COEP Technological University Pune

(A Unitary Public University of Govt. of Maharashtra)

School of Engineering & Technology

Curriculum Structure

One Year Post Graduate Diploma in Data Science and AI

Duration

1 Year (3 Semesters)

Computer Science and Engineering

(Effective from: A.Y. 2025-26)

in

PG Diploma in Data Science and AI

Computer Science and Engineering

Sr.No.	Subject	Credits		Total Credits	Duration
		Theory	Lab		
Trimester-1					
1.	Programming Fundamentals using Python	1	2	3	16 weeks
2.	Mathematics for Data Science	3	-	3	
3.	Data Warehousing using MySQL and MongDB	2	1	3	
4.	Machine Learning Fundamentals	2	1	3	
5.	Data Visualization using Tableau and PowerBI	1	2	3	
		Total Credits		15	
Trimester-2					
6.	Introduction to Artificial Intelligence	3	-	3	16 weeks
7.	Big Data Technologies	2	1	3	
8.	Deep learning	3	-	3	
9.	Natural Language Processing	2	1	3	
		Total Credits		12	
Trimester-3					
11.	Ethical AI and Advanced Machine Learning	3	-	3	16 Weeks
12.	Programming Fundamentals using R	1	2	3	
13.	Computer Vision	3	-	3	
14.	Industry Project	9	-	9	
		Total Credits		18	
PGDDSAI Course Total Credits				45	

Trimester-1

(PGD-01) Mathematics for Data Science

Teaching Scheme
Lectures: 3Hrs/ Week

Evaluation Scheme
Theory: Test 1 : 20 Marks,
Test 2: 20 Marks, ESE: 60 Marks

Course Outcomes

Students will be able to:

1. Design appropriate automata for modeling the solution for various computational problems.
2. Apply transformation between multiple representations of automata/machines.
3. Make use of the pumping lemma to show that a language is not regular/context-free.
4. Distinguish different formal computing languages and classify their respective types.
5. Describe the limitations of a computing machine in terms of language recognition.

Course Contents

Unit 1: Linear Algebra for Data Science
Vectors, scalars, Vector projection, Orthogonal vectors, Linear combination, Linear span, Linear independence, Cosine similarity; Matrices: Basics of Matrices, Scalar and matrix, Types of Matrix, Matrix Operations, Linear transformation, Eigen values & eigen vectors
Hands-on: Creating different vectors and matrix, Performing vector and matrix operations using Python.

Unit 2: Applied Linear Algebra
Matrix Decomposition, LU Decomposition, Singular Value Decomposition, Dimensionality Reduction, Principal Component Analysis, Linear Discriminant Analysis, Regression: Linear Regression, Ordinary Least Square Method, Multiple Regression, Polynomial Regression, Logistic Regression.

Unit 3: Descriptive Statistics
Measures of central tendency: mean and its types, median, mode; Measures of dispersion: range, mean absolute deviation, standard deviation, variance, quartiles, skewness, kurtosis; Measure of correlations: Correlation (Pearson and Rank) and covariance, Measures of Frequency Distribution: Histograms, Relative and cumulative frequency
Hands-on: Performing all descriptive statistics operations using Python.

Unit 4: Inferential Statistics and Scaling
Statistical Inference, Sample and Population, Data Sampling and its methods, Resampling Techniques, Hypothesis Testing: Null and Alternate Hypothesis, Student's T-Test, Z-Score and Z-Test, Chi-Square Test, Data Scaling: Normalization and Standardization
Hands-on: Data Sampling and resampling, Z-test and Chi-Square Test using Python.

Unit 5: Foundations of Probability:

Concepts of Probability, Types and examples, Probability axioms and basic principles, Sample space, events, and probability calculations, independent events, mutually exclusive events, marginal, conditional, and joint probability, Permutations and combinations Bayes' Theorem and its applications, Naive Bayes Classification method.
Hands-on: Calculating probabilities, applying Bayes' Theorem in Python

Unit 6: Probability Distributions and Time Series Analysis
Continuous random variables and probability distribution functions (PDF); Common continuous distributions: uniform, exponential, Poisson, normal; Standard normal distribution, t-distribution, chi-squared distribution; Cumulative distribution function (CDF); Conditional PDF and its properties. Introduction to time series data; Trends, seasonality, and stationarity; ARIMA modelling for forecasting.
Hands-on: Generating random numbers from continuous distributions, fitting distributions in Python, analysing time series data, fitting ARIMA models, and making forecasts in Python

Text Books:

- "Theory of computer science", E. V. Krishnamurthy, 2004, Affiliated East Press Publications, ISBN-10: 038791255X / ISBN-13: 978-0387912554.

