# Controllable text generation

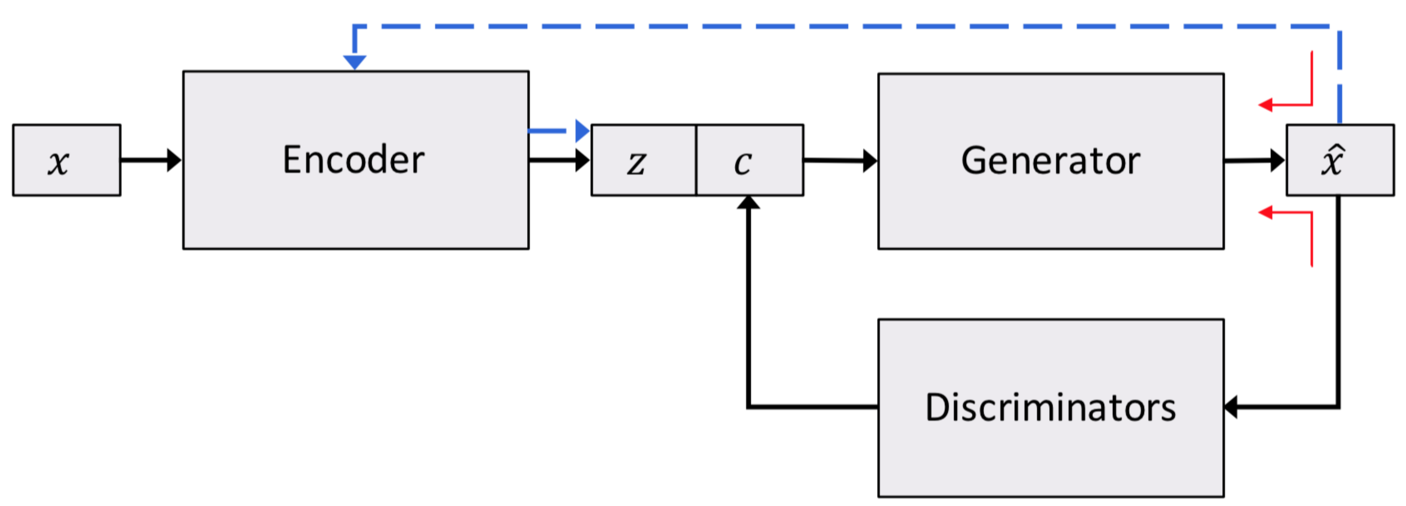
Boris Kovalenko (@kboris), *CS229 Machine Learning*

## Introduction

In last few years generative models advanced greatly in the visual domain. Proposed solutions to problems like image generation and interpretable image representation learning achieve very impressive results and advance quickly. Unlike the tasks of text generation. Text generation is a challenging task. Test samples are discrete and as a result are non-differentiable. This doesnt allow the use of global discriminator, which are common in the visual domain. For example in generative adversarial networks used for image generation. An alternative is a variational autoencoder with element-wise reconstruction loss, but this approach loses the ability to assess generated sentence as a whole. For controllable text generation, one more challenge is learning disentangled latent representation. Varying individual elements of latent representation can cause unpredictable results in the generated sample.

In my project, I work on model described in paper "Toward Controllable Text Generation". The paper proposes a model that addresses issues stated above. The model is variational autoencoder with an extended wake-sleep procedure and structured latent representation, consisting of vector sampled from prior distribution and attributes used for imposing desired semantic properties. Each attribute has dedicated discriminator, which makes learning model, which produces text samples with desired semantic properties possible.

## Model description




Given observation $x$ *Encoder* infers latent vector $z$. Which is a representation of $x$ in latent space:

$$z \sim E[x] = q_E(z|x)$$

Given latent code $(z, c)$ *Decoder* produces plausible text sample, which possesses semantic property, set by $c$:

$$\hat{x} \sim G(z, c) = p_G(\hat{x}|x, c) = \prod_t p(\hat{x}_t|\hat{x}^{<t}, z, c)$$

Text sample generation is iterative. At each step new word is sampled from multinomial distribution, with conditioning on previous word and latent code.

