MACHINE LEARNING ESSENTIALS: FROM BASICS TO MODEL DEPLOYMENT

Module 1: Introduction to Machine Learning

Lesson 1 : What is Machine Learning?

- Overview of machine learning and its importance in data science.
- Applications of machine learning across industries.

Lesson 2 : Types of Machine Learning

- **Supervised Learning:** classification, regression.
- **Unsupervised Learning:** clustering, dimensionality reduction.
- **Reinforcement Learning:** understanding agent, reward, and environment.

Lesson 3 : Essential Concepts in ML

- Classification and regression problems.
- Important elements: data, features, model, evaluation.
- The curse of dimensionality and its impact on model performance.

Module 2: Exploratory Data Analysis (EDA)

Lesson 1 : Introduction to EDA

- The role of EDA in data science and machine learning.
- Techniques for understanding the dataset: univariate, bivariate, multivariate analysis.

Lesson 2 : Data Visualization

- Visualizing continuous variables: histograms, box plots.
- Visualizing discrete variables: bar plots, pie charts.
- Visualizing time-series variables: line plots.
- Correlation heatmaps for feature relationships.
- Data distribution plots for identifying skewness.

Module 3: Data Preprocessing and Wrangling

Lesson 1 : Data Cleaning and Preparation

- Handling missing values (mean/mode imputation, KNN, etc.).
- Dealing with outliers: z-score, IQR method.

Lesson 2 : Feature Engineering

- Adding new features based on domain knowledge.
- Feature transformations: log transformation, polynomial features, etc.

Lesson 3 : Feature Scaling and Normalization

- Scaling techniques: min-max scaling, standardization.
- Normalization and its impact on distance-based algorithms.

Lesson 4 : Encoding and Data Wrangling

- Encoding categorical variables: one-hot encoding, label encoding.
- Introduction to dummy variables.

Module 4: Feature Selection and Dimensionality Reduction

Lesson 1 : Feature Selection Techniques

- Filter methods: correlation matrix, chi-square test.
- Wrapper methods: recursive feature elimination (RFE).
- Embedded methods: regularization (Lasso, Ridge).

Lesson 2 : Dimensionality Reduction

- Introduction to Principal Component Analysis (PCA).
- Sparse PCA and Kernel PCA.
- Singular Value Decomposition (SVD).
- Non-negative matrix factorization (NMF).

Module 5: Regression

Lesson 1 : Introduction to Regression

- Understanding regression as a supervised learning technique.
- Mathematical foundations of regression models.

Lesson 2 : Types of Regression

- Simple Linear Regression.
- Multiple Linear Regression.
- Polynomial Regression.
- Lasso and Ridge Regression.
- Elastic Net Regression.

Lesson 3 : Evaluation Metrics for Regression

- Mean Absolute Error (MAE).
- Mean Squared Error (MSE).
- Root Mean Squared Error (RMSE).
- R^2 and Adjusted R^2.

Module 6: Classification

Lesson 1 : Introduction to Classification Problems

- Overview of binary and multiclass classification.
- Bias-Variance tradeoff in classification models.

Lesson 2 : Common Classification Algorithms

- **K-Nearest Neighbors (KNN):** working with distance-based classification.
- **Logistic Regression:** mathematical background, optimization techniques, and stochastic gradient descent.
- **Support Vector Machines (SVM):** linear SVM, kernel-based SVM (RBF, polynomial, sigmoid).
- **Naive Bayes:** Bayes theorem, types of Naive Bayes (Bernoulli, Multinomial, Gaussian).
- **Decision Trees:** CART algorithm, impurity measures (Gini, cross-entropy), feature importance.
- **Random Forest and Ensemble Learning:** bagging, boosting (AdaBoost, Gradient Boost, XGBoost).

Lesson 3: Model Evaluation Metrics for Classification

- Confusion matrix, accuracy, F1-score, precision, and recall.
- Sensitivity, specificity, true positive rate (TPR), false positive rate (FPR).
- ROC curve and AUC (Area Under Curve).

Lesson 4 : Handling Imbalanced Datasets

- Techniques for dealing with imbalanced classes (oversampling, undersampling, SMOTE).

Module 7: Clustering and Unsupervised Learning

Lesson 1 : Introduction to Clustering

- Overview of unsupervised learning.
- Applications of clustering in business.

Lesson 2 : K-Means Clustering

- Finding the optimal number of clusters: elbow method.
- Cluster stability and optimizing inertia.

Lesson 3 : Hierarchical Clustering

- Agglomerative clustering and dendrograms.
- DBSCAN clustering for non-spherical clusters.

Lesson 4 : Association Rules and Market Basket Analysis

- Apriori algorithm for association rule learning.
- Implementing recommendation engines using collaborative filtering.

Module 8: Time Series Forecasting

Lesson 1 : Introduction to Time Series Data

- Components of time series: trend, seasonality, noise.
- Understanding stationary vs. non-stationary time series.

Lesson 2 : Time Series Models

- AR (Autoregressive model).
- ARMA, ARIMA (Autoregressive Integrated Moving Average).
- SARIMA, SARIMAX (Seasonal ARIMA).