Reference Books:

1. "Introduction to Probability and Statistics", William Mendenhall, Robert J. Beaver, Barbara M. Beaver, 2013, Cengage Learning, ISBN-10: 1305269470 / ISBN-13: 978-1305269477.
2. "Statistics", David Freedman, Robert Pisani, Roger Purves, 2007, W. W. Norton & Company, ISBN-10: 0393929728 / ISBN-13: 978-0393929720.
3. "Linear Algebra and Its Applications", David C. Lay, Steven R. Lay, Judi J. McDonald, 2019, Pearson, ISBN-10: 0135188925 / ISBN-13: 978-0135188927.
4. "Calculus: Early Transcendentals", James Stewart, 2015, Cengage Learning, ISBN-10: 1285741552 / ISBN-13: 978-1285741550.
5. "Time Series Analysis: Forecasting and Control", George E. P. Box, Gwilym M. Jenkins, Gregory C. Reinsel, Greta M. Ljung, 2015, Wiley, ISBN-10: 1118675029 / ISBN-13: 978-1118675021.

(PGD-02) Data Visualization using Powerbi

Teaching Scheme
Lectures: 3Hrs/ Week

Evaluation Scheme
Theory: Test 1 : 20 Marks,
Test 2: 20 Marks, ESE: 60 Marks

Course Outcomes

Students will be able to:

1. Understand the principles of data visualization and their application in Power BI.
2. Gain proficiency in data preparation, modeling, and visualization techniques.
3. Learn to design and develop interactive dashboards for effective data analysis and storytelling.
4. Master advanced features and best practices for creating professional-grade visualizations in Power BI.

Course Contents

Unit 1: Introduction to Power BI and Data Preparation
Overview of Power BI and its features; Installing and setting up Power BI Desktop
Importing data into Power BI from various sources; Introduction to data cleaning and transformation techniques in Power Query Editor.

Unit 2: Data Modeling and Basic Visualizations
Understanding data types, relationships, and modeling concepts; Creating relationships between tables and basic DAX calculations; Creating basic visualizations: bar charts, line charts, etc.; Formatting and customizing visuals

Unit 3: Advanced Visualizations and Dashboard Design
Introduction to advanced visuals: treemaps, heatmaps, etc.; Custom visuals and integrating them into Power BI; Dashboard design principles and layout; Adding interactivity with slicers, filters, and bookmarks.

Unit 4: Data Storytelling and Collaboration
Techniques for storytelling with data; Creating narrative-driven visualizations and reports; Sharing and publishing reports to Power BI Service; Collaborative features: sharing, commenting, and data security.

Unit 5: Advanced Features and Integration
Advanced DAX functions and calculations; Using parameters for dynamic analysis; Integrating Power BI with other tools (Excel, SQL Server, etc.); Real-world applications and case studies;

Unit 6: Final Project and Review
Capstone project: Design and develop a comprehensive data visualization solution;
Presentation of final projects and peer feedback; Reflection and discussion on lessons learned.

Reference Book:

1. "Power BI Datasets and Reports for Beginners", Paul Turley, 2020, Affiliated East Press Publications, ISBN-10: 038791255X / ISBN-13: 978-0387912554.
2. "Analyzing Data with Power BI and Power Pivot for Excel", Alberto Ferrari, Marco Russo, 2017, Affiliated East Press Publications, ISBN-10: 038791256X / ISBN-13: 978-0387912561.
3. "Mastering Microsoft Power BI", Brett Powell, 2018, Affiliated East Press Publications, ISBN-10: 038791257X / ISBN-13: 978-0387912578.

(PGD-03) Machine Learning Fundamentals

Teaching Scheme
Lectures: 3Hrs/ Week

Evaluation Scheme
Theory: Test 1 : 20 Marks,
Test 2: 20 Marks, ESE: 60 Marks

Course Outcomes

Students will be able to:

1. Understand the fundamental concepts of machine learning, including supervised learning, unsupervised learning, and reinforcement learning.
2. Apply various machine learning algorithms, such as linear regression, logistic regression, decision trees, and ensemble methods, to solve regression and classification problems.
3. Preprocess and clean datasets effectively, including handling missing values, encoding categorical variables, and scaling features.
4. Evaluate machine learning models using appropriate metrics, such as accuracy, precision, recall, F1-score, and ROC-AUC.
5. Perform model selection and hyperparameter tuning to optimize the performance of machine learning models.
6. Apply techniques for feature engineering, including dimensionality reduction and feature selection, to improve model performance and interpretability.

Course Contents

Unit 1: Basics of Machine Learning and Introduction to ML Lifecycle
Overview of machine learning concepts: supervised learning, unsupervised learning, and reinforcement learning; Understanding the machine learning lifecycle: data collection, data preprocessing, model building, model evaluation, and deployment; Applications of machine learning in various fields such as healthcare, finance, marketing, and robotics; Hands-on exercises and case studies on real-world applications of machine learning.

Unit 2: Introduction to NumPy, Pandas, and Data Preprocessing
Introduction to NumPy: arrays, indexing, operations, and functions; Introduction to pandas: data structures (Series, DataFrame), data manipulation, and data analysis; Data preprocessing techniques: handling missing values, data normalization, encoding categorical variables; Hands-on exercises on data preprocessing using NumPy and pandas.

Unit 3: Supervised Learning: Regression and Classification
Introduction to supervised learning: regression and classification; Linear regression: theory, implementation, and evaluation; Logistic regression, decision trees, and ensemble methods (Random Forest, Gradient Boosting) for classification; Evaluating regression and classification models: metrics (RMSE, MAE, accuracy, precision, recall, F1-score)

Unit 4: Unsupervised Learning
Introduction to unsupervised learning: clustering and dimensionality reduction; -means clustering, hierarchical clustering, and dimensionality reduction techniques (PCA, t-SNE); Evaluating clustering algorithms and applications of unsupervised learning; Hands-on exercises on clustering and dimensionality reduction.

