[GUIDE] AI curriculum for new staff engineer

Version 1.0: 05/12/2023

Module 1: Math Fundamental (Week 1 - 6)

1. **Linear Algebra [1] - Low**
   a. Vectors, matrices, eigenvalues, eigenvectors, vector spaces, linear transformations
   b. Matrix operations, determinant, inverse
2. **Calculus [2] - Low**
   a. Derivaties: product rule, quotient rule, chain rule, higher-order derivatives, implicit differentitation.
   b. Integrals: substitution, integration by parts.
   c. Partial derivatives, multiple integrals, gradient, divergence.
   d. First-order and higher-order ordinary differential equations.
   e. Laplace transforms.
3. **Probability and Statistics [3] - Low**
   a. Probability distributions: Bernoulli, Binomial, Poisson, Uniform, Normal, Exponential, Gamma.
   b. Descriptive statistics: Measures of central tendency, Measures of Dispersion, Percentiles and Quartiles.
   c. Hypothesis testing: Formulating Hypotheses, Types of Errors, Statistical Tets, P-Value, Confidence intervals.
4. **Numercial methods [4] - Medium**
   a. IEEE arithmetic
   b. Root Finding: Bisection method, newton's method

c. Systems of equations: Gaussian elimination, LU decomposition, Partial pivoting
d. Least-squares approximation
e. Interpolation: Polynomial interpolation, Piecewise linear interpolation, cubic spline
f. Integration: Elementary formulas, Composite rules
g. Ordinary differential equations: initial value problem, boundary value problem, eigenvalue problem

5. **Mathematical Optimization (Linear Programming) [5] - High**
    a. Simplex method, Duality theory, Convex analysis
    b. Game theory, Regression, Network flow problems
    c. Structural optimization, KKT system

Module 2: Machine Learning Fundamentals [6] (Week 2)

1. **Introduction to Machine learning - Low**
    a. Supervised learning, unsupervised learning, reinforcement learning.
    b. Feature engineering and selection.
2. **Supervised Learning Algorithms - Medium**
    a. Linear regression, logistic regression, decision trees, support vector machines.
    b. Model evaluation metrics.
3. **Unsupervised Learning Algorithms - Medium**
    a. Clustering (K-means, hierarchical), dimensionality reduction (PCA)
    b. Anomaly detection
4. **Model Training and Validation - Low**
    a. Cross-validation, hyperparameter tuning
    b. Overfitting and underfitting

Module 3: Image Processing Fundamentals [7] (Week 3 - 4)

1. **Intensity Transformations and Spatial Filtering - Low**
    a. Basic intensity transformation functions: image negatives, log transformations, power-law transformations.
    b. Histogram Processing: histogram equalization, local histogram processing.
    c. Fundamentals of spatial filters: smoothing (lowpass), sharpening (higpass) spatial filters, bandreject.
2. **Filtering in the Frequency domain - Medium**
    a. Fourier transform and sampled functions.
    b. 2-D DFT and IDFT.
    c. Selective filtering, fast Fourier transform.
3. **Image restoration and reconstruction - Medium**
    a. Noise models, noise reduction using Frequency domain filtering.
    b. Degradation function, inverse filtering, wiener filtering.
4. **Wavelet transform - Medium**
    a. Matrix-based transforms, correlation, walsh-hadamard transforms, slant transform, haar transform, wavelet transforms.
5. **Image Compression - Low**
    a. Huffman coding, arithmetic coding, LZW coding.
6. **Morphological Image - Medium**
    a. Erosion and dilation.
    b. Opening and closing.
    c. Morphological algorithms: Boundary Extraction, Convex Hull, Thinning, Thickening, Skeletons, Pruning.
7. **Image Segmentation - Medium**
    a. Point, Line and edge detection.
    b. Region splitting and merging algorithm, clustering and superpixels, graph cuts, Morphological Watersheds.

