1.How I'd Learn AI in 2025 (if I could start over)

Road map To Make ChatGpt

Linkedin: https://www.linkedin.com/in/alihassanofficial/

0.Basic Requirements

1. python

2. NumPy,

3.Pandas,

4.Matplotlib,

5.Seaborn

6. Git, GitHub

1. Core Concepts

1. Perceptron
2. MLP and its Notation
3. Forward Propagation
4. Backpropagation
5. Chain Rule of Derivative in Backpropagation
6. Vanishing Gradient Problem
7. Exploding Gradient

Activation Functions

List of Activation Functions

1. Linear Function
2. Binary Step Function
3. Sigmoid Function (Logistic Function)
4. Tanh (Hyperbolic Tangent Function)
5. ReLU (Rectified Linear Unit)
6. Leaky ReLU
7. Parametric ReLU (PReLU)
8. Exponential Linear Unit (ELU)
9. Scaled Exponential Linear Unit (SELU)
10. Softmax
11. Swish.
12. SoftPlus
13. Mish
14. Maxout
15. GELU (Gaussian Error Linear Unit)

16. SiLU (Sigmoid Linear Unit)
17. Gated Linear Unit (GLU)
18. SwiGLU
19. Mish Activation Function

Derivative of Activation Functions

Properties of Activation Functions

1. Saturating vs Non-Saturating
2. Smooth vs Non-Smooth
3. Generalized vs Specialized
4. Underflow and Overflow
5. Undefined and Defined
6. Computationally Expensive vs Inexpensive.
7. 0-Centered and Non-0-Centered
8. Differentiable vs Non-Differentiable
9. Bounded and Unbounded
10. Monotonicity
11. Linear Vs Non Linear

Ideal Activation Function Characteristics

1. Non-Linearity
2. Differentiability
3. Computational Efficiency
4. Avoids Saturation
5. Non-Sparse (Dense) Gradients
6. Centered Output (0-Centered)
7. Prevents Exploding Gradients
8. Monotonicity (Optional)
9. Sparse Activations (Optional)
10. Resilience to Outliers
11. Noise Robustness
12. Stable Training Dynamics
13. Minimal Parameter Dependency
14. Compatibility with Modern Techniques
15. Efficient in Hardware
16. The Function Must Be Continuous and Infinite in Domain
17. Vanishing Gradient Problem
18. Dynamic Range Adaptation
19. Scalability to Deeper Networks
20. Biological Plausibility (Optional)
21. Simplicity in Implementation

22. Gradient Smoothness
23. Compatibility with Unsupervised Objectives

Loss Functions

1. **Mean Squared Error (MSE)**
2. **Mean Absolute Error (MAE)**
3. **Root Mean Squared Error (RMSE)**
4. **Root Mean Squared Log Error (RMSLE)**
5. **Huber Loss**
6. **Hinge Loss**
7. **Binary Cross-Entropy (BCE)**
8. **Categorical Cross-Entropy**
9. **Focal Loss**
10. **Contrastive Loss**
11. **KL Divergence (Kullback-Leibler Divergence)**
12. **Triplet Loss**
13. **Smooth L1 Loss**:
14. **Dice Loss**:

Optimizers

1. **Gradient Descent**
2. **Stochastic Gradient Descent (SGD)**
3. **Mini-Batch Gradient Descent**
4. **Exponentially Weighted Moving Average (EWMA)**
5. **Gradient Descent with Momentum**
6. **Nesterov Accelerated Gradient**
7. **AdaGrad (Adaptive Gradient)**
8. **RMSProp (Root Mean Squared Propagation)**
9. **AdaDelta**
10. **Adam (Adaptive Moment Estimation)**
11. **Nadam (Nesterov-accelerated Adaptive Moment Estimation)**
12. **LAMB (Layer-wise Adaptive Moments)**:
13. **SGDW/AdamW**

Improving Performance of Neural Networks

- Effect of Batch Size on Training
- Memoization

Weight Initialization

1. Zero Initialization

2. Non-Zero Constant Value Initialization
3. Random Initialization (with small values, large values)
4. Xavier (Glorot) Initialization
5. He Initialization
6. LeCun Initialization
7. Uniform Initialization
8. Normal (Gaussian) Initialization
9. Bilinear Initialization
10. Orthogonal Initialization

Regularization

1. L1 Regularization (Lasso)
2. L2 Regularization (Ridge) (weight decay)
3. Elastic Net Regularization
4. Dropout
5. Early Stopping
6. Data Augmentation
7. Batch Normalization
8. Residual Connections
9. Label Smoothing
10. Parameter Sharing
11. Weight Constraint
12. Adversarial Training

Normalization

1. Normalizing Inputs
2. Batch Normalization (BatchNorm)
3. Layer Normalization (LayerNorm)
4. Instance Normalization (InstanceNorm)
5. Group Normalization (GroupNorm)
6. RMSNorm
7. Filter Response Normalization
8. Weight Normalization

Other Techniques

Gradient Clipping and Gradient Checking

Hyperparameter Tuning

Learning Rate Scheduling

1. Step Decay
2. Exponential Decay
3. Cosine Annealing
4. Cyclical learning rate
5. OneCycleLR
6. Warmup

Convolutional Neural Networks (CNNs):