Unit 5: Model Evaluation and Hyperparameter Tuning

Cross-validation techniques: k-fold cross-validation, stratified cross-validation; Hyperparameter tuning using grid search and randomized search; Model evaluation metrics: precision, recall, F1-score, ROC-AUC; Fine-tuning machine learning models for optimal performance.

Unit 6: Data Visualization with Matplotlib and Seaborn
Introduction to data visualization: importance and principles; Line plot, scatter plot, bar plot, histogram, and other types of plots using Matplotlib and Seaborn; Customizing plots: labels, colors, styles, annotations; Hands-on exercises on data visualization with real-world datasets.

Reference Book:

1. "Introduction to Machine Learning with Python" by Andreas C. Müller and Sarah Guido.
2. "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow", Aurélien Géron, 2019, Affiliated East Press Publications, ISBN-10: 038791258X / ISBN-13: 978-0387912585.
3. "Pattern Recognition and Machine Learning", Christopher M. Bishop, 2006, Affiliated East Press Publications, ISBN-10: 038791259X / ISBN-13: 978-0387912592.
4. "Python Machine Learning", Sebastian Raschka, Vahid Mirjalili, 2019, Affiliated East Press Publications, ISBN-10: 038791260X / ISBN-13: 978-0387912608.

(PGD-04) Programming Fundamental using Python

Teaching Scheme
Lectures: 3Hrs/ Week

Evaluation Scheme
Theory: Test 1 : 20 Marks,
Test 2: 20 Marks, ESE: 60 Marks

Course Outcomes

Students will be able to:

1. Understand the fundamental concepts of Python programming language, including syntax, data types, and control structures.
2. Write Python code to solve a variety of programming problems, ranging from simple tasks to more complex algorithms.
3. Utilize built-in data structures such as lists, tuples, dictionaries, and sets effectively to store and manipulate data.
4. Implement control flow mechanisms like loops, conditional statements, and functions to create modular and reusable code.
5. Demonstrate proficiency in handling files and performing file I/O operations using Python.
6. Develop an understanding of object-oriented programming (OOP) concepts and apply them to design and implement Python classes and objects.
7. Apply error handling and exception handling techniques to write robust and reliable Python code.

Unit 1: Introduction to Python:
Overview of Python programming language: history, features, and applications; Python programming language Installing and setting up the development environment; Basic syntax and data types in Python, Working with variables; Input/output operations in Python. **[6 hrs]**

Unit 2: Control Structures:
Operators: Arithmetic, logical, special, bitwise; Introduction to control structures: if, elif, else statements Looping constructs: for and while loops Nested loops and loop control statements Using break and continue statements; Practical exercises on implementing control structures. **[6 hrs]**

Unit 3: Data Structures:
Overview of built-in data structures: lists, tuples, dictionaries, and sets ; Operations and methods on data structures; Indexing, slicing, and iterating over data structures; Practical exercises on manipulating data structures **[6 hrs]**

Unit 4: Functions and Modules: Understanding functions in Python; Defining and calling functions; Function arguments and return values; Scope of variables in functions; Introduction to modules and importing modules in Python. Types of functions: user defined, recursion, lambda. **[6 hrs]**

Unit 5: File Handling: Reading from and writing to files in Python; File modes and file objects; Handling exceptions during file operations; Reading from and writing to files: text files, CSV files; Opening and closing files using the open() function; Reading file contents: read(), readline(), readlines() methods; Writing data to files: write() and writelines() methods; Practical exercises on file manipulation. **[6 hrs]**

Unit 6: Object-Oriented Programming: Introduction to object-oriented programming (OOP) concepts: Classes and objects in Python, Inheritance and polymorphism, Encapsulation and abstraction, Practical exercises on implementing OOP concepts in Python. **[6 hrs]**

Reference Book:

1. "Python Crash Course", Eric Matthes, 2019, Affiliated East Press Publications, ISBN-10: 038791261X / ISBN-13: 978-0387912615.
2. "Learning Python", Mark Lutz, 2013, Affiliated East Press Publications, ISBN-10: 038791262X / ISBN-13: 978-0387912622.
3. "Python Programming: An Introduction to Computer Science", John Zelle, 2016, Affiliated East Press Publications, ISBN-10: 038791263X / ISBN-13: 978-0387912639.

(PGD-05) Data Warehousing using MySQL and MongoDB

Teaching Scheme
Lectures: 3Hrs/ Week

Evaluation Scheme
Theory: Test 1 : 20 Marks,
Test 2: 20 Marks, ESE: 60 Marks

Course Outcomes

Students will be able to:

1. Understand the principles and fundamentals of relational and NoSQL databases.
2. Learn to design, implement, and administer databases using MySQL and MongoDB.
3. Develop proficiency in querying and manipulating data using SQL and MongoDB query language.
4. Gain practical experience in database administration tasks such as backup, recovery, and performance tuning.
5. Apply database management techniques to solve real-world business problems.

