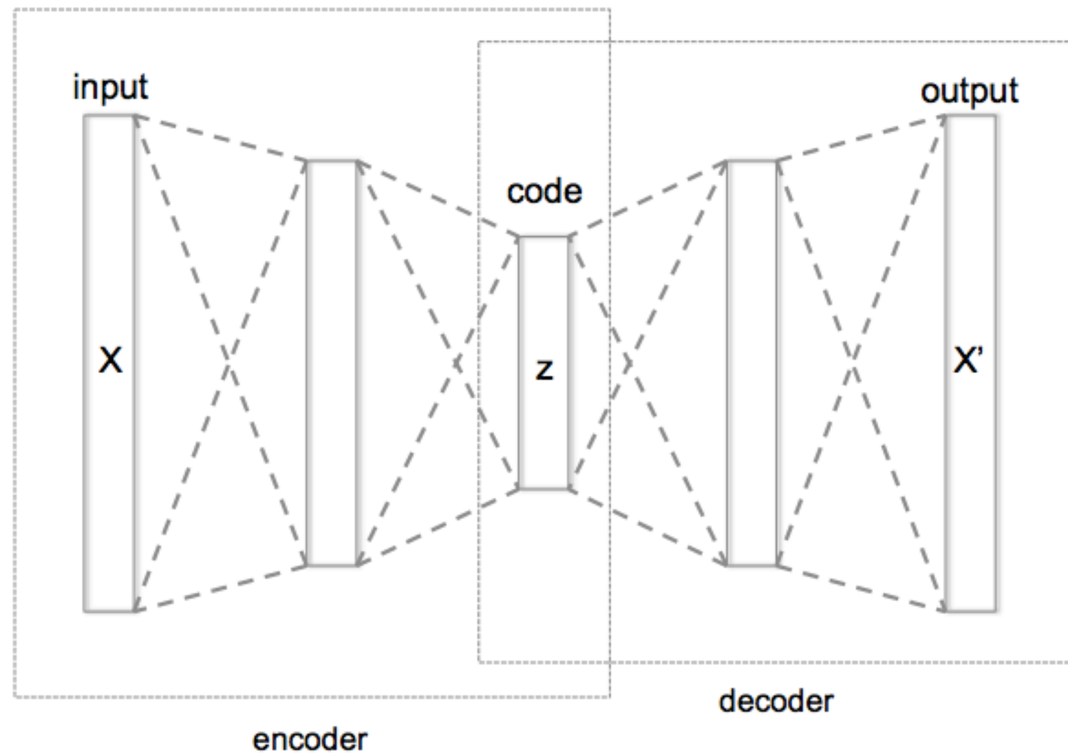
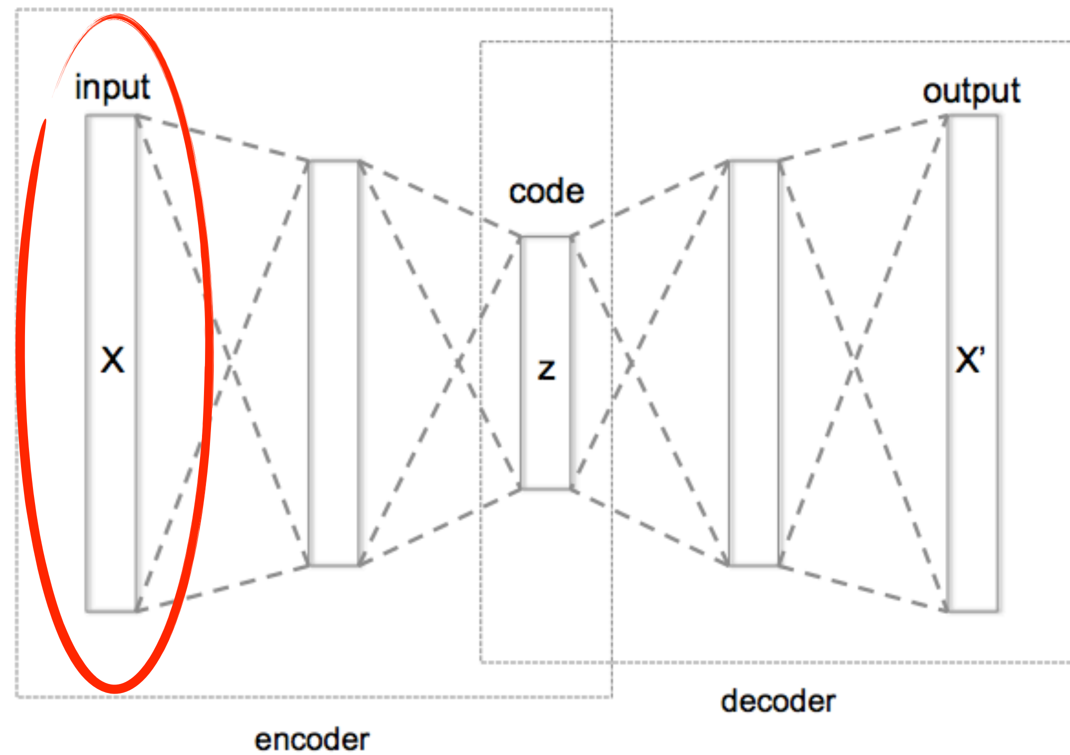
Autoencoder

