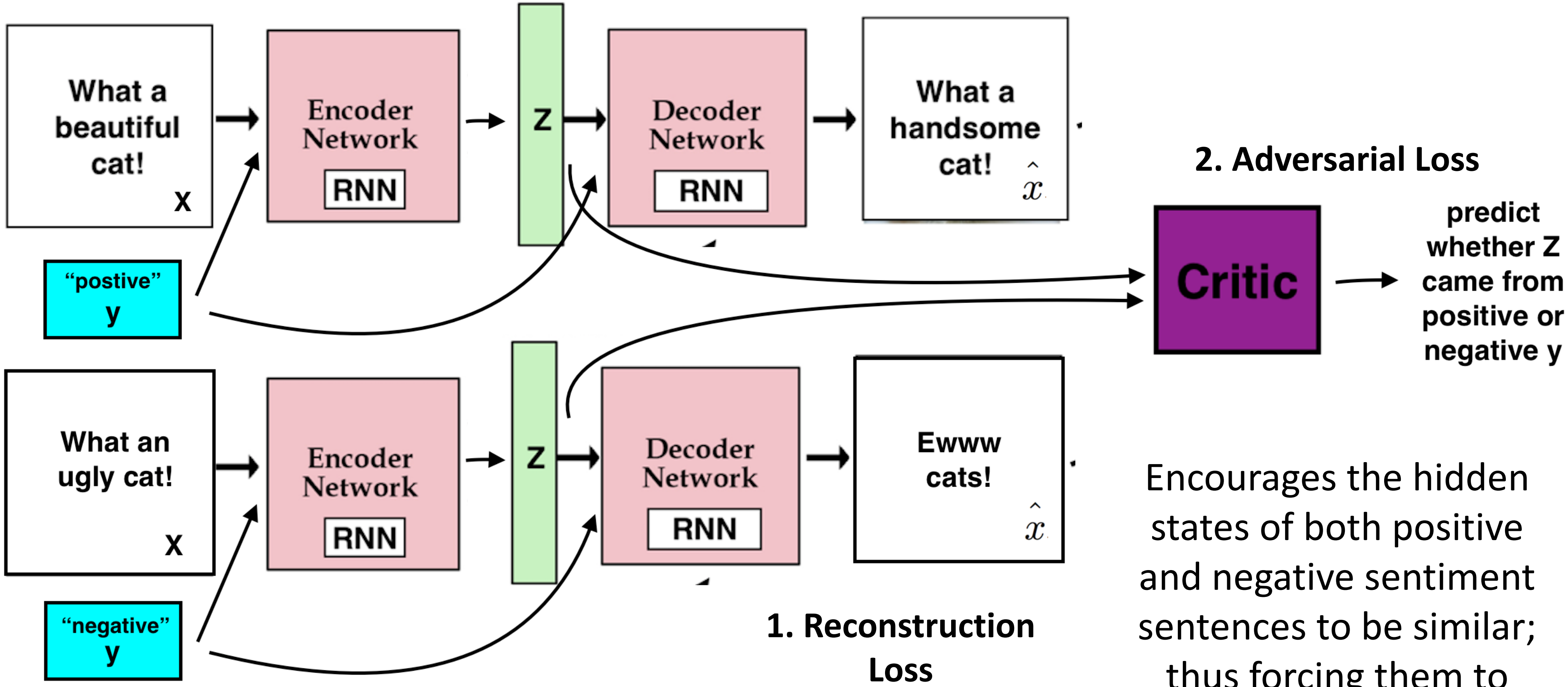
Experiments and Results

- https://arxiv.org/pdf/1703.00955.pdf
- Semisupervised (convnet trained on different data)
  - Std = standard SST
  - S-VAE = augmented w/ gen. from semi-supervised VAE (give label, reconstruct Z)
  - H-reg = aug w/ gen. from critic with entropy regularization on classifier
  - Ours = reconstruct c as well with critic
- Stanford Sentiment Treebank (subset 250 sentences + full)
- IMDB Text Corpus
- Lexicon (word level labels)
- Timebank Tense