Lesson 3 : Model Validation and Selection

- ACF and PACF for time series model diagnosis.
- Cross-validation techniques for time series.

Module 9: Model Selection and Hyperparameter Tuning

Lesson 1 : Cross-Validation

- Understanding cross-validation (k-fold, stratified cross-validation).
- Overfitting and underfitting.
- Bias-variance tradeoff in model selection.

Lesson 2 : Hyperparameter Tuning

- GridSearchCV and Randomized Search CV for hyperparameter optimization.
- Practical example using Scikit-learn.

Module 10: Model Deployment and Pipelines

Lesson 1 : Introduction to ML Pipelines

- Building and automating end-to-end machine learning pipelines.
- Integrating feature engineering, model training, and evaluation.

Lesson 2 : Joblib and Model Serialization

- Saving and loading machine learning models using Joblib.
- Model persistence for production-ready models.

Lesson 3 : Model Deployment in Flask

- Building a web application to deploy ML models using Flask.
- Implementing REST APIs to serve predictions.

CAPSTONE PROJECTS

Project 1 : Classification Problem: Predicting Customer Churn

- Build, train, and evaluate a classification model to predict customer churn.
- Handle imbalanced datasets and perform feature engineering.

Project 2 : Regression Problem: House Price Prediction

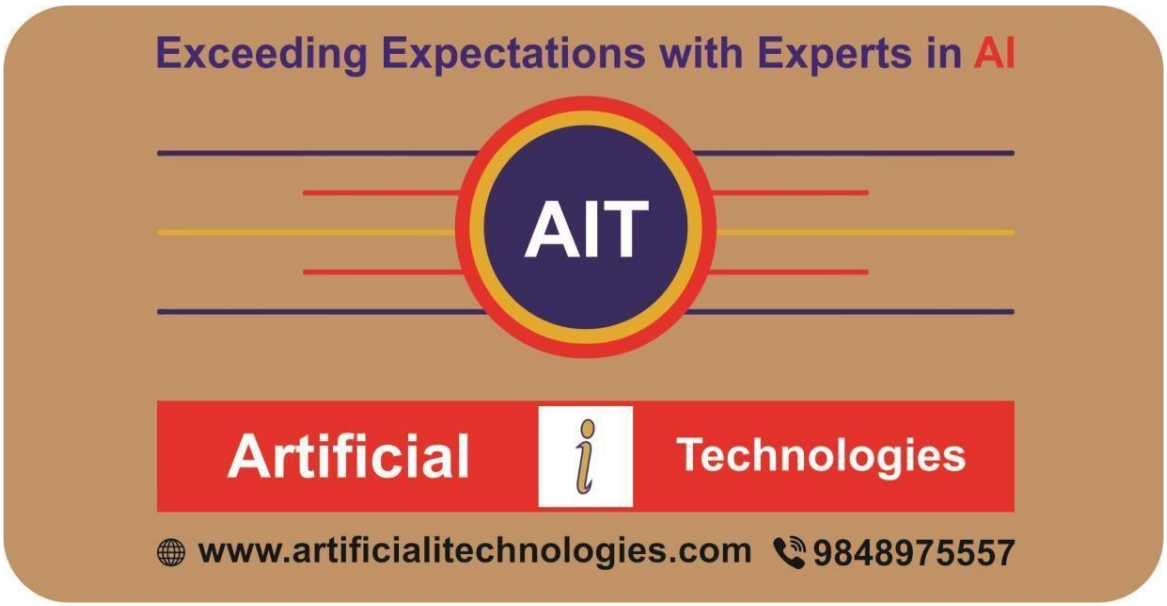
- Apply regression algorithms to predict house prices based on various features.
- Implement feature scaling, normalization, and model evaluation.

Project 3 : Time Series Forecasting: Stock Price Prediction

- Use ARIMA/SARIMA models to forecast future stock prices.
- Evaluate the model's performance on time series data.

Project 4 : Model Deployment: Flask Web App for Predicting Loan Approval

- Create a Flask web application that takes user input and predicts loan approval using an ML model.



DEEP LEARNING MASTERY: FROM BASICS TO ADVANCED TECHNIQUES

Module 1: Introduction to Deep Learning and Neural Networks

Lesson 1 : Introduction to Deep Learning

- Overview of machine learning and deep learning.
- The evolution of deep learning and its importance in AI.
- Applications of deep learning in various industries.

Lesson 2 : Neural Networks: The Foundation

- Biological neurons vs. artificial neurons.
- Introduction to perceptrons: structure and basic learning rule.
- Limitations of the perceptron and introduction to Multilayer Perceptron (MLP).
- MLP architecture: hidden layers, output layers, and loss functions.

Lesson 3 : Activation Functions and Their Importance

- Introduction to activation functions (ReLU, Sigmoid, Tanh).
- Why activation functions are crucial in deep learning.
- Choosing the right activation function for your neural network.

Module 2: Training Neural Networks

Lesson 1 : Backpropagation and Gradient Descent

- Understanding forward and backward propagation.
- Introduction to cost functions and how they are minimized.
- Gradient Descent: the optimization process.
- Vanishing and exploding gradient problems.

Lesson 2 : Regularization Techniques

- Introduction to regularization (L1, L2, dropout) to prevent overfitting.
- Batch normalization: improving model stability.