1. **Convolutional Layer:**
2. **Filter (Kernel):**
3. **Stride:**
4. **Padding:**
5. **Feature Map:**
6. **Pooling Layer(Max Pooling, Average Pooling)**

Data Augmentation in CNN

Pretrained models in CNN

What is Transfer Learning? Fine Tuning Vs Feature Extraction

Famous CNN Architectures

1. LeNet
2. AlexNet
3. VGGNet
4. ResNet
5. U-Net
6. GoogleNet (Inception)
7. DenseNet
8. EfficientNet
9. MobileNet
10. ShuffleNet

Autoencoders and Variational Encoders

1.Autoencoders

1. Basic Autoencoders
2. Denoising Autoencoders
3.Sparse Autoencoders
4.Convolutional Autoencoders

5.Stacked Autoencoders

2.Variational Encoders

1.VAE Architecture
2.Encoder
3.Decoder
4.Latent Space
5.KL Divergence Loss

- Basic VAE
- Conditional VAE
- Beta-VAE

Recurrent Neural Networks (RNNs)

1. RNN
2. LSTM
3. GRU
4. Deep Stacked RNN, Bidirectional

Sequence-to-Sequence Models

1. Encoder-Decoder Architecture
2. Attention Mechanism

Natural Language Processing (NLP)

•**Tokenization**: Sentence tokenization, word tokenization, and subword tokenization (BPE, WordPiece).
Text Preprocessing: Lowercasing, stemming, lemmatization, stopword removal etc...
Text Vectorization:

1. One-Hot Encoding
2. Bag of Words (BoW)
3. TF-IDF
4. Word Embeddings (Word2Vec, GloVe, FastText)
5. Contextual Embeddings (ELMo, BERT, GPT, etc.)

Complete NLP Basics

Transformers

1. Vanilla Transformer

2. Vision Transformer
3. Swin Transformer
4. BERT (Bidirectional Encoder Representations from Transformers)
5. GPT, GPT-2, GPT-3
6. RoBERTa
7. DistilBERT
8. XLNet
9. T5

Evaluation Metrics

1. **Accuracy**
2. **Precision**
3. **Recall**
4. **F1 Score**
5. **Confusion Matrix**
6. **ROC Curve**
7. **AUC (Area Under the Curve)**
8. **Mean Squared Error (MSE)**
9. **Mean Absolute Error (MAE)**
10. **Root Mean Squared Error (RMSE)**
11. **R-squared (R^2)**
12. **Perplexity**
13. **BLEU Score**
14. **IoU (Intersection over Union)**
15. **Log Loss**
16. **Hamming Loss**
17. **Cohen's Kappa**
18. **Matthews Correlation Coefficient (MCC)**

Graph Neural Networks (GNNs)

- Graph Convolutional Networks (GCNs)
- Graph Attention Networks (GATs)
- GraphSAGE

Generative Adversarial Networks (GANs)

1. Vanilla GAN
2. DCGAN
3. LSGAN
4. WGAN

5. RLSGAN
6. CycleGAN
7. Pix2Pix
8. Conditional GAN
9. BigGAN
10. StyleGAN

CLIP (Contrastive Language-Image Pretraining)

Stable Diffusion

Dalle

Multomodel like paliGemma

Other Topics

- **Neural Style Transfer**
- **Self-supervised Learning**
- **Meta-learning**
- **Few-shot Learning**
- **Zero-shot Learning**

Siamese Neural Network

TensorBoard

MLflow

NeuroEvolution of Augmenting Topologies (NEAT)

AWS SageMaker

Confusion Matrix

PCA (Principal Component Analysis)

t-SNE

Techniques for Model Optimization

1. **Knowledge Distillation**
2. **Neural Architecture Search (NAS)**
3. **Quantization**
4. **Pruning**
5. **Low-rank Factorization**
6. **Model Compression**
7. **Model Quantization**
8. **Weight Clustering**
9. **Transfer Learning**
10. **Fine-tuning**
11. **Feature Extraction**
12. **Multitask Learning**

Reinforcement Learning

1. **Markov Decision Process (MDP)**
2. **Q-Learning**
3. **Deep Q-Network (DQN)**
4. **Policy Gradients**
5. **Actor-Critic Models**
6. **Proximal Policy Optimization (PPO)**
7. **Trust Region Policy Optimization (TRPO)**
8. **Reinforcement Learning from Human Feedback (RLHF)**
9. **Monte Carlo Methods**
10. **Temporal Difference Learning**
11. **Exploration vs. Exploitation**

Tools and Frameworks

1. PyTorch and PyTorch Lightning(TorchServe, Flask/FastAPI, Export to ONNX , Deployment with Docker, **PyTorch Mobile** or **TorchScript**, Kubernetes)
2. Hugging Face for pre-trained models
3. TensorBoard for visualization
4. MLflow for model tracking
5. ONNX for model deployment

MLOps and Industry Practices

1. CI/CD for Machine Learning Models
2. Monitoring and Scaling Models in Production

3. Using Cloud Platforms (AWS SageMaker, Google AI Platform, Azure AI)

Mathematics for AI

1.Linear Algebra (Matrices, Vectors).
2.Calculus (Derivatives, Gradients).
3.Probability and Statistics (Bayes' Theorem, Hypothesis Testing).

Bhanat Math And Maths by Prof. Iqbal Haider Bhatti