Style Transfer from Non-Parallel Text: Aligned Autoencoder

- x = sentence
- y = style
- z = content
- E(x, y) → z
- G(y, z) → x

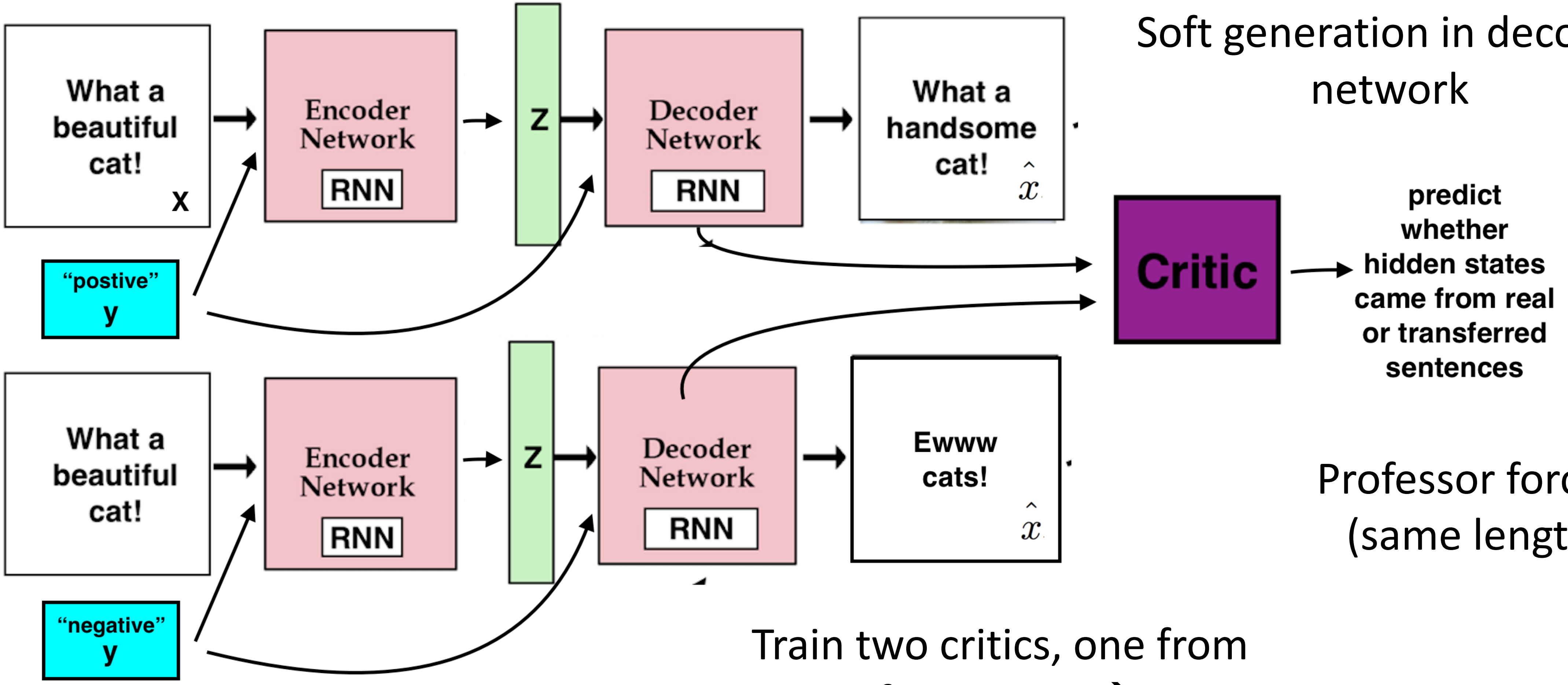
The goal is to separate x into y and z



Encourages the hidden states of both positive and negative sentiment sentences to be similar; thus forcing them to primarily encode content.

Assumes distribution of "content" is similar among both corpora.

Style Transfer from Non-Parallel Text: Cross-Aligned Autoencoder



Soft generation in decoder network

Professor forcing (same length)

Train two critics, one from transferring pos → neg and one neg → pos

Experiments and Results

- https://arxiv.org/pdf/1705.09655.pdf
- Sentiment Modification – yelp
- Word Substitution Decipherment – one to one MT
- Word Order Recovery

Measurements and Comparison

- No quantitative measure of whether the content of the sentence is retained
- Would like to see cross-aligned AE performance on SST and VAE model performance on Yelp data
- "Independency constraint" – reconstructing the code to ensure usage is effective
- VAE model performance on Yelp seems very low
- (Cross-)Aligned Autoencoder models get rid of explicit assumptions on priors (compared to VAEs)
- How finicky are VAEs to optimize? Lower optimization time/resources is a major advantage on its own…
- Further applications
  - Machine translation for non-parallel corpora (finding sentences w/ similar meaning)
  - Entailment conditioned generation (https://arxiv.org/pdf/1606.01404.pdf)