Unit 1: Introduction to Databases and Database Management Systems (DBMS)
Overview of databases: types, models, and architectures.; Introduction to relational databases and SQL (Structured Query Language); Introduction to NoSQL databases and MongoDB login **[6 hrs]**

Unit 2: Database Design and Data Modeling
Understanding database design principles: Conceptual Design using entity-relationship (ER) modeling; Logical Database Design using relational model; Hands-on exercises on creating database schemas and data models; **[6 hrs]**

Unit 3: Querying Relational Databases with SQL
Introduction to SQL: basic syntax, data retrieval, filtering, sorting, and aggregating; Writing SQL queries to retrieve and manipulate data in MySQL databases; Hands-on exercises on querying relational databases using SQL. **[6 hrs]**

Unit 4: Querying NoSQL Databases with MongoDB
Introduction to MongoDB query language (MongoDB Query API); Writing MongoDB queries to retrieve and manipulate data in MongoDB databases; Hands-on exercises on querying NoSQL databases using MongoDB; **[6 hrs]**

Unit 5: Database Administration and Management
Overview of database administration tasks: backup, recovery, security, and performance tuning; Managing MySQL databases: creating users, granting privileges, and configuring settings; Managing MongoDB databases: setting up authentication, configuring replication, and monitoring performance. **[6 hrs]**

Unit 6: Advanced Topics in Database Management
Indexing and optimization techniques for improving query performance; Data consistency and integrity constraints in relational databases; Scalability and sharding in NoSQL databases; Case studies and applications of database management in various industries. **[6 hrs]**

Reference Book:

1. "Learning MySQL: Get a Handle on Your Data", Seyed M.M. Tahaghoghi, Hugh E. Williams, 2020, O'Reilly Media, ISBN-10: 1492054848 / ISBN-13: 978-1492054849.
2. "MySQL 8 Cookbook: Over 150 Recipes for High-Performance Database Querying and Administration", Karthik Appigatla, 2021, Packt Publishing, ISBN-10: 1803234361 / ISBN-13: 978-1803234360.
3. "MongoDB: The Definitive Guide: Powerful and Scalable Data Storage", Kristina Chodorow, Michael Dirolf, 2020, O'Reilly Media, ISBN-10: 1491954469 / ISBN-13: 978-1491954461.
4. "MongoDB in Action: Covers MongoDB version 4.2", Kyle Banker, Peter Bakkum, Shaun Verch, 2016, Manning Publications, ISBN-10: 1617291609 / ISBN-13: 978-1617291609.
5. "Database Systems: Design, Implementation, and Management", Carlos Coronel, Steven Morris, Peter Rob, 2019, Cengage Learning, ISBN-10: 1337627909 / ISBN-13: 978-1337627900.

SEMESTER – II

(PGD-06) Natural Language Processing

Teaching Scheme
Lectures: 3Hrs/ Week

Evaluation Scheme
Theory: Mid Sem Exam: 30 Marks,
Assignment/Quizz: 20 Marks
ESE: 50 Marks

Course Outcomes

Students will be able to:

1. Apply the concepts of morphology, syntax, semantics, discourse & pragmatics of natural language.
2. Demonstrate semantics and pragmatics of English language for text processing.
3. Analyze and apply the Language Modelling for Natural Language Processing.
4. Apply the semantic techniques for Natural Language Processing.
5. Apply POS Tagging using Neural Model.
6. Apply Information Extraction models to the problems of Natural Language Processing.

Course Contents:

Unit 1: Overview and language modeling:
Overview: Origins, What is NLP, Fundamental and Scientific goals, Engineering goals, stages of NLP and challenges of NLP-Language and Grammar-Processing Indian Languages- NLP Applications-Information Retrieval with example.Language Modeling: Various Grammar- based Language Models-Statistical Language Model.Empirical Laws of language, zipf's law, Heap's law. **[6 hrs]**

Unit 2: Basic Text Processing:
Tokenization: Tokenization, Types of Tokenization, Need of Tokenization, Use cases of Tokenization, Challenges in Tokenization, feature extraction, word segmentation, text segmentation, normalization, case folding, Spelling Correction, Morphology. Lemmatization: lemmatization, spelling correction - dynamic programming approach for finding edit distance, N-gram Language Modeling- context sensitive spelling correction, probabilistic language model, auto completion prediction, Evaluation and perplexity, Smoothing techniques. **[6 hrs]**

Unit 3: Stemming: Stemming, Porters Algorithm
POS Tagging: Sequence labeling tasks of NLP, POS tagging, POS tag sets, Hidden Markov Mode, HMM characterization- Likelihood of a sequence, Best state sequence, Re-estimation. Models for Sequential tagging – Maximum Entropy, Conditional Random Field.
[6 hrs]

Unit:4
Information Extraction: Web Scraping; Introduction to Named Entity Recognition and Relation Extraction, Natural Language Generation-Architecture, Machine Translation, Sentiment and Emotion Analysis, Text Entailment, Question Answering, Analytics and Social Networks, Multilingual word embedding, Introduction to Transformer learning.
[6 hrs]

Unit 5: Computational Semantics: Lexical semantics: WordNet and FrameNet, Word Sense Disambiguation, Distributional Semantics & Word-Space models, ord/Sentence/Text embeddings. **[6 hrs]**

Unit 6:

Machine Translation: Rule based techniques, Statistical Machine Translation (SMT), Cross Lingual Translation Sentiment Analysis, Question Answering, Text Entailment, Discourse Processing, Dialog and Conversational Agents, Natural Language Generation.
[6 hrs]

Reference/Text Book:

1. Delip Rao, Brian McMahan, "Natural Language Processing with PyTorch", O'Reilly Media, 1st edition, February 2019, ISBN: 9781491978238.

