Roadmap: Deep Learning Projects and Resources

Date : 07, June, 2024

1. **Introduction to Deep Learning**
   - **Project Idea**: Implement a basic neural network for binary classification.
   - **Resources**: GeeksforGeeks tutorials on basic neural networks.

2. **Difference Between Artificial Intelligence vs Machine Learning vs Deep Learning** – **Project Idea**: None directly, focus on understanding concepts.

3. **Basic Neural Network**
   - **Project Idea**: Implement a multi-layer perceptron (MLP) for MNIST digit classification.
   - **Resources**: TensorFlow or PyTorch documentation, Kaggle datasets.

4. **Difference between ANN and BNN**
   - **Project Idea**: None directly, focus on theoretical understanding.

5. **Single Layer Perceptron in TensorFlow**
   - **Project Idea**: Implement a single layer perceptron for binary classification.
   - **Resources**: TensorFlow documentation, Kaggle datasets.

6. **Multi-Layer Perceptron Learning in TensorFlow**
   - **Project Idea**: Extend to a multi-layer perceptron for classification or regression.
   - **Resources**: TensorFlow documentation, Kaggle datasets.

7. **Deep Neural net with forward and back propagation from scratch - Python**
   - **Project Idea**: Implement a deep neural network with backpropagation for classification.
   - **Resources**: Python programming resources, NumPy for matrix operations.

8. **Understanding Multi-Layer Feed Forward Networks**
   - **Project Idea**: Implement a feedforward neural network for image classification.
   - **Resources**: TensorFlow or PyTorch documentation, MNIST or CIFAR-10 datasets.

9. **List of Deep Learning Layers**
   - **Project Idea**: Implement a custom deep learning architecture (e.g., VGG, ResNet) for image classification.
   - **Resources**: TensorFlow or PyTorch documentation, ImageNet dataset.

–

10. **Activation Functions**
    - **Project Idea**: Compare different activation functions (ReLU, Sigmoid, Tanh) on a classification task.
    - **Resources**: TensorFlow or PyTorch documentation, Kaggle datasets.

11. **Artificial Neural Network**
    - **Project Idea**: Implement an ANN for a real-world dataset (e.g., housing price prediction).
    - **Resources**: Scikit-learn for preprocessing, TensorFlow or PyTorch for modeling.

12. **Gradient Descent Optimization in Tensorflow**
    - **Project Idea**: Implement gradient descent variants (SGD, Adam) for optimizing a neural network.
    - **Resources**: TensorFlow documentation, Kaggle datasets.

13. **Choose Optimal Number of Epochs to Train a Neural Network in Keras**
    - **Project Idea**: Experiment with early stopping techniques on a neural network project.
    - **Resources**: Keras documentation, TensorFlow tutorials.

14. **Classification**
    - **Project Idea**: Implement a CNN for image classification (e.g., CIFAR-10 dataset). – **Resources**: TensorFlow or PyTorch documentation, Kaggle datasets.

15. **Python | Classify Handwritten Digits with Tensorflow**
    - **Project Idea**: Implement a CNN for handwritten digit recognition (MNIST dataset).
    - **Resources**: TensorFlow documentation, MNIST dataset.

16. **Train a Deep Learning Model With Pytorch**
    - **Project Idea**: Implement a deep learning model using PyTorch for a specific task (e.g., image classification).
    - **Resources**: PyTorch documentation, Kaggle datasets.

17. **Regression**
    - **Project Idea**: Implement a linear regression model using PyTorch or TensorFlow. – **Resources**: TensorFlow or PyTorch documentation, Kaggle datasets.

–

18. **Linear Regression using PyTorch**
    - **Project Idea**: Extend to multiple linear regression or polynomial regression. – **Resources**: PyTorch documentation, NumPy for data preprocessing.

19. **Linear Regression Using Tensorflow**
    - **Project Idea**: Implement linear regression for a real-world dataset (e.g., housing prices).
    - **Resources**: TensorFlow documentation, Scikit-learn for data handling.

20. **Hyperparameter tuning**
    - **Project Idea**: Perform hyperparameter tuning on a deep learning model (e.g., CNN).
    - **Resources**: Grid search or random search techniques, TensorFlow or PyTorch documentation.

21. **Introduction to Convolution Neural Network**
    - **Project Idea**: Implement a simple CNN architecture (LeNet) for image classification.
    - **Resources**: TensorFlow or PyTorch documentation, LeNet architecture.

22. **Digital Image Processing Basics**
    - **Project Idea**: Explore basic image processing techniques for data augmentation in deep learning projects.
    - **Resources**: OpenCV documentation, TensorFlow or PyTorch for integration.

