Assignment : 4

- **Aim**

Use Autoencoder to implement anomaly detection. Build the model by using the following:

a. Import required libraries
b. Upload / access the dataset
c. The encoder converts it into a latent represento.
d. Decoder networks converts it back to the origional inpute.
e. compile the models with optimizer, loss and Evaluation matrics.

- **Course objectives**

1. To apply the algorithms to a real-world problem, optimise the models learned and report on the expected accuracy that can be achieved by applying the models.

- **Course outcomes**

CO3 : Apply Deep learning techniques like CNN and RNN Auto encoders to solve real-world problems

- **softwares and hardwares requirement.**

Srno	Hardwores and softwares	Version / specification
1.	Jupyter notebook	V.7.13.008
2.	Computer / PC	7 Version, 64 bits. 8GB RAM.

Theory

steps and algorithm of Autoencoder to anomaly detection

1. Dataset collect and libraries import libraries
2. Import following libraries from tensorflow keras models, layers, optimizer, dataset and set to respective values
3. Grab to ECG.csv required dataset
4. ~~Use~~ Find shape of dataset
5. Use train_test_split from sklearn to build model (e.g. train_test_split).
6. Scale the data using MinMaxScalar.
7. ~~select~~ /create Autoencoder subclass by extending model class from keras
8. select parameters as i. encoder, layer ii. Decoder iii. Activation function : Relu . iv. model : sequential
9. configure model with following parameters : epochs = 20, batch size = 512 and compile with MSL:

10. Plot loss, val-loss, Epochs and msle loss
1
11. find threshold for anomaly and do prediction.
12. Get accuracy score.

Autoencoders

- Autoencoders are a class of neural network architectures used for unsupervised learning.
- They are particularly useful for encoding real-world data into a lower-dimensional Representation. also known as the 'latent space' or 'encoding space'.
- Autoencoders consists of two main parts : encoder, which maps the input data to the latent space, and the decoder, which reconstruct the data from the encoded representation
- The objective is to minimize the difference betn the origional input and the reconstructed output, encouraging the model to learn a compact representation of the data.

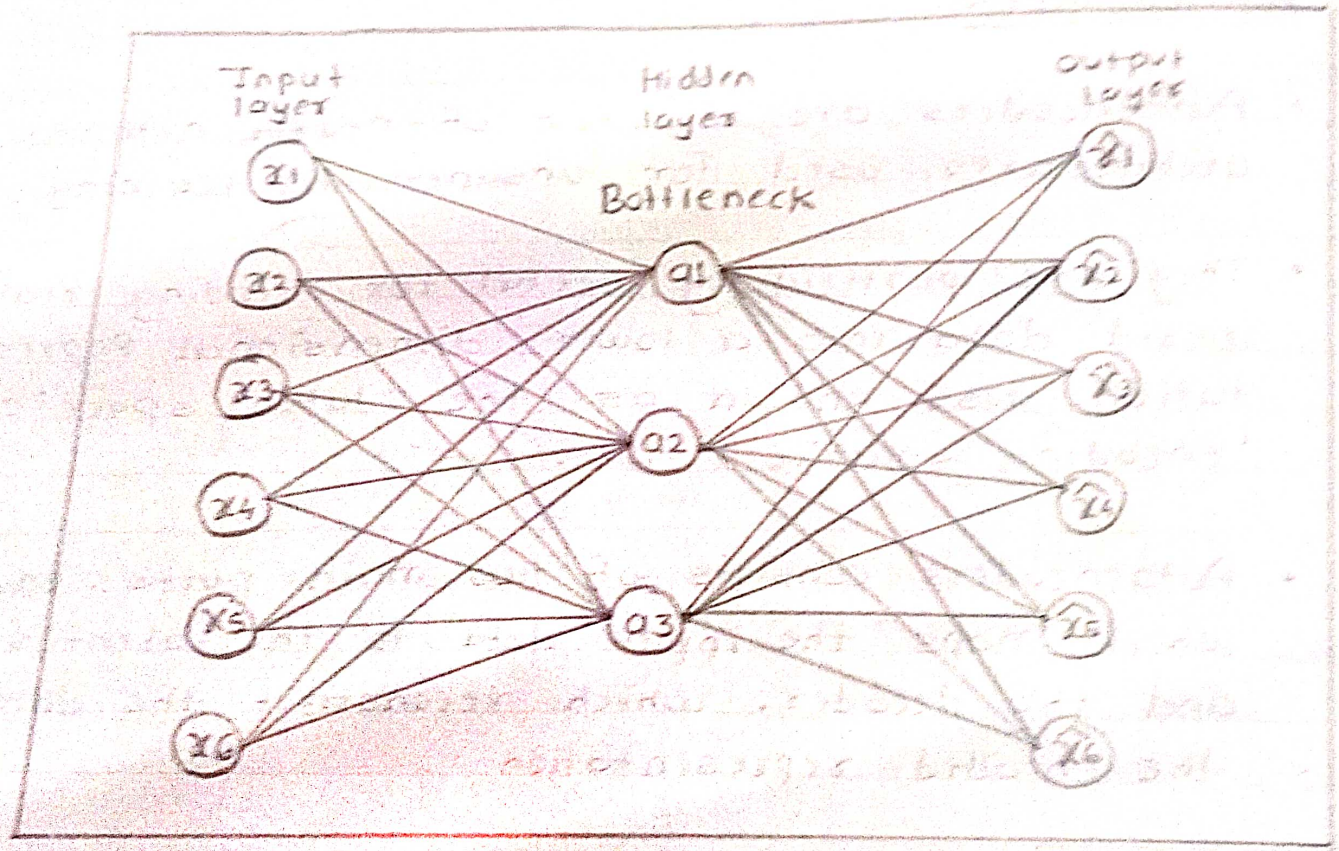

fig (Autoencoder)

Algorithms for anomaly detection

i. decision trees
ii. support vector machines (SVM)
iii. K-nearest neighbour (KNN)
iv. Isolation forest algorithm

Architecture of Auto-encoder in Deep learning

- The general architecture of an auto-encoders includes an encoder, decoder and bottleneck layer

1. Encoder:

- Input layer takes raw input data
- Hidden layer progressively reduce the dimentional of input, capturing important feature and patterns
- The bottleneck layer (latent space)

2. Decoder

- The bottleneck layer takes the encoded representati and encodes representation and erpands it back to the dimentionality of origional input
- The hidden layers progressively increase the dime ntionality and aim to reconstruct the origional input

3. The loss function used during training is typically a reconstruction loss, measuring the difference bet input and reconstructed output.

Applications of encoders

1. Data compression

Compress high-dimentional data into lower dimension data into lower dimensions (encoding) and reconstruct it back (decoding).
e.g. (e image or audio compression)

2. Feature extraction

Encoders are used in deep learning to extract important features from raw data, which are used in tasks like image recognition and NLP.

3. 3D object Reconstruction

Encoders are applied 3D modelling, where 2D img are encoded into 3D shapes. for use in augmented Reality, virtual reality, and game development.

4. Text Representation in NLP

Encoder such as in transformer or word2vec, encodes text data into vector representation for tasks like sentiment analysis, question answering or machine translation

Conclusion

In this practical, the autoencoder reconstructs normal data and detects anomalies by measuring reconstruction errors, which are higher for anomalies. fine-tuning the model and threshold improves detection accuracy