Lesson 3 : Optimizers and Learning Rate Schedulers

- Overview of optimizers: SGD, Adam, RMSprop, etc.
- Understanding learning rate and its role in convergence.
- Hyperparameter tuning: fine-tuning optimizers and regularization.

Lesson 4 : Introduction to PyTorch

- Why PyTorch? Overview of its features and advantages.
- Setting up PyTorch and creating your first neural network.
- Practical example: building and training a simple model with PyTorch.

Module 3: Deep Learning with TensorFlow

Lesson 1 : Introduction to TensorFlow

- What is TensorFlow? Advantages of using TensorFlow for deep learning.
- Installation and setup of TensorFlow.
- Basic TensorFlow syntax.

Lesson 2 : TensorFlow Computational Graphs

- Understanding graphs in TensorFlow: how they work.
- Variables and placeholders in TensorFlow.
- Practical: creating a simple computation graph.

Lesson 3 : Building Neural Networks with TensorFlow and Keras

- Introduction to Keras API: a high-level API for TensorFlow.
- Building and training neural networks with Keras.
- Practical example: constructing an MLP using TensorFlow/Keras.

Module 4: Artificial Neural Networks (ANNs)

Lesson 1 : ANN Architecture

- Overview of ANN architecture and its components.
- Forward propagation and backward propagation.
- Epoch, batch size, and learning rate.

Lesson 2 : Training and Fine-Tuning ANNs

- Dealing with vanishing gradients during training.
- Tuning hyperparameters: number of layers, neurons, learning rate.
- Choosing the right activation function for different layers.
- Practical exercise: building and tuning an ANN for a classification problem.

Module 5: Recurrent Neural Networks (RNNs)

Lesson 1 : Introduction to RNNs

- What is an RNN? Understanding its architecture and applications.
- RNN vs. ANN: when to use which.
- Backpropagation through time (BPTT) and the challenges.

Lesson 2 : Long Short-Term Memory (LSTM) and GRU

- Introduction to LSTMs and their importance in sequence modeling.
- GRUs: a simplified alternative to LSTMs.
- Implementing LSTM and GRU in PyTorch and TensorFlow.
- Practical exercise: time-series prediction using LSTM.

Lesson 3 : Advanced RNN Architectures

- Bidirectional RNNs: improving performance on sequential tasks.
- Sequence-to-sequence models (Encoder-Decoder architecture).
- Practical exercise: building a sequence-to-sequence model for machine translation.

Module 6: Convolutional Neural Networks (CNNs)

Lesson 1 : Introduction to CNNs and Image Processing

- Basics of image processing: filters, edges, and image histograms.
- Introduction to CNNs: convolution layers, pooling, and fully connected layers.
- Understanding the ImageNet dataset and image classification tasks.

Lesson 2 : Deep Dive into CNN Architectures

- Exploring popular CNN architectures: AlexNet, VGG, ResNet, etc.
- How to design and implement your own CNN from scratch.
- Practical exercise: image classification using a custom CNN.

Lesson 3 : Transfer Learning and Pre-Trained Models

- What is transfer learning and when to use it?
- Using pre-trained models for image classification.
- Practical exercise: leveraging a pre-trained ResNet for image classification.

Module 7: Transformers and BERT for NLP

Lesson 1 : Introduction to Transformers

- What are transformers? Overview of the transformer architecture.
- Self-attention mechanism and its importance.
- Practical application of transformers in NLP tasks.

Lesson 2 : BERT (Bidirectional Encoder Representations from Transformers)

- Understanding BERT architecture and how it improves NLP tasks.
- Pre-training and fine-tuning BERT for specific NLP tasks.
- Practical exercise: text classification using BERT.

Lesson 3 : Advanced NLP with Generative AI (ChatGPT)

- Introduction to generative models (GPT, GPT-3, ChatGPT).
- Understanding the generative process for text generation.
- Practical exercise: building a simple text generation model.

Module 8: Reinforcement Learning

Lesson 1 : Introduction to Reinforcement Learning

- Overview of reinforcement learning concepts (agent, environment, reward).
- Q-learning and policy gradients.

Lesson 2 : RL in Computer Vision

- Applying reinforcement learning techniques in image-based tasks.
- Using OpenAI Gym and PyTorch to build reinforcement learning models.
- Practical exercise: training a deep reinforcement learning agent.

Module 9: Deep Learning Applications in Computer Vision

Lesson 1 : Advanced Image Classification

- Implementing complex CNN architectures for image classification.
- Fine-tuning CNNs for small datasets.
- Practical exercise: image classification using data augmentation techniques.

Lesson 2 : Object Detection and YOLO

- Introduction to object detection: YOLO (You Only Look Once) algorithm.
- Installing and setting up YOLO for object detection tasks.
- Practical exercise: real-time object detection with YOLOv3.

Lesson 3 : Transfer Learning and Pre-trained Models

- Transfer learning for image classification using VGG, ResNet, Inception, etc.
- Fine-tuning pre-trained models for specific tasks.
- Practical exercise: transfer learning using TensorFlow and Keras.

