Variational Autoencoders for Sequential Data

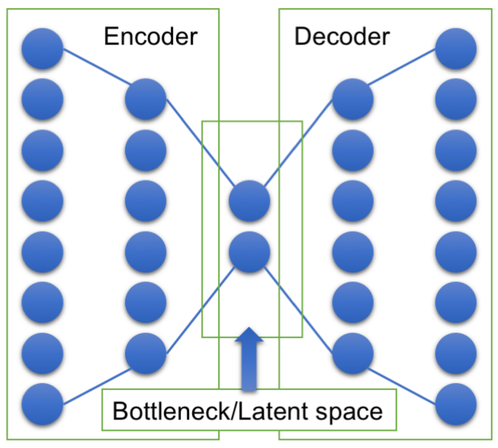
Johan Hyrefeldt, Anton Karlsson

Chalmers University of Technology

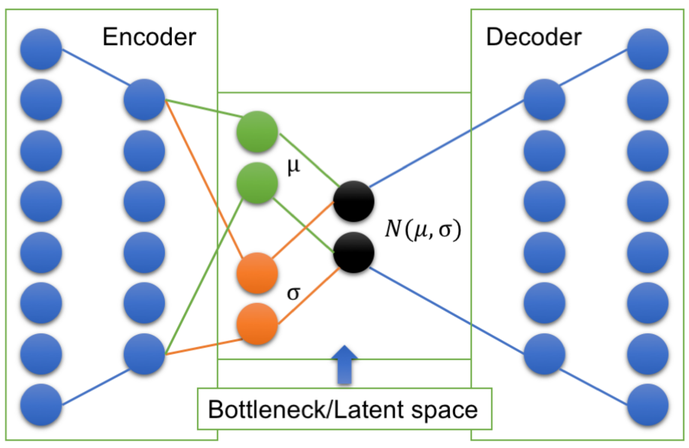
Introduction

An autoencoder is a special setup of a neural network comprising:

- Encoder – encodes the input to a vector in a latent space.
- Bottleneck – the latent space
- Decoder – decodes a vector in the bottleneck back to the input space.



A variational autoencoder instead learns Gaussian distributions (mean and standard deviation) from which it samples the latent vector [2].



The network is trained to reproduce its input, using a loss function

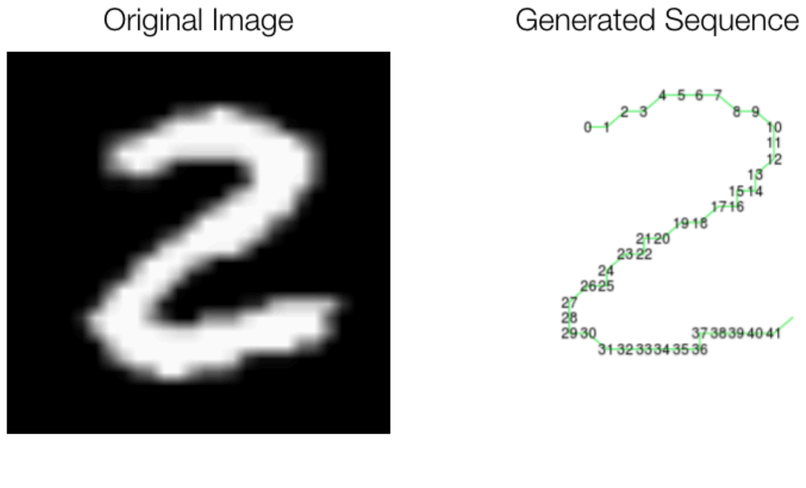
$$L(X) = |X - f(X)|^2 + KL[N(\mu(X), \sigma(X)) \,||\, N(0,1)] \quad (1)$$

where

- X is the input sequence
- $f(X)$ is the output of the complete VAE
- μ is the latent mean vector
- σ is the latent standard deviation vector
- $KL[p(X)||N(0,1)]$ is a measure of how far the distribution $p(X)$ is from $N(0,1)$

Dataset

The dataset in this experiment is based on the famous MNIST dataset of handwritten digits. Instead of each digit being a pixel map this dataset consists of sequences of coordinates as illustrated the figure below [1].