Autoencoders are - besides LSTMs - the most exiting neural network topologies. They can do amazing things, but let’s first understand how they work.

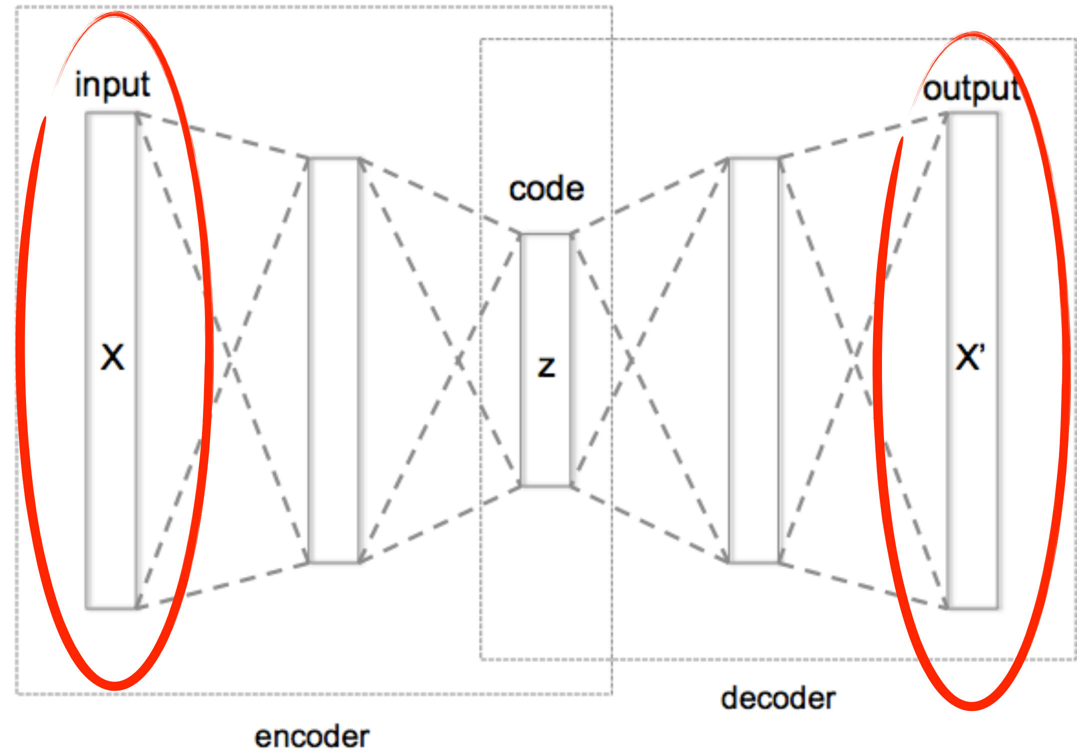


So auto encoders are link other neural networks with a couple of exceptions.



By Chervinskii (Own work) [CC BY-SA 4.0 (https://creativecommons.org/licenses/by-sa/4.0)], via Wikimedia Commons

As in ordinary neural networks we have an input layer where we present examples of our set as vectors X. Usually, neural networks are trained to find a hidden function f which maps from X to Y. But an auto encoder is different.



By Chervinskii (Own work) [CC BY-SA 4.0 (https://creativecommons.org/licenses/by-sa/4.0)], via Wikimedia Commons

It tries to X to X prime. Or in other words. It tries to reconstruct what ever vector X it sees on the left hands side on the right hand side.