Module 10: Capstone Projects

Capstone 1 : Image Classification with CNNs

- Build a CNN for image classification using TensorFlow.
- Fine-tune the CNN using hyperparameter tuning.
- Evaluate model performance on a custom dataset.

Capstone 2 : Text Generation with RNNs

- Create a text generation model using LSTM and RNN.
- Train the model on a large text corpus.
- Generate text based on input prompts.

Capstone 3 : Object Detection with YOLO

- Build a real-time object detection system using YOLOv3.
- Implement the system with live video feed from a webcam.
- Deploy the model on edge devices (Raspberry Pi or Jetson Nano).

MASTERINGCOMPUTERVISIONWITHOPENCV

Module1:IntroductiontoComputerVisionandOpenCV

Lesson1:HumanVisionvs.Computer Vision

- Understandinghowhumanvisionworks.
- Keydifferencesbetweenhumanvisionandcomputer vision.
- Real-world applications of computer vision (autonomous cars, healthcare, surveillance).

Lesson2:Introductionto OpenCV

- WhatisOpenCV?OverviewanditsimportanceinComputer Vision.
- SettingupOpenCVonyoursystem(Windows,macOS,Linux).
- FirststepswithOpenCV:reading,displaying,andsavingimages.

Lesson 3:Image Basicsin OpenCV

- Workingwithimages:loading,displaying,andsavingimages.
- Manipulatingpixelsandchannels.
- Grayscale,RGB,andothercolorspaces.

Module2: Image ProcessingwithOpenCV

Lesson1:DrawingandWritingon Images

- Drawingshapes:rectangles,circles,lines,andpolygons.
- Addingtexttoimages.
- Practicalexercise:overlayingshapesandtextonimages.

Lesson 2:Basic Image Transformations

- Resizing,scaling,androtatingimages.
- Croppingandflippingimages.
- Imageblurringand sharpening.

Lesson3:ThresholdingandContours

- Applyingimagethresholding(binary,adaptive).
- Detectinganddrawingcontours.
- Findingandlabelingobjectsinimagesusingcontours.

Module3:AdvancedImageProcessing Techniques

Lesson1:MorphologicalOperations

- Erosion,dilation,opening,andclosingtechniques.
- Practical use cases of morphological transformations (noise reduction, image enhancement).

Lesson 2:Edge Detection

- ApplyingedgedetectionusingCannyandSobelfilters.
- DetectinglinesandshapesinimagesusingtheHough Transform.
- Objectdetectionusingcontoursand edges.

Lesson3:ImageHistogramsandEqualization

- Understandingimagehistogramsandtheirimportance.
- Histogramequalizationforimagecontrastenhancement.
- Applyinghistogram-basedimageenhancementtechniques.

Module4:ConvolutionalNeuralNetworks(CNNs)

Lesson1:IntroductiontoCNNs

- WhatareCNNs?Theirimportanceinimage processing.
- Real-worldapplicationsofCNNsinimage recognition.
- OverviewofCNNarchitectures(AlexNet,VGG,ResNet).

Lesson2:CNN ArchitectureBreakdown

- Understandingconvolutionallayers,kernels,andfilters.
- Poolinglayers:MaxPooling,AveragePooling,GlobalPooling.
- Fullyconnectedlayersandsoftmaxforclassification.

Lesson3:Striding,Padding,andDataAugmentation

- ImportanceofstridingandpaddinginCNNs.
- Practicalimageaugmentationtechniques(rotation,flipping,zooming).
- TrainingCNNswithaugmenteddatasetsforrobustness.

Module5:DeepLearningandObjectDetectionwith OpenCV

Lesson 1:Face Detection with OpenCV

- IntroductiontoHaarCascadesforfacedetection.
- Implementingfacedetectionusingpre-trainedHaarCascades.
- Real-timefacedetectionusingOpenCVandwebcam input.

Lesson2:YOLOv3AlgorithmforObject Detection

- IntroductiontotheYOLO(YouOnlyLookOnce) algorithm.
- ImplementingYOLOv3withOpenCVforobjectdetection.
- Real-timeobjectdetectionwithYOLOand OpenCV.

Lesson3:ObjectTrackingwithOpenCV

- Understandingobjecttrackingconcepts.
- ImplementingobjecttrackingusingOpenCV'sbuilt-inalgorithms (BOOSTING, MIL, KCF, etc.).
- Real-timeobjecttrackingusingwebcamandOpenCV.

Module6: VideoProcessingwithOpenCV

Lesson1:Basics ofVideoProcessing

- ReadinganddisplayingvideofileswithOpenCV.
- Extractingframesfromavideo.
- Workingwithwebcamsforreal-timevideoinput.

Lesson2:ObjectDetectionin Video

- Detectingmovingobjectsinvideousingbackground subtraction.
- Applyingobjectdetectionmodels(YOLO,HaarCascades)tovideostreams.
- Buildingasimpleobjectdetectionpipelineforvideo input.

Lesson3:Motion DetectionandTracking

- Implementingbasicmotiondetectionusingframedifferencing.
- Trackingmovingobjectsusingcontourdetection.
- AdvancedmotiontrackingusingOpticalFlowtechniques.