23. **Difference between Image Processing and Computer Vision**
    - **Project Idea**: Implement computer vision tasks like object detection or image segmentation using deep learning.
    - **Resources**: TensorFlow or PyTorch documentation, Pascal VOC or COCO datasets.

24. **CNN | Introduction to Pooling Layer**
    - **Project Idea**: Understand the impact of pooling layers in a CNN model.
    - **Resources**: TensorFlow or PyTorch documentation, CNN tutorials.

25. **CIFAR-10 Image Classification in TensorFlow**
    - **Project Idea**: Implement a CNN model for the CIFAR-10 dataset.
    - **Resources**: TensorFlow documentation, CIFAR-10 dataset.

–

26. **Implementation of a CNN based Image Classifier using PyTorch**
    - **Project Idea**: Implement an advanced CNN architecture (e.g., ResNet) for image classification.
    - **Resources**: PyTorch documentation, ImageNet dataset.

27. **Convolutional Neural Network (CNN) Architectures**
    - **Project Idea**: Implement state-of-the-art CNN architectures like VGG, ResNet, or Inception.
    - **Resources**: Research papers, TensorFlow or PyTorch documentation.

28. **Object Detection vs Object Recognition vs Image Segmentation**
    - **Project Idea**: Implement an object detection model using YOLO or SSD.
    - **Resources**: YOLO or SSD documentation, Pascal VOC or COCO datasets.

29. **YOLO v2 - Object Detection**
    - **Project Idea**: Implement YOLO v2 for object detection tasks.
    - **Resources**: YOLO v2 documentation, VOC or COCO datasets.

30. **Recurrent Neural Network**
    - **Project Idea**: Implement an RNN for time series prediction or natural language processing tasks.
    - **Resources**: TensorFlow or PyTorch documentation, LSTM or GRU architectures.

31. **Natural Language Processing (NLP) Tutorial**
    - **Project Idea**: Implement a sentiment analysis model using RNNs or transformers.
    - **Resources**: Transformers documentation, IMDB or Twitter sentiment datasets.

32. **Introduction to NLTK: Tokenization, Stemming, Lemmatization, POS Tagging**
    - **Project Idea**: Preprocess text data for NLP tasks using NLTK.
    - **Resources**: NLTK documentation, text datasets.

33. **Word Embeddings in NLP**
    - **Project Idea**: Implement word embeddings (e.g., Word2Vec, GloVe) for an NLP task.
    - **Resources**: Word2Vec or GloVe documentation, text datasets.

–

34. **Sentiment Analysis with an Recurrent Neural Networks (RNN)**
    – **Project Idea**: Implement sentiment analysis using an RNN or LSTM.
    – **Resources**: TensorFlow or PyTorch documentation, sentiment datasets.

35. **What is LSTM - Long Short Term Memory?**
    – **Project Idea**: Implement an LSTM model for time series prediction or text generation.
    – **Resources**: LSTM documentation, TensorFlow or PyTorch tutorials.

36. **Long Short Term Memory Networks Explanation**
    – **Project Idea**: Extend LSTM to a more complex NLP task like text generation
    – **Resources**: TensorFlow or PyTorch documentation, text generation techniques.

37. **LSTM - Derivation of Back propagation through time**
    – **Project Idea**: Implement backpropagation through time (BPTT) for an LSTM model.
    – **Resources**: LSTM documentation, TensorFlow or PyTorch tutorials.

38. **Text Generation using Recurrent Long Short Term Memory Network**
    – **Project Idea**: Implement an LSTM-based text generation model.
    – **Resources**: TensorFlow or PyTorch documentation, text generation datasets.

39. **Gated Recurrent Unit Networks**
    – **Project Idea**: Implement a GRU-based model for an NLP task like machine translation.
    – **Resources**: TensorFlow or PyTorch documentation, GRU architectures.

40. **ML | Text Generation using Gated Recurrent Unit Networks**
    – **Project Idea**: Extend GRU to text generation tasks with character-level or wordlevel models.
    – **Resources**: TensorFlow or PyTorch documentation, text generation techniques.

41. **Generative Learning**
    – **Project Idea**: Implement an autoencoder for dimensionality reduction or anomaly detection.
    – **Resources**: TensorFlow or PyTorch documentation, anomaly detection datasets.

–

42. **Autoencoders -Machine Learning**
    - **Project Idea**: Implement a variational autoencoder (VAE) for generative modeling.
    - **Resources**: TensorFlow or PyTorch documentation, generative modeling techniques.