IDENTITY FUNCTION

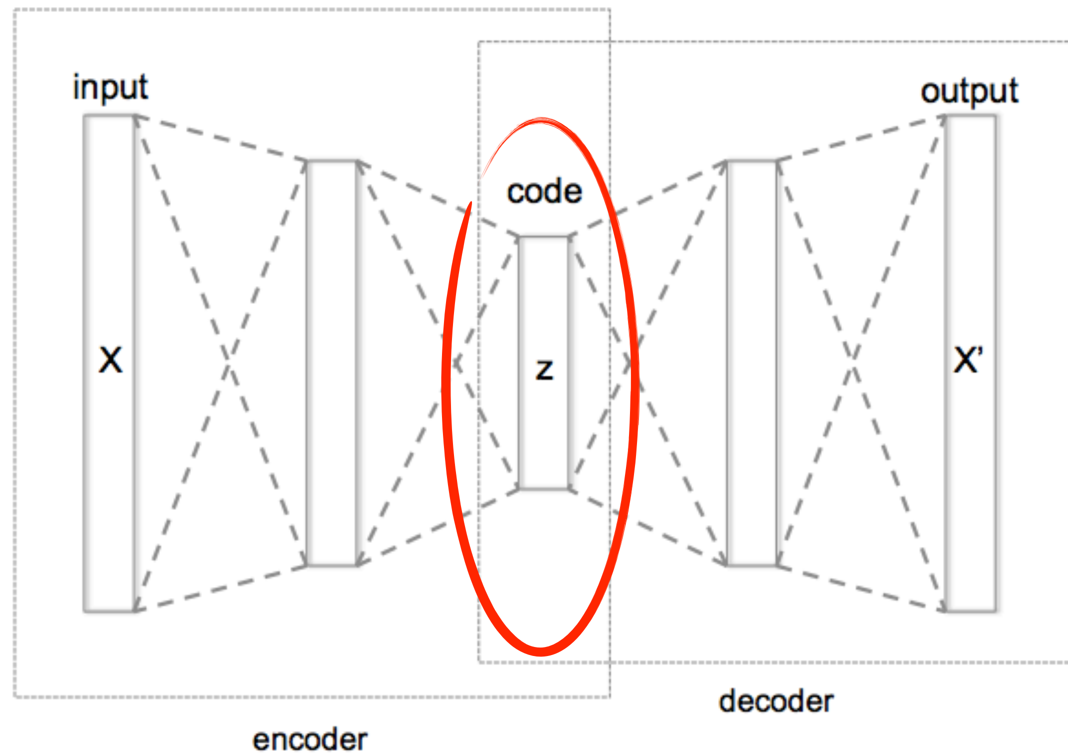
This is also know as the identity function. And as we know that neural networks are

a3_m1_v5_autoencoder_v1

UNIVERSAL FUNCTION APPROXIMATORS

…universal function approximators it is obvious that a neural network can learn the identity function for any data set.

a3_m1_v5_autoencoder_v2



But now there's a catch. As there are no direct connections between X and X prime all data has to pass layer z. Since layer z has far less neurons than X all data needs to flow through this so called neural bottleneck. This leads to a sort of identity function which is resistant to noise and doesn't learn any irrelevant data. We will use auto encoders throughout the course, but let me show you some simple applications of it.

ANOMALY DETECTION

 Romeo Kienzler

Anomaly detection. Since an auto encoder can't learn irrelevant data it will do a good job in learning the whole training data set as good as possible. So whenever data is shown to the auto encoder which it hasn’t seen before it will do a worse job in reconstructing it. This can be used for anomaly detection.

a3_m1_v5_autoencoder_v3

DIMENSION REDUCTION

 Romeo Kienzler

Dimension reduction is one of the most famous unsupervised learning or data mining techniques. Prominent algorithms are PCA which stands for principal component analysis or T-SNE which is t-distributed stochastic neighbour embedding. Whereas PCA is a linear transformation and therefore is less powerful. T-SNE is a non-linear dimension reduction. It has the interesting property that neighbouring distances are preserved. Therefore it is well suited for 2D or 3D plots of high dimensional data. Autoencoders are outperforming T-SNE from a optimisation point of view. Therefore it is very interesting to use them for such tasks.

a3_m1_v5_autoencoder_v4

Neural Network Training

Now we've learned a lot about different neural network types. We've learned that the weight matrix double u is essential for learning and training but we have never learned how to actually come up with a good value set for double u. So let's cover this in the next lecture.