Module 7:ImageSegmentation andFeatureDetection

Lesson 1:ImageSegmentation

- Whatisimagesegmentation?Usecasesinmedicalimaging,satellite imagery, etc.
- Applyingbasicsegmentationtechniques(thresholding,watershed).
- Imagesegmentationusingk-meansclustering.

Lesson2:FeatureDetectionandMatching

- DetectingfeaturesusingSIFT,SURF,andORB.
- MatchingfeaturesbetweenimagesusingFLANNand BFMatcher.
- Buildingareal-timeimagematchingapplicationusingOpenCV.

Lesson3:Image StitchingandPanoramaCreation

- Imagestitchingtechniquesusingfeaturematching.
- Creatingpanoramasbyaligningandmergingmultiple images.
- Practicalexercise:stitchingmultipleimagesintoapanorama.

Module8:IntroductiontoReinforcementLearning

Lesson1:BasicsofReinforcement Learning

- Whatisreinforcementlearning?Keyconcepts(agent,environment, reward).
- Types of reinforcement learning algorithms (Q-learning, policy gradient).
- Applicationsofreinforcementlearningincomputervision.

Lesson2:ReinforcementLearningwithOpenAIGym

- SettingupandworkingwithOpenAIGymenvironments.
- BuildingasimpleRLagenttointeractwithavirtual environment.
- VisualizingRLagentbehaviorusingOpenCVand Python.

Lesson3:PolicyGradient Theory

- UnderstandingpolicygradientmethodsforRL.
- Implementingpolicygradientsinreinforcementlearning.
- ApplyingRLwithpolicygradientstoreal-worldscenarios.

Module9: MachineLearningIntegration withOpenCV

Lesson1:MachineLearningAlgorithmsforComputer Vision

- Introduction to ML algorithms (SVM, Random Forest, k-NN) for image classification.
- UsingOpenCV'sbuilt-inMLmodelsforbasicimageclassification.
- TrainingandevaluatingMLmodelsonimage datasets.

Lesson2:DeepLearningIntegration withOpenCV

- Usingpre-traineddeeplearningmodels(ResNet,MobileNet)inOpenCV.
- IntegratingTensorFloworPyTorchmodelswithOpenCVforimage recognition.
- Buildingareal-timeimageclassificationsystemusingOpenCVanddeep learning.

Lesson3:ModelDeploymentand Optimization

- ExportinganddeployingOpenCVmodelsinproduction.
- Modeloptimizationtechniques(quantization,pruning)forfasterinference.
- Deployingcomputervisionmodelsonedgedevices(RaspberryPi,Jetson Nano).

CAPSTONE PROJECTS

Capstone1:Real-TimeObjectDetection System

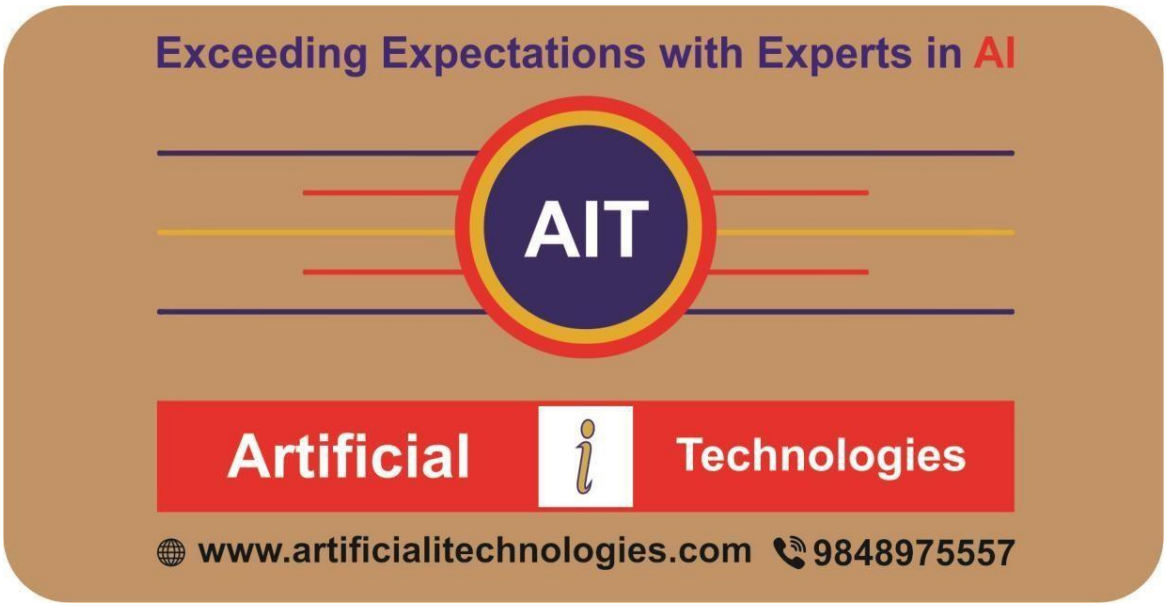
- Buildareal-timeobjectdetectionsystemusingYOLOandOpenCV.
- Implementmulti-objecttrackinganddetectionfromwebcaminput.
- Displaydetectionresultswithboundingboxesandlabelsonavideo stream.