2. Aman Kedia, Mayank Rasu, "Hands-On Python Natural Language Processing",Packt Publishing Limited, 1st edition, June 2020, ISBN-10 -1838989595.

(PGD-07) Big Data Analytics

Teaching Scheme

Lectures: 3Hrs/ Week

Evaluation Scheme

Theory: Mid Sem Exam: 30 Marks,
Assignment/Quizz: 20 Marks
ESE: 50 Marks

Course Outcomes

Students will be able to:

1. Explain the need of Big Data, challenges and technology stack of big data.
2. Explain and work on Hadoop Framework and ecosystems.
3. Explain and Analyze Big Data using Map Reduce programming framework in both Hadoop and Apache Spark.
4. Analyze large datasets using Apache Pig.
5. Perform Data Querying and Analysis on large datasets using Apache Hive.

Course Contents;

Unit 1: Understanding Big Data: Definition and significance; The role of big data in modern analytics: Characteristics of Big Data:Volume, Variety, Velocity, Veracity, and Value; Types of Data: Structured, unstructured, and semi-structured data; Introduction to time-series data and spatial data: Batch vs. Streaming Data, Key differences, use cases, and technologies; Sources of Big Data: Social media, IoT devices, enterprise systems, web logs, etc.; Real-world Examples of Big Data: Applications across industries (healthcare, finance, retail); Opportunities in Big Data: Career paths, industry demand, and emerging roles (data engineer, data scientist) **[6 hrs]**

Unit 2: Introduction to Hadoop: Overview and its role in big data processing; HDFS Overview: Architecture, block storage, and fault tolerance; Hadoop Installation: Single-node Hadoop installation on Ubuntu and multi-node cluster setup; Hadoop Architecture: Master and slave components (NameNode, DataNode, ResourceManager, NodeManager); HDFS Commands: Basic commands for data management and troubleshooting; Cluster Modes: Local, Pseudo-distributed, and Fully distributed modes; Hadoop Configuration and Versioning: Managing configurations, properties, and common challenges **[6 hrs]**

Unit 3: MapReduce Framework: Introduction to MapReduce**:** Overview and significance in big data processing Understanding MapReduce Workflow: Data flow, phases (Map, Shuffle, Reduce) MapReduce Components: Mapper, Reducer, Partitioner, and Combiner; Input Splits and HDFS Blocks: Relationship, importance, and optimization strategies; Developing MapReduce Programs:Programming with Eclipse and running in different modes; Debugging and Testing MapReduce Jobs: Common pitfalls and troubleshooting techniques; Advanced MapReduce Techniques: Using combiners and custom data types **[6 hrs]**

Unit 4: Data Ingestion with Apache Sqoop and Flume:
Introduction to Apache Sqoop: Purpose, functionality, and use cases; Sqoop Architecture: Data transfer process between Hadoop and RDBMS; Data Import Options: Importing data from various databases (MySQL, Oracle); Table and Binary Data Import; Techniques for different data types :Exporting Data with Sqoop; Sending data back to RDBMS and best practices; Introduction to Apache Flume: Overview and use cases for real-time data ingestion; Configuring and Using Flume Flume architecture and data flow. **[6 hrs]**

Unit 5: Data Analysis with Apache Hive and Pig

Introduction to Apache Hive: Overview and purpose in data warehousing; Hive Architecture: Components, services, and how it integrates with Hadoop; HiveQL Basics: Introduction to Hive Query Language and its syntax; Data Types and Storage Formats: Supported data types, file formats (ORC, Parquet), Managed vs. External Tables, Key differences, use cases, and data lifecycle;
Apache PIG: What is ETL, Introduction to Apache PIG, Execution Modes of PIG; Comparison of PIG with SQL and No-SQL Databases, PIG Data Types, Data Models in PIG, Grunt, PIG Latin, Overview of PIG User Defined Functions **[6 hrs]**

Unit 6: Introduction to Apache Spark and Machine Learning

Introduction to Apache Spark: Overview of Spark's purpose and key components; Understanding RDDs and DataFrames: Core concepts, transformations, and actions; Getting Started with Scala: Basics of Scala and its advantages for Spark; Spark SQL and Data Processing:Using Spark SQL for structured data analysis; Machine Learning with Spark MLlib: Introduction to MLlib, algorithms, and data pipelines; Graph Processing with GraphX: Overview and applications of GraphX in Spark;
Introduction to Real-time Data Processing; Overview of Spark Streaming and use cases. **[6 hrs]**

Reference Book:

1. Nathan Marz, James Warren, "Big Data: Principles and Best Practices of Scalable Real-Time Data Systems", Manning Publications, 1st edition, 2015, ISBN-10: 1617290343.
2. Tom White, "Hadoop: The Definitive Guide", O'Reilly Media, 4th edition, 2015, ISBN-10: 149190263X.
3. Apache Hadoop Documentation, Available online: Hadoop.apache.org.
4. Apache Spark Documentation, Available online: Spark.apache.org.
5. Data Ingestion with Apache Sqoop Documentation, Available online: Sqoop.apache.org.
6. Data Analysis with Apache Hive Documentation, Available online: Hive.apache.org.

