Autoencoder :—

→ Autoencoders — unsupervised learning process that lets you take advantage of your unlabelled data and learn interesting things abt the structure of the data.

- It is used for dimensionality reduction or feature learning.

→ Works by compressing the i/p into a latent space representation and then reconstructing the o/p from this representation.

→ Autoencoder consist of 2 parts : Decoder and Encoder.

  Encoder — takes a i/p data and compress it into a smaller representation
  (eg img)

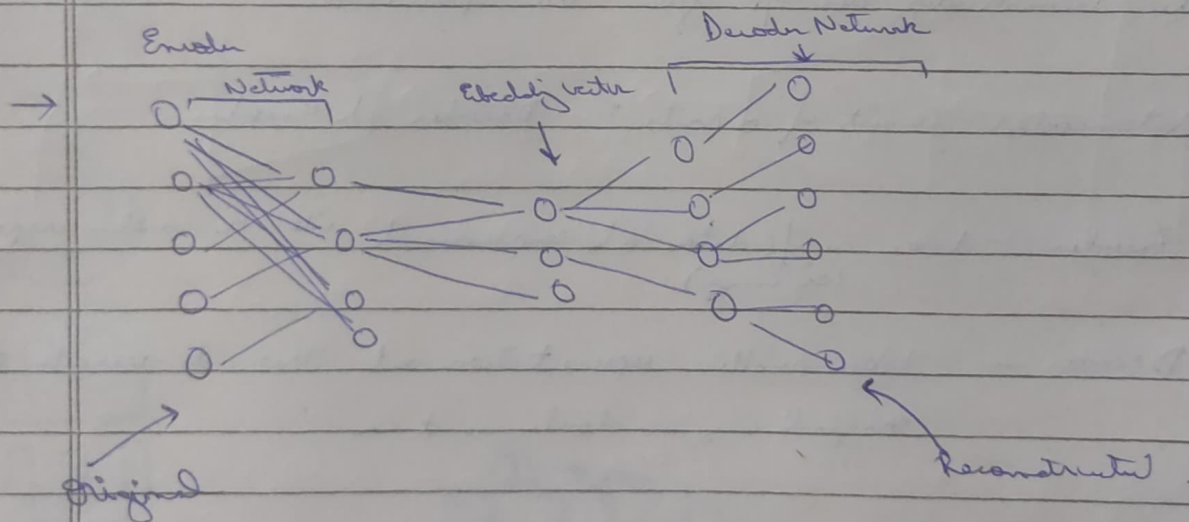 Decoder — takes smaller representation and tries to recreate the original img as closely as it can.

Eg. Imagine you have a magical toy box that can shrink and the bring back your toys.

1) Encoder (Shrinking phase): You put your toys into box. It looks at your toys and tries to figure out the most imp things abt them, like colors, shapes and sizes. Then, it squishes them down into tiny toy version.

2) Decoder (Bring Back phase): Now toys are tiny. It looks at the tiny toys and tries to make them big again. But here's

the cool part: the Toy Brings has to use only the tiny version of the toy to bring back the big ones. So it has to remember all ip stuff it learned during shrinking phase.

→ Useful a variety of reasons like reducing storage al increasing computational efficiency, noise reduction, feature extraction

1) Data Compression
2) Feature learning
3) Anomaly detection
4) Data generation — Once learned ip features, they can use knowledge to create new data that looks similar but are a little different

Encoder Network   Embedding vector   Decoder Network

Original

Reconstructed

Reconstruction error = Reconstructed - Original

Train the network to minimizing the reconstruction error

Encoder al decoder work together to find the most efficient way to condense the ip data into a lower dimension by minimizing the reconstruction error
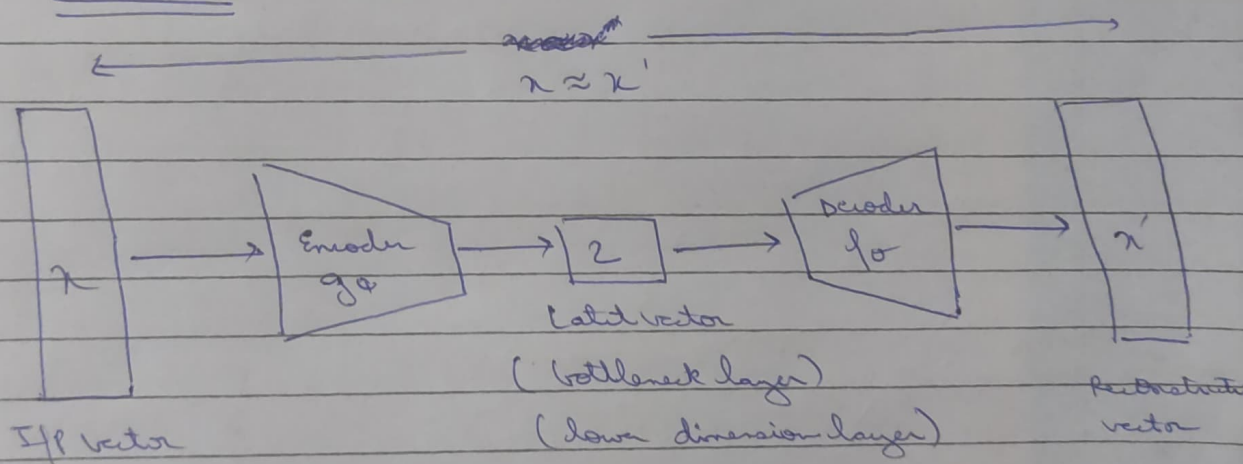
→ For making it as work as feature extractor , after finding the appropriate dimension of embedding matrix for reconstruction we remove the decoder part.

This helps us in reducing the dimensions al extract feature from encoder.

→ For anomaly detection we use full autoencode

Anomaly Score = Reconstruction error.

→ **Architecture :-**

$$x \approx x'$$

Encoder g_ϕ → | 2 | → Decoder f_θ → x'

I/P vector

Latent vector

(bottleneck layer)

(lower dimension layer)

Reconstructed vector

→ Benefit of autoencoder over PCA : • allows model to generalize better
• lesser loss for reconstruction
• works on non-linear data set
PCA - works on linear data set

→ Lossy Compression — when a file can afford to loose some data al/or if storage space needs to be drastically `freed up`.
— the reconstructed img is not exactly the original img but is close in autoencoders