Capstone2:Motion DetectionandTrackingSystem

- Createasystemtodetectandtrackmovingobjectsinreal-time.
- Useframedifferencingandopticalflowformotiondetection.
- Integratethesystemwithasecuritycamerafeed.

Capstone3:ImageRecognitionand Classification

- BuildanimageclassificationsystemusingCNNand OpenCV.
- Trainthesystemonadataset(CIFAR-10orMNIST)anddeployitin real-time.
- Optimizethemodelforperformanceanddeployitonedge devices.



ADVANCED NATURAL LANGUAGE PROCESSING (NLP)

Module 1: Introduction to Natural Language Processing

Lesson 1 : Introduction to NLP

- What is NLP? Real-world applications (chatbots, sentiment analysis, etc.).
- Key challenges in NLP (ambiguity, context, etc.).
- NLP pipeline: Tokenization, PoS tagging, parsing, etc.

Lesson 2 : Overview of Popular NLP Libraries

- Introduction to NLTK, spaCy, TextBlob, Gensim, and Transformers.
- Installing and setting up these libraries in Python.
- Choosing the right NLP library for different use cases.

Module 2: Text Cleaning and Preprocessing

Lesson 1 : Text Cleaning Techniques

- Removing punctuation, stop words, and special characters.
- Handling contractions and converting text to lowercase.
- Regular expressions for advanced text cleaning.

Lesson 2 : Tokenization

- What is tokenization? Importance in NLP tasks.
- Tokenizing words and sentences using NLTK, spaCy, and TextBlob.
- Dealing with complex tokens: handling emojis, hashtags, and mentions.

Lesson 3 : Stemming and Lemmatization

- Difference between stemming and lemmatization.
- Stemming using NLTK and spaCy.
- Lemmatization with NLTK, spaCy, and TextBlob.

Module 3: Feature Extraction and Vectorization

Lesson 1 : Bag of Words

- What is Bag of Words (BoW)? Use cases in text classification.
- Implementing BoW using scikit-learn.
- Limitations of BoW and handling high-dimensional data.

Lesson 2 : TF-IDF Vectorization

- Understanding Term Frequency-Inverse Document Frequency (TF-IDF).
- Implementing TF-IDF with scikit-learn.
- TF-IDF with n-grams (Unigram, Bigram, Trigram).

Lesson 3 : Word Embeddings: Word2Vec, GloVe, and FastText

- Introduction to word embeddings: capturing semantic relationships.
- Implementing Word2Vec using Gensim.
- Understanding GloVe embeddings and when to use them.
- Using pre-trained embeddings (Word2Vec, GloVe).

Lesson 4 : Document Embeddings with Doc2Vec

- Introduction to document-level embeddings (Doc2Vec).
- Implementing Doc2Vec using Gensim.
- Applications in text classification, sentiment analysis, etc.

Module 4: Text Classification and Topic Modeling

Lesson 1 : Text Classification Basics

- Overview of text classification tasks.
- Building a basic text classifier using Bag of Words or TF-IDF.
- Evaluating text classification models (precision, recall, F1-score).

Lesson 2 : Topic Modeling with LDA and NMF

- Understanding Latent Dirichlet Allocation (LDA) for topic modeling.
- Implementing LDA with Gensim.
- Non-negative Matrix Factorization (NMF) for topic modeling.

Lesson 3 : Advanced Text Classification with Embeddings

- Text classification using Word2Vec, GloVe, and FastText embeddings.
- Using deep learning models (RNN, CNN) for text classification.
- Evaluating deep learning-based text classifiers.

Module 5: Named Entity Recognition (NER) and Part-of-Speech (PoS) Tagging

Lesson 1 : Part-of-Speech Tagging

- Introduction to PoS tagging and its importance.
- Implementing PoS tagging using NLTK and spaCy.
- Customizing PoS tagging for specific use cases.

Lesson 2 : Named Entity Recognition (NER)

- What is NER? Use cases in extracting entities (names, dates, organizations, etc.).
- Implementing NER with NLTK and spaCy.
- Custom NER models with spaCy.

Lesson 3: Dependency Parsing

- Introduction to dependency parsing and its role in understanding sentence structure.

- Implementing dependency parsing using spaCy.
- Use cases of dependency parsing in complex NLP tasks.

Module 6: Sentiment Analysis and TextBlob

Lesson 1 : Introduction to Sentiment Analysis

- Overview of sentiment analysis: applications and challenges.
- Sentiment analysis using lexicons and rule-based approaches.
- Implementing sentiment analysis with TextBlob and VADER.

Lesson 2 : Twitter Sentiment Analysis

- Preprocessing tweets (hashtags, mentions, emojis).
- Using NLTK's Twitter Corpus for text cleaning and tokenization.
- Performing sentiment analysis on tweets using TextBlob.

Lesson 3 : Advanced Sentiment Analysis with Machine Learning

- Building machine learning models for sentiment classification.
- Using deep learning models (LSTMs, BERT) for sentiment analysis.
- Evaluating sentiment analysis models on real-world datasets.

Module 7: Transformer Models for NLP

Lesson 1 : Introduction to Transformer Models

- Overview of transformer models (BERT, GPT, T5, etc.).
- Advantages of transformers over traditional NLP models.
- Pre-trained transformers and transfer learning in NLP.