VAEs vs. GANs for Generation

- Taken from a wonderful reddit post: https://www.reddit.com/r/MachineLearning/comments/4r3pjy/variational_autoencoders_vae_vs_generative/
- "The VAE naturally collapses most dimensions in the latent representations, and you generally get very interpretable dimensions out, the the training dynamics are generally a bit weird."
- "GAN is explicitly set up to optimize for generative tasks, though recently it also gained a set of models with a true latent space (BiGAN, ALI + site)."
- "There is some worry that VAE models spread probability mass to places it might not make sense, whereas GAN models may "miss modes" of the true distribution altogether. "

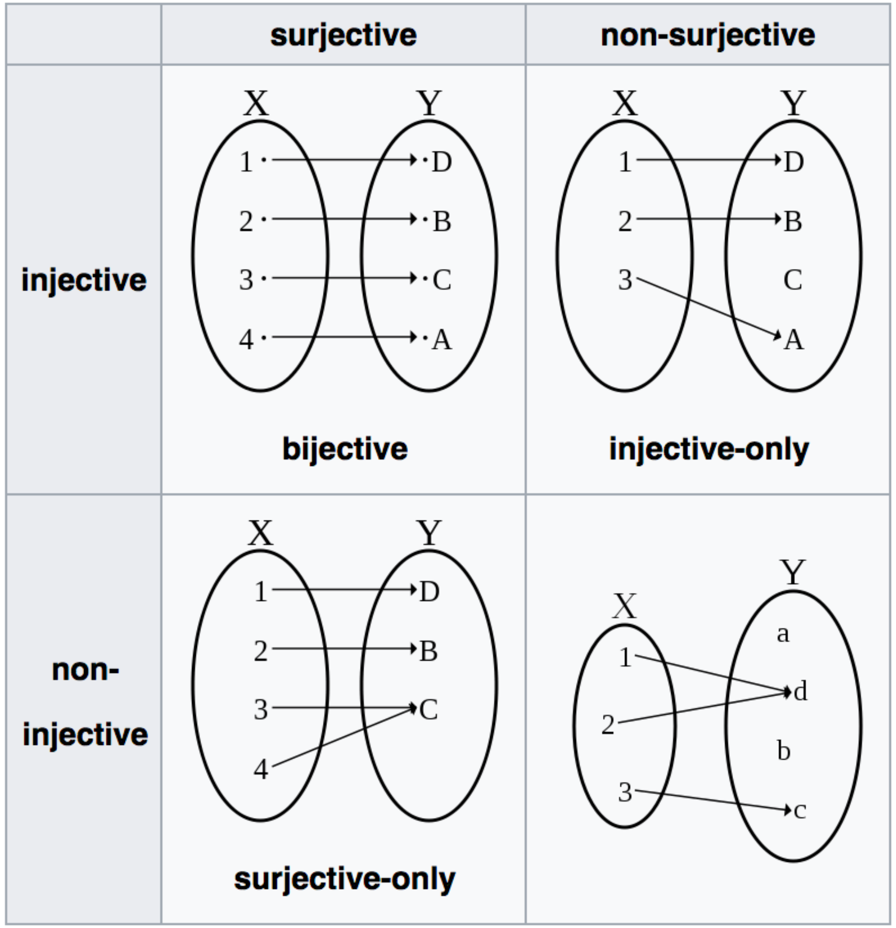
Learning Mappings from Inputs to Latent Space + Generations

(my impressions)

- Autoencoders
  - Simple, but no explicit guarantees on latent space properties.
- Autoregressive Models (Language models, MT, pixel CNNs)
  - Nice nice nice for generation. Interpolations? Disentanglement?
- VAEs
  - Strong prior assumption on latent space. Can work quite well, but training is tricky?
- GANs
  - Transform sampled noise to output. Can miss modes of real data's distribution. Problem of ensuring that the noise is used (not ignored by G).
- ALIGAN/BiGAN
  - https://ishmaelbelghazi.github.io/ALI/
  - Feature matching: https://arxiv.org/abs/1606.03498
- Adversarial Regularization on Z space
  - Fader networks: https://arxiv.org/abs/1706.00409
  - DrNet (Emily Denton): https://sites.google.com/view/drnet-paper/

Ponderings…

- What are good ways to learn mappings to latent spaces? Without restrictive priors? Aligning the hidden states limits lengths of generations
- VAEs seem to try to force a specific mapping
- Mapping noise to outputs → generally requires randomly sampling noise
  - Should a generator be bijective? Or just surjective?
  - Mapping different noise to the same/similar output seems natural as halfway between a car and a dog shouldn’t look real… right?



Non-injective generators map multiple noise to same output

Non-surjective generators miss modes