Discriminator returns probability of semantic attribute in given text sample:

$$D(x) = q_D(c|x)$$

## Model training

To account for constrains on latent space and text sample reconstruction loss, following loss term is used:

$$L_{VAE} = -KL(q_E(z|x)||p(z)) + E_{q_E(z|x)q_D(c|x)}[\log p_G(x|z, c)]$$

To provide additional learning signal, which enforces generator to produce text samples with semantic attribute $c$, following loss term is used:

$$L_{Attr,c} = E_{p(z),p(c)}[\log q_D(c|\hat{G}_\tau(z, c))]$$

where $\hat{G}_\tau(z, c)$ - average vector of probabilities of words for generated sentence.

To force disentangled representation we add following loss term:

$$L_{Attr,z} = E_{p(z),p(c)}[\log q_E(z|\hat{G}_\tau(z, c))]$$

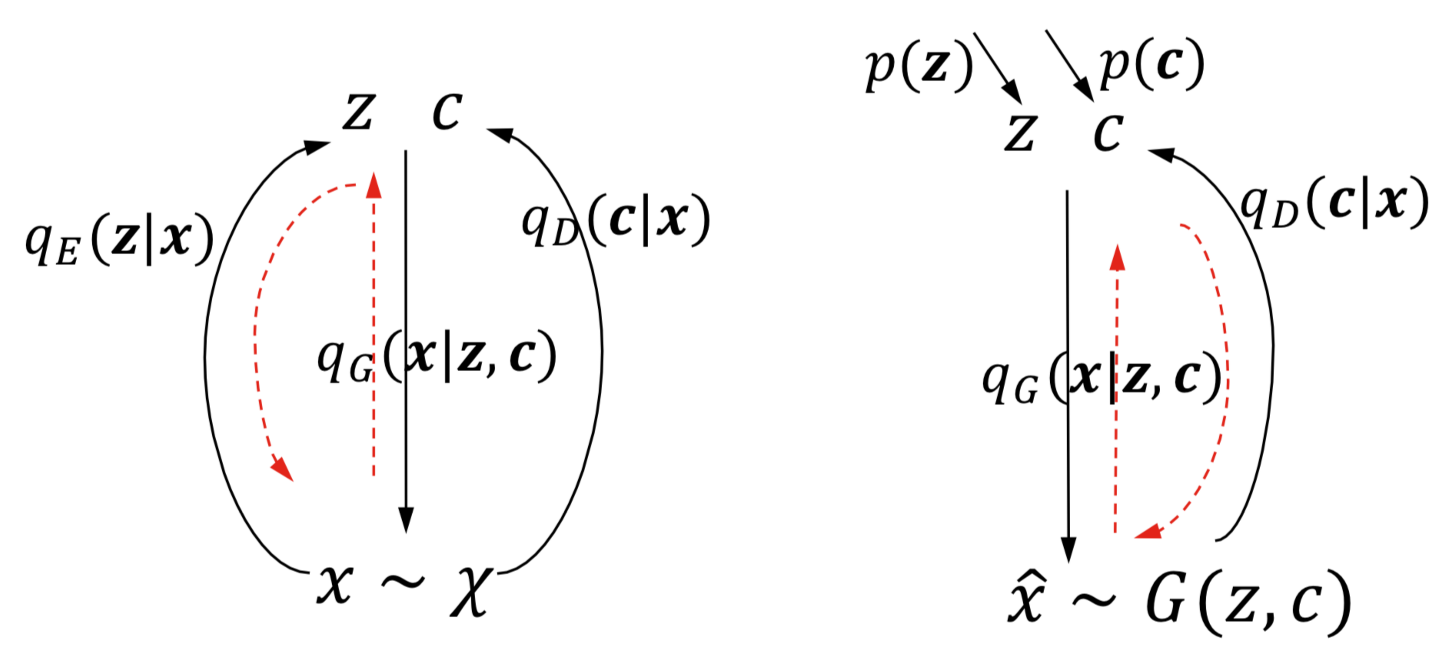
where encoder used to produce latent code $z$ from generated text sample $\hat{G}_\tau(z, c)$

Combining all loss terms, we get generator loss:

$$L_G = L_{VAE} + \lambda_c L_{Attr,c} + \lambda_z L_{Attr,z}$$

Discriminator is trained using labelled samples:

$$L_D = E_{x_L}[\log q_D(c_L|x_L)]$$




Training process depicted on image is resembles advanced wake-sleep mechanism. First step is VAE initialisation. VAE is trained by minimising $L_{VAE}$. p(c) is sampled from uniform distribution.