Lesson 2: Using BERT for NLP Tasks

- Fine-tuning BERT for text classification.
- Performing Named Entity Recognition (NER) using BERT.
- Using BERT for question answering and text summarization.

Lesson 3 : GPT Models for Text Generation

- Introduction to GPT and generative models in NLP.
- Fine-tuning GPT models for text generation tasks.
- Evaluating the quality and coherence of generated text.

Module 8: NLP Applications and Case Studies

Lesson 1 : Text Summarization

- Introduction to extractive vs. abstractive text summarization.
- Building a text summarizer using spaCy and transformers.
- Applications of text summarization in content aggregation and reporting.

Lesson 2 : Machine Translation

- Overview of machine translation systems (rule-based, statistical, neural).
- Building a basic translator using transformers (BERT, T5).
- Evaluating translation quality and handling multilingual text.

Lesson 3 : Question Answering Systems

- Building QA systems using BERT and transformers.
- Pre-processing text for QA tasks.
- Evaluating QA systems in real-world scenarios.

Module 9: NLP in Production

Lesson 1 : Deploying NLP Models

- Best practices for deploying NLP models.
- Using REST APIs and Flask/Django for model deployment.
- Deploying NLP models on cloud platforms (AWS, GCP, Azure).

Lesson 2 : Model Optimization and Scalability

- Optimizing NLP models for faster inference (quantization, pruning).
- Scaling NLP systems to handle large datasets.
- Monitoring and improving NLP models in production environments.

Module 10: Capstone Projects

Capstone 1: Twitter Sentiment Analysis Dashboard

- Build a real-time dashboard for sentiment analysis using live Twitter data.
- Use TextBlob and transformers for sentiment prediction.
- Create a visual dashboard using Power BI or Tableau.

Capstone 2: Text Summarization for News Articles

- Build a system that summarizes news articles using transformer models.
- Preprocess and clean the text before summarization.
- Evaluate the performance using human-written summaries.

Capstone 3: Named Entity Recognition for Legal Documents

- Implement a custom NER model to extract legal terms, clauses, and parties from contracts.
- Train a custom NER pipeline using spaCy and transformers.

Deploy the model for use in document review automation.

ADVANCED GENERATIVE AI: CONCEPTS, TECHNIQUES, AND APPLICATIONS

Module 1: Introduction to Generative AI

Lesson 1 : What is Generative AI?

- Overview of AI and machine learning.
- Key differences between discriminative and generative models.

Lesson 2 : History and Evolution of Generative Models

- Milestones in generative AI, from statistical models to deep learning.

Lesson 3 : Applications of Generative AI

- Creative arts, content creation, music generation, and text-to-image models.

Module 2: Generative Models: Theory and Fundamentals

Lesson 1 : Understanding Probabilistic Models

- Bayesian inference and Markov models.
- Generative vs. discriminative models (Logistic regression vs. Naive Bayes).

Lesson 2 : Generative Adversarial Networks (GANs)

- Theory behind GANs (Generator and Discriminator).
- Variants of GANs (DCGAN, StyleGAN, Conditional GANs, CycleGAN).

Lesson 3 : Variational Autoencoders (VAEs)

- Theory and implementation of VAEs.
- Comparing VAEs and GANs.

Lesson 4 : Diffusion Models

- Introduction to denoising diffusion probabilistic models (DDPMs).
- Recent trends in diffusion-based image synthesis.

Module 3: Prompt Engineering

Lesson 1 : Introduction to Prompt Engineering

- What is prompt engineering and why it's critical for generative AI.
- Understanding the role of prompts in large language models (LLMs).

Lesson 2 : Crafting Effective Prompts for Text Generation

- Techniques for writing concise, clear, and informative prompts.
- Prompt chaining and using few-shot learning.

Lesson 3 : Advanced Prompting Strategies

- Using templates and context for consistent results.
- Dynamic prompts for real-time data interaction.

Lesson 4 : Applications of Prompt Engineering in Chatbots and Content Creation

- Leveraging prompt engineering for intelligent chatbots and dialogue systems.
- Best practices for generating specific types of content (e.g., summaries, creative writing).

Lesson 5 : Testing and Optimizing Prompts

- Methods for testing prompts to achieve optimal performance.
- Prompt optimization and A/B testing for varied responses.

Module 4: Text Generation Models

Lesson 1 : Recurrent Neural Networks (RNNs) and LSTMs

- Understanding sequential data and time-series generation.
- Limitations of RNNs in generating long-form content.

Lesson 2 : Transformers and Attention Mechanisms

- The Transformer architecture and why it replaced RNNs.
- Self-attention mechanism and sequence-to-sequence learning.

Lesson 3 : GPT (Generative Pre-trained Transformer)

- Pre-training and fine-tuning stages in GPT.
- Evolution from GPT to GPT-4 and beyond.

Lesson 4 : Large Language Models (LLMs) and Chatbots

- Application of models like GPT-4, PaLM, LLaMA.
- Implementing chatbots and personalized response generation.

Module 5: Image Generation Models

Lesson 1 : Convolutional Neural Networks (CNNs) for Image Generation

- How CNNs are used to generate images.
- Image classification, segmentation, and synthesis tasks.