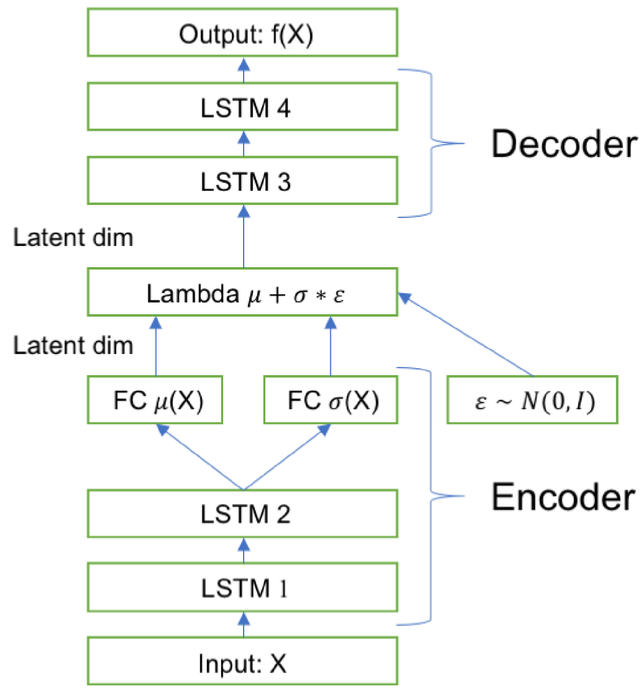


- Each sequence consists of relative coordinates. We transform this to absolute coordinates by accumulating x- and y-values along the sequence.
- To account for varying lengths in the coordinate sequences, interpolated points were randomly inserted or deleted, fixing the input sequence length, T, at 60.

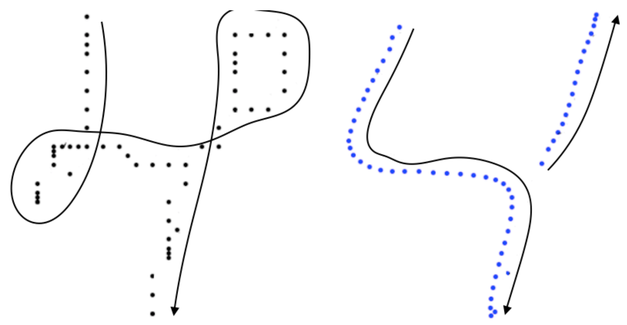
Network architecture

The specific architecture used in this experiment.

- Input dim: $2 \times T$
- Encoder
  - LSTM 1 and 2 output dim: 512
  - FC $\mu(X)$ and FC $\sigma(X)$ output dim: 64
- Bottleneck – Lambda layer
  - Input
    - Vector of 64 means from FC $\mu(X)$
    - Vector of 64 standard deviations from FC $\sigma(X)$
    - Vector ε – 64 samples from $\mathcal{N}(0,1)$
  - Output: 64-dim vector sampled from $\mathcal{N}(\mu(X), \sigma(X))$
- Decoder
  - LSTM 3 output dim: $512 \times T$
  - LSTM 4 output dim: $2 \times T$



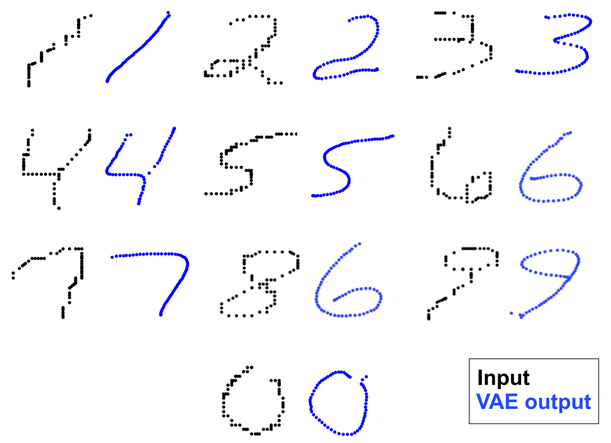
Results



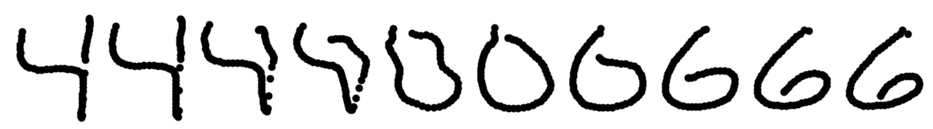
To test the VAE we can decode an encoded test sequence. The VAE learns the ”typical” sequence of a four and may or may not include stylistic features such as loops and multiple strokes. The digit to the left is the input and to the right we have its decoded encoding.

We can also randomly sample latent vectors from the neighbourhood of a typical ”two” to obtain slight variations of the input, which can be useful as data augmentation.

Some more examples of decoded encodings is shown to the right. Some letters are not reproduced correctly, but most of them are still a realistic looking digit, indicating a dense latent space.



We can also move through the latent space by encoding two different images and interpolate a straight line between their latent representations. The latent ”zero-space” happens to lie on the line connecting a four and a six as can be seen in the image below.



References

[1] E. de Jong.
Mnist sequence data.
https://edwin-de-jong.github.io/blog/mnist-sequence-data/.
Accessed: 2018-10-12.

[2] C. Doersch.
Tutorial on variational autoencoders.
arXiv preprint arXiv:1606.05908, 2016.

antkarls@student.chalmers.se, hyrefelt@student.chalmers.se