Module 4: Deep Learning [8] (Week 4 - 6)

1. **Fundamentals - High**
    a. Activation functions: ReLU, GELU, Sigmoid, Tanh, Leaky ReLU, Swish, Mish, Softmax.
    b. Multilayer Perceptrons, CNNs, RNNs.
    c. Transformers: BERT, RoBERTa, XLNet, GPT, Longformer, PaLM, Transformer-XL, CLIP.
    d. Attention mechanisms: Deformable attention, Neighborhood, local attention, linear attention, grouped-query, axial attention.
2. **Neural Networks - Medium**
    a. Basic architecture: AlexNet, VGG, Resnet, ResNext, Densenet, Mobilenet.
    b. LSTM, GRU, Bi-LSTM, Encoder-decoder, Sequence-to-Sequence, Beam search.
3. **Optimization - Medium**
    a. Convexity, gradient descent, stochastic gradient descent.
    b. Momentum, Adagrad, RMSProp, Adadelta, Adam, AdamW.
    c. Learning rate scheduling: Factor scheduler, multi factor scheduler, cosine scheduler, warmup.

Module 5: Natural Language Processing [9] (Week 6)

1. **Introduction to NLP - Low**
    a. Tokenization, stemming, lemmatization.
    b. Named Entity Recognition (NER), sentiment analysis.
2. **Word Embeddings - Low**
    a. Word2VEC, GloVe, and embeddings in NLP.
    b. Continuous Bag of Words (CBOW) and Skip-gram models.
3. **Sequence-to-Sequence Models - Medium**
    a. BERT, GPT.
4. **Language Models: - High**
    a. Overview of language modeling.
    b. N-gram models, Markov models.
    c. Modern Language Models: GPT, LLAMA, T5, XLNet.

Module 6: Computer Vision [10] (Week 7 - 8 - 9)

1. **Object Detection (2D/3D) - High**
    a. anchor-based, free anchor
    b. RCNN, Yolo, SSD, CenterNet.
    c. DeTR, DeTR3D, CenterPoint, 3DSSD.
    d. NMS, Soft-NMS, Hungarian matching.
    e. CLRNet, CondLaneNet.
2. **Feature Pyramid - Low**
    a. FPN, BiFPN, PAN, NAS-FPN.
3. **Generative models - Low**
    a. GAN, DCGAN, VAE.
    b. Diffusion model.
4. **Semantic Segmentation - Medium**
    a. UNET, HANET, PSPNet.

Module 7: Pytorch Framework [11] (Week 8 - 12)

1. **Introduction to PyTorch:**
    - Overview of PyTorch and its key features.
    - Tensor operations and basic usage.
2. **Building Neural Networks with PyTorch:**
    - Defining and training simple neural networks.
    - Loss functions, optimization, and backpropagation in PyTorch.
3. **Advanced PyTorch Concepts:**
    - Custom datasets and data loaders.
    - Transfer learning and fine-tuning pre-trained models.
4. **PyTorch for NLP and Computer Vision:**
    - Applying PyTorch to natural language processing tasks.
    - Leveraging PyTorch for computer vision applications.

Module 8: Robotic Framework [12] (Week 12 - 15)

1. **Sensors in robotics - Medium**
    a. Vision sensors (cameras), LiDAR, radar, IMU.
    b. Sensor fusion for robus perception.
2. **ROS Language - Medium**
    a. ROS Concepts: nodes, topics, and messages.
    b. Transformation and coordinate system.

Module 9: Practical Projects (Week 13 - 16)

1. **Dolphin Autopilot Architecture System.**
2. **Dolphin Perception Module.**
3. **Dolphin Localization Module.**
4. **Dolphin Planning Module.**
5. **Reception Chatbot Platform.**
6. **Reception Keyword Detection.**
7. **Drone Image Matching.**
8. **Drone Object Detection.**

Resources

[1] linearalgebra.pdf

[2] calculus.pdf

[3] python_probability_statistics.pdf

[4] numerical_analysis_9th.pdf

[5] linear_programming.pdf

[6] mml-book.pdf

[7] image_processing.pdf

[8] d2l-en.pdf

[9] nlp_book.pdf

[10] computer_vision.pdf

[11] pytorch-deep-learning

[12] ros2_tutorials