Lesson 2 : Generating Images using GANs

- Training a GAN to generate realistic images.
- Use of DCGAN, StyleGAN for high-quality image generation.

Lesson 3 : Text-to-Image Models

- Introduction to DALL-E, Stable Diffusion, and MidJourney.
- How text prompts are used to generate high-fidelity images.

Module 6: Retrieval-Augmented Generation (RAG)

Lesson 1 : What is RAG?

- Overview of retrieval-augmented generation.
- How RAG combines retrieval from external documents with generative models.

Lesson 2 : Architecture of RAG Models

- Two-stage process: retrieval (encoder) and generation (decoder).
- Combining transformers with retrieval models.

Lesson 3 : Applications of RAG

- Real-world use cases in question-answering systems, personalized search engines, and chatbot enhancements.

Lesson 4 : Implementing RAG Models

- Practical implementations using Hugging Face or PyTorch.
- Fine-tuning RAG models on custom datasets.

Module 7: Vector Databases and Semantic Search

Lesson 1 : Introduction to Vector Databases

- What are vector databases and why they are crucial in generative AI?
- Overview of key players: Pinecone, Milvus, Weaviate, and FAISS.

Lesson 2 : Storing and Querying with Vector Databases

- How embeddings are stored in vector databases.
- Use cases in semantic search and similarity matching.

Lesson 3 : Practical Implementation

- Building a vector-based search engine using Python and FAISS.
- Integration of vector databases with AI models for fast retrieval.

Module 8: LangChain and LLM Applications

Lesson 1 : Introduction to LangChain

- What is LangChain? How it simplifies large language model (LLM) integration.
- Key features: prompt templates, chains, memory, and agents.

Lesson 2 : Building Applications with LangChain

- Step-by-step guide to build custom LLM-powered apps.
- Creating multi-step chains, integrating memory, and handling user interactions.

Lesson 3 : Advanced LangChain Techniques

- Using agents for dynamic task automation.
- Working with external APIs and data sources.

Lesson 4 : LangChain in Production

- Best practices for scaling LangChain applications.
- Deployment strategies and integrating with cloud services.

Module 9: LlamaIndex (GPT Index)

Lesson 1 : Overview of LlamaIndex (formerly GPT Index)

- Introduction to LlamaIndex: a framework for indexing and querying large-scale document collections.
- Why LlamaIndex is essential for working with text data in LLMs.

Lesson 2 : Architecture of LlamaIndex

- How LlamaIndex builds indexes for efficient querying and retrieval.
- Working with document stores and vector databases.

Lesson 3 : Practical Applications of LlamaIndex

- Creating search interfaces using LlamaIndex.
- Real-world use cases: research assistants, knowledge databases, and document summarization.

Module 10: LangGraph

Lesson 1 : Introduction to LangGraph

- What is LangGraph? Building graph-based workflows for large language models.
- How it simplifies the orchestration of complex interactions across models.

Lesson 2 : Building Graph Workflows

- How to design workflows using LangGraph.
- Use cases: multi-agent systems, complex dialogue flows, and distributed computing.

Lesson 3 : Advanced LangGraph Concepts

- Integrating LangGraph with other tools like LangChain, LlamaIndex, and vector databases.
- Scaling LangGraph workflows for enterprise applications.

Module 11: Music, Video, and Art Generation

Lesson 1 : Music Generation Models

- Using RNNs, transformers, and VAEs for generating music sequences.
- Applications like JukeBox and MuseNet.

Lesson 2 : Video Generation

- Challenges in video synthesis (time and spatial coherence).
- Using GANs for video generation.

Lesson 3 : AI for Digital Art

- Style transfer and neural art.
- Using AI tools for creating dynamic artwork.

Module 12: Training, Fine-Tuning, and Evaluation

Lesson 1 : Fine-tuning Pre-trained Models

- Transfer learning and using pre-trained weights.
- Practical steps for customizing AI models.

Lesson 2 : Evaluating Generative Models

- Quantitative evaluation metrics (FID, IS, BLEU).
- Subjective evaluations and human feedback.

Lesson 3: Training Stable Generative Models

- Tips for stabilizing GAN training.
- Overfitting and generalization concerns.

Module 13: Ethical Considerations in Generative AI

Lesson 1 : Bias in Generative Models

- Understanding and mitigating bias in AI-generated content.

Lesson 2 : Deepfakes and Misinformation

- The dangers of malicious applications like deepfakes.
- Techniques to detect and prevent fake content.

Lesson 3 : Ownership and Copyright Issues

- Who owns AI-generated content?
- Legal implications of generative models.

CAPSTONE PROJECT

Choose from one of the following

- **Text Generation Capstone:** Build a custom text generation model using GPT.
- **Image Generation Capstone:** Fine-tune a GAN for generating high-quality images.
- **Video Synthesis Capstone:** Build a simple video generation model with GANs.
- **RAG & LangChain Capstone:** Build an application combining RAG, LangChain, and vector databases for enhanced search or chatbot functionality.
- **Prompt Engineering Capstone:** Develop a sophisticated prompt-based system for automating a task using an LLM.