43. **How Autoencoders works ?**
    - **Project Idea**: Implement a denoising autoencoder for image reconstruction tasks.
    - **Resources**: TensorFlow or PyTorch documentation, image denoising datasets.

44. **Variational AutoEncoders**
    - **Project Idea**: Implement a VAE for unsupervised learning tasks like clustering.
    - **Resources**: TensorFlow or PyTorch documentation, clustering datasets.

45. **Contractive Autoencoder (CAE)**
    - **Project Idea**: Implement a CAE for robust feature extraction from noisy data.
    - **Resources**: TensorFlow or PyTorch documentation, noisy dataset.

46. **ML | AutoEncoder with TensorFlow 2.0**
    - **Project Idea**: Implement an autoencoder using TensorFlow 2.0 for a specific application.
    - **Resources**: TensorFlow 2.0 documentation, TensorFlow tutorials.

47. **Implementing an Autoencoder in PyTorch**
    - **Project Idea**: Implement an autoencoder using PyTorch and apply it to an image dataset.
    - **Resources**: PyTorch documentation, image dataset.

48. **Generative adversarial networks**

    Project Idea: Implement a basic GAN for generating images like faces or handwritten digits.
    - **Resources**: TensorFlow or PyTorch documentation, GAN tutorials.

49. **Basics of Generative Adversarial Networks (GANs)**
    - **Project Idea**: Extend GAN to conditional GANs for controlled image generation.
    - **Resources**: TensorFlow or PyTorch documentation, conditional GAN tutorials.

–

50. **Generative Adversarial Network (GAN)**
    - **Project Idea**: Implement a DCGAN for generating high-resolution images.
    - **Resources**: TensorFlow or PyTorch documentation, DCGAN

51. **Cycle Generative Adversarial Network (CycleGAN)**
    - **Project Idea**: Implement CycleGAN for image-to-image translation tasks (e.g., day to night image conversion).
    - **Resources**: TensorFlow or PyTorch documentation, CycleGAN tutorials.

52. **StyleGAN - Style Generative Adversarial Networks**
    - **Project Idea**: Implement StyleGAN for high-quality image synthesis.
    - **Resources**: TensorFlow or PyTorch documentation, StyleGAN tutorials.

53. **Reinforcement Learning**
    - **Project Idea**: Implement Q-learning for solving simple reinforcement learning problems (e.g., maze navigation).
    - **Resources**: OpenAI Gym environment, reinforcement learning tutorials.

54. **Understanding Reinforcement Learning in-depth**
    - **Project Idea**: Extend to more complex reinforcement learning algorithms like Deep Q-learning.
    - **Resources**: Reinforcement learning research papers, advanced RL tutorials.

55. **Introduction to Thompson Sampling | Reinforcement Learning**
    - **Project Idea**: Implement Thompson Sampling for multi-armed bandit problems.
    - **Resources**: Thompson Sampling tutorials, RL libraries.

56. **Markov Decision Process**
    - **Project Idea**: Implement MDP for decision-making in dynamic environments.
    - **Resources**: Markov Decision Process tutorials, RL libraries.

57. **Bellman Equation**
    - **Project Idea**: Implement value iteration or policy iteration using Bellman equations.
    - **Resources**: RL textbooks, reinforcement learning tutorials.

58. **Meta-Learning in Machine Learning**
    - **Project Idea**: Implement few-shot learning techniques using meta-learning approaches.
    - **Resources**: Meta-learning research papers, meta-learning frameworks.

–

59. **Q-Learning in Python**
    - **Project Idea**: Implement Q-learning algorithm for grid-world navigation.
    - **Resources**: OpenAI Gym environment, reinforcement learning tutorials.

60. **ML | Reinforcement Learning Algorithm : Python Implementation using Q-learning**
    - **Project Idea**: Extend Q-learning to a more complex environment with rewards and penalties.
    - **Resources**: Reinforcement learning libraries, RL tutorials.

61. **Deep Q Learning**
    - **Project Idea**: Implement Deep Q-learning for Atari game playing.
    - **Resources**: OpenAI Gym, Deep Q-learning tutorials.

62. **Deep Q-Learning**
    - **Project Idea**: Extend Deep Q-learning to handle continuous action spaces.
    - **Resources**: Deep Q-learning research papers, RL frameworks.

63. **Implementing Deep Q-Learning using Tensorflow**
    - **Project Idea**: Implement DQN using TensorFlow for a specific game or task.
    - **Resources**: TensorFlow documentation, DQN tutorials.

64. **AI Driven Snake Game using Deep Q Learning**
    - **Project Idea**: Implement Deep Q-learning to play the classic Snake game.
    - **Resources**: OpenAI Gym environment, Snake game implementation guides.