After VAE warming up we update model elements in a loop until convergence. First update discriminator using $L_D$. Then we update generator and encoder, using losses $L_G$ and $L_{VAE}$

## Dataset

As dataset for training we used IMDB Movie reviews dataset. Dataset consist of 25000 movies reviews from IMDB. Each review is labeled by sentiment positive/negative. Text has been preprocessed. Only reviews shorter than 15 words are used in training.

## Results

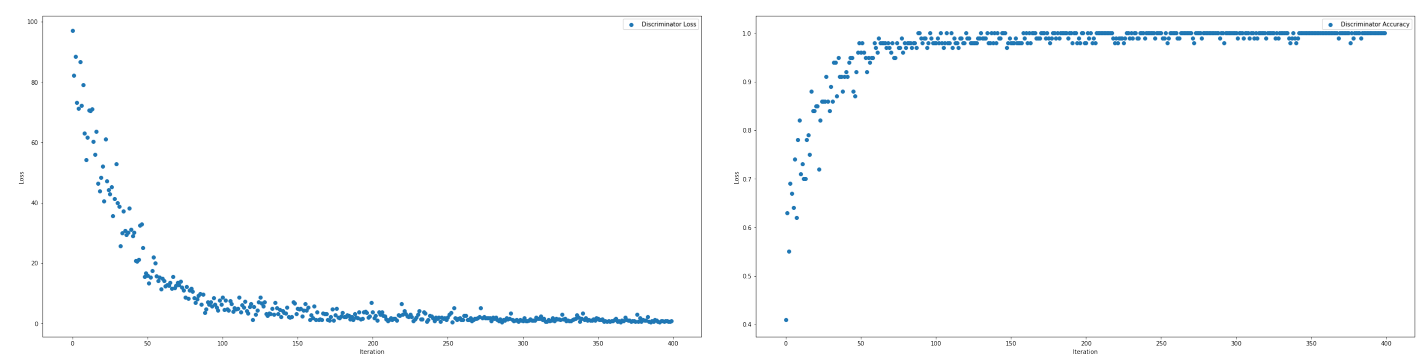


Figure 1: Training Loss and Accuracy for *Discriminator*

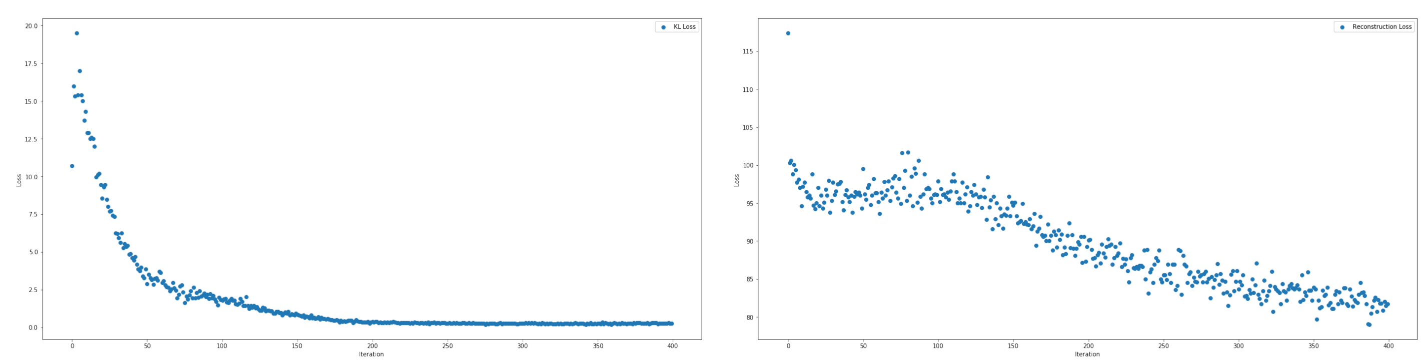


Figure 2: Training KL Loss and Reconstruction Loss for *VAE*

**Generated sequence examples:**

Input sequence - "this is one of the better dance films"

Generated sequence with positive sentiment - "this is one of the best movies i have ever seen in my life"

Generated sequence with negative sentiment - "this is the worst movie i have ever seen in my life and i have"

Input sequence - "his acting was very good"

Generated sequence with positive sentiment - "one of the most respected movies i've seen in tears for the first time"

Generated sequence with negative sentiment - "this is a stupid movie i have ever seen it was the first time i"

Input sequence - "i wish i would never seen this movie"

Generated sequence with positive sentiment - "i am a huge fan of the unk movies that i have seen it"

Generated sequence with negative sentiment - "this is one of the worst movies i have ever seen it was a unk"