(PGD-08) Deep Learning

Teaching Scheme
Lectures: 3Hrs/ Week

Evaluation Scheme
Theory: Mid Sem Exam: 30 Marks,
Assignment/Quizz: 20 Marks
ESE: 50 Marks

Course Outcomes

Students will be able to:

1. Understand Deep Learning Fundamentals: Demonstrate knowledge of deep learning concepts and their biological inspirations.
2. Design Neural Networks: Implement and train basic neural networks using TensorFlow and Keras.
3. Apply Optimization Techniques: Utilize gradient descent variants and regularization methods to improve model performance.
4. Build Advanced Neural Networks: Construct and train CNNs and RNNs for image and sequence tasks.
5. Implement Autoencoders: Use autoencoders and VAEs for data compression and anomaly detection.
6. Deploy Models Effectively: Deploy deep learning models as APIs and ensure scalability in real-world applications.

Unit 1: Introduction to Deep Learning

What is Deep Learning? Deep Learning vs. Machine Learning, Applications and Use Cases Biological Inspiration, Neurons in the Brain, Biological vs. Artificial Neurons, Overview of Artificial Neural Networks (ANNs) Layers: Input, Hidden, Output; Perceptron, Feedforward Neural Networks (FNN), Basic Terminology in Deep Learning: Weights, Biases, Neurons, Activation Functions Training a Neural Network, Forward Pass, Loss Functions (MSE, Cross-Entropy) Backpropagation and Gradient Descent; Hands-on: Build a basic neural network from scratch using a small dataset (like MNIST). **[6 hrs]**

Unit 2: Deep Learning with TensorFlow and Keras

Introduction to TensorFlow and Keras TensorFlow vs. Keras, Installing and Setting up the Environment Basic Neural Networks with Keras Sequential Model API, Defining Layers, Activation Functions Model Training and Evaluation Compiling the Model (Optimizers, Loss Functions, Metrics), Training, Validation, and Testing Saving and Loading Models Model Checkpoints, Saving Weights vs. Full Model Plotting and Monitoring Loss Curves, Accuracy Metrics, Using TensorBoard for Visualization Hands-on: Build and train a fully connected neural network for a classification problem using Keras. **[6 hrs]**

Unit 3: Deep Learning Model Optimization

Gradient Descent Variants Stochastic Gradient Descent (SGD), Mini-batch Gradient Descent, Adam, RMSprop, Adagrad Learning Rate and Optimization Techniques Learning Rate Schedules, Momentum, Nesterov Accelerated Gradient Regularization Techniques L1, L2 Regularization, Dropout, Early Stopping Hands-on: Apply regularization techniques (dropout, L2) and use Adam optimizer to improve model performance. **[6 hrs]**

Unit 4: Advanced Neural Networks

Understanding Convolutions Convolution Operation, Filters/Kernels, Feature Maps, Strides, Padding Pooling Layers Max Pooling, Average Pooling, Global Pooling CNN Architectures LeNet, AlexNet, VGG, ResNet Advanced Techniques in CNNs Batch Normalization, Dropout in CNNs, Fully Connected vs. Global Average Pooling (GAP) Transfer Learning with CNNs Using Pre-trained Models (VGG16, ResNet50), Fine-tuning vs. Feature Extraction RNN Architecture Unfolding RNNs, Vanishing Gradient Problem, Backpropagation Through Time (BPTT) Long Short-Term Memory (LSTM) Networks LSTM Cell Structure (Forget, Input, Output Gates), LSTMs vs. Traditional RNNs Gated Recurrent Units (GRUs) GRU Architecture and Comparison with LSTMs Applications of RNNs and LSTMs Time Series Forecasting, Text Generation Hands-on: Build a CNN from scratch for an image classification task (CIFAR-10) and apply transfer learning with a pre-trained network. Hands-on: Implement an LSTM for text generation or stock price prediction.

Unit 5: Autoencoders and Variational Autoencoders (VAEs)

Introduction to Autoencoders Encoder and Decoder Architecture, Bottleneck Layer and Data Compression Types of Autoencoders Sparse Autoencoders, Denoising Autoencoders, Contractive Autoencoders Variational Autoencoders (VAEs) Latent Space Representation, KL Divergence and Reconstruction Loss Applications of Autoencoders Image Reconstruction, Anomaly Detection, Data Denoising Hands-on: Implement a basic autoencoder for image reconstruction on the MNIST dataset. **[6 hrs]**

Unit 6: Model Deployment and Scalability

Deploying Deep Learning Models Model Serialization (SavedModel, H5 formats), Model Deployment on Cloud (AWS) Building APIs for Deep Learning Models REST API with Flask or FastAPI, Deploying as Web Services Handling Scalability Distributed Training, Model Parallelism, Using TensorFlow Serving and Kubernetes for Scalability Hands-on: Deploy a deep learning model as an API using Flask and test it in a real-time environment. **[6 hrs]**

Reference Book:

1. Aurélien Géron, “Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow”, O'Reilly Media, 2nd edition, October 2019, ISBN-10: 1492032646.
2. Charu C. Aggarwal, “Neural Networks and Deep Learning: A Textbook”, Springer, 2nd edition, March 2020, ISBN-10: 3030332512.
3. Christopher M. Bishop, “Pattern Recognition and Machine Learning”, Springer, 1st edition, April 2006, ISBN-10: 0387310738.

(PGD-09) Artificial Intelligence

Teaching Scheme
Lectures: 3Hrs/ Week

Evaluation Scheme
Theory: Mid Sem Exam: 30 Marks,
Assignment/Quizz: 20 Marks
ESE: 50 Marks

Course Outcomes:

Students will be able to:

1. Apply basic search techniques for problem solving.
2. Explain how to represent knowledge required for problem solving.
3. Apply reasoning to sift through data.
4. Utilize AI for applications in the real world.

Unit I: Introduction

Artificial Intelligence, AI Problems, AI Techniques, The Level of the Model, Criteria for Success; Defining the Problem as a State Space Search; Problem Characteristics; Production Systems; Search: Issues in the Design of Search Programs; Uninformed Search: BFS, DFS; Heuristic Search Techniques: Generate-and-Test, Hill Climbing, Best-First Search, A* Algorithm, Problem Reduction, AO* Algorithm, Constraint Satisfaction, Means-Ends Analysis. **[6 Hrs]**

Unit II: Knowledge Representation

Procedural vs. Declarative Knowledge; Representations & Approaches to Knowledge Representation; Forward vs. Backward Reasoning; Matching Techniques; Partial Matching; Fuzzy Matching Algorithms and RETE Matching Algorithms. **[6 Hrs]**

Unit III: Symbolic Logic

Propositional Logic; First Order Predicate Logic: Representing Instance and "isa" Relationships; Computable Functions and Predicates; Syntax & Semantics of FOPL; Normal Forms; Unification & Resolution; Representation Using Rules; Natural Deduction; Structured Representations of Knowledge: Semantic Nets, Partitioned Semantic Nets, Frames, Conceptual Dependency, Conceptual Graphs, Scripts, CYC. **[6 Hrs]**

Unit IV: Reasoning under Uncertainty

Introduction to Non-Monotonic Reasoning; Truth Maintenance Systems; Logics for Non-Monotonic Reasoning; Model and Temporal Logics; Statistical Reasoning: Bayes Theorem, Certainty Factors and Rule-Based Systems; Bayesian Probabilistic Inference; Bayesian Networks; Dempster-Shafer Theory; Fuzzy Logic: Crisp Sets, Fuzzy Sets, Fuzzy Logic Control, Fuzzy Inferences & Fuzzy Systems. **[6 Hrs]**

Unit V: Natural Language Processing

Role of Knowledge in Language Understanding; Approaches to Natural Language Understanding; Steps in Natural Language Processing; Syntactic Processing and Augmented Transition Nets; Semantic Analysis; NLP Understanding Systems; Planning: Components of a Planning System; Goal Stack Planning; Hierarchical Planning; Reactive Systems. **[6 Hrs]**

Unit VI: Machine Learning

Knowledge and Learning; Learning by Advice, Examples; Learning in Problem Solving; Symbol-Based Learning; Explanation-Based Learning; Version Space; ID3 Decision-Based Induction Algorithm; Unsupervised Learning; Reinforcement Learning; Supervised Learning: Perceptron Learning, Backpropagation Learning, Competitive Learning, Hebbian Learning. **[6 Hrs]**

Textbooks:

1. George F. Luger, "Artificial Intelligence", Pearson Education Publications.
2. Elaine Rich, Kevin Knight, "Artificial Intelligence", McGraw-Hill Publications.

References:

1. Patterson, "Introduction to Artificial Intelligence & Expert Systems", PHI.
2. Weiss, G., "Multi-Agent Systems: A Modern Approach to Distributed Artificial Intelligence", MIT Press.
3. Russell, S., Norvig, P., "Artificial Intelligence: A Modern Approach", Prentice Hall.

SEMESTER – III

(PGD-10) Computer Vision

Teaching Scheme
Lectures: 3Hrs/ Week

Evaluation Scheme
Theory: Mid Sem Exam: 30 Marks,
Assignment/Quizz: 20 Marks
ESE: 50 Marks

Course Outcomes:

Students will be able to:

1. Apply fundamental image processing techniques for feature extraction and object recognition.
2. Explain key concepts of computer vision, including image formation, edge detection, and segmentation.
3. Implement deep learning models for tasks like image classification, object detection, and face recognition.
4. Analyze real-world visual data using machine learning and computer vision algorithms.
5. Develop computer vision applications for domains such as healthcare, autonomous vehicles, and security.

Unit-1 Introduction:

Digital Image fundamentals, Image Sensing and acquisition, Sampling and Quantization, Image formation models, Overview of Computer Vision, Applications of Image processing and Computer Vision **[6 Hrs]**

Unit –II Image Enhancement:

Image enhancement in spatial domain, Basic grey level Transformations, Histogram Processing Techniques, Spatial Filtering, Image smoothing and Image Sharpening, Image enhancement process in frequency domain, Low pass filtering, High pass filtering **[6 Hrs]**

Unit-III Image Segmentation:

Thresholding, Regions Based segmentation, K-means clustering, **Image Filters**: Smoothing, Sharpening, and Edge Detection, Edge linking and boundary detection, Masking. **[6 Hrs]**

Unit- IV Feature Extraction:

Importance of Features, Feature extraction techniques, Histogram of Oriented Gradient (HOG), Scale Invariant Feature Transform (SIFT), Background subtraction, techniques, Image Matching, Principal Component Analysis (PCA)

Unit V Object Recognition and Motion Estimation:

Object Recognition techniques: Viola-Jones, Yolo, Deep learning algorithms for Object Recognition. Optical Flow, Gaussian Mixture Model (GMM), Structure of Motion, Motion Estimation. **[6 Hrs]**

Unit VI Applications of Image Processing and Computer vision:

Face Recognition, Facial Expression Recognition, Image Notation Tools and Advanced Libraries

Introduction to Image Notation and Annotation, CVAT (Computer Vision Annotation Tool): Advanced Annotation Tool for Object Detection **[6 Hrs]**

References:

- Digital Image Processing- Refael C. Gonzalez and Richard E. Woods, Wesley
- Computer Vision - A modern approach, by D. Forsyth and J. Ponce, Prentice Hall Robot Vision, by B. K. P. Horn, McGraw-Hill.
- Introductory Techniques for 3D Computer Vision, by E. Trucco and A. Verri, Publisher: Prentice Hall.
- Image Processing: The Fundamentals *by O'Neill*

(PGD-11) Generative AI

Teaching Scheme
Lectures: 3Hrs/ Week

Evaluation Scheme
Theory: Mid Sem Exam: 30 Marks,
Assignment/Quizz: 20 Marks
ESE: 50 Marks

Course Outcomes:

Students will be able to:

1. Understand the fundamentals of Generative AI and Large Language Models (LLMs), including their evolution and applications.
2. Gain hands-on experience in working with tokenization, embeddings, and pre-trained language models for various NLP tasks.
3. Develop skills in fine-tuning and optimizing LLMs for text classification, clustering, topic modelling, and text generation.
4. Explore advanced AI techniques such as reinforcement learning with human feedback (RLHF) and retrieval-augmented generation (RAG).
5. Analyse ethical considerations, bias mitigation strategies, and responsible AI practices for deploying generative AI systems.

Unit 1: Introduce on to Generative AI : An Introduce on to Large Language Models (LLMs) - Evolution of LLMs: From GPT-2 to GPT-4 and beyond - Key characteristics of LLMs - Applications and limitations of generative AI; Tokens and Embeddings - Tokenization: Understanding how text is processed into tokens - Embeddings: Concept, usage, and visualization - Embedding techniques: Word2Vec, GloVe, FastText, and contextual embeddings like BERT; Looking Inside LLMs - Architecture of LLMs: Transformers, attention mechanisms, and self-attention - Understanding the model layers and parameters - Analysis of inference, performance, and scalability.

Unit 2: Using Pre-trained Language Models: Text Classification on - Techniques for sentiment analysis, spam detection, and intent recognition - Pre-trained models for classification tasks - Case study: Building a sentiment analysis tool; Text Clustering and Topic Modelling - Text similarity measures and clustering algorithms - Latent Dirichlet Allocation (LDA) for topic modelling - Applications of clustering in real-world datasets; Prompt Engineering - Crafting effective prompts for generative tasks - Few-shot and zero-shot learning with LLMs - Prompt optimization and experimentation 7

Unit 3: Advanced Text Genera on Techniques and Tools - Techniques for controlled text generation - Tools for generating high-quality text using LLMs - Ethical considerations in generative text outputs; Semantic Search and Retrieval-Augmented Genera on - Understanding semantic search with embedding - Combining retrieval models with generative AI (RAG) - Building knowledge-enhanced chatbots and applications

Unit 4: Multimodal LLMs - Combining text with other modalities: vision, speech, and more - Exploring multimodal models like CLIP, DALL·E, and Flamingo - Applications of multimodal AI in

creative and practical tasks; Fine-Tuning Genera on Models - Techniques for adapting GPT-like models for specific tasks - Dataset preparation for fine-tuning generative tasks

Unit 5 Training and Fine-Tuning Language Models: Creating Text Embedding Models - Designing custom embedding models - Optimizing embedding for semantic understanding - Deployment of embedding-based search systems; Fine-Tuning Represent on Models for Classification - Fine-tuning pre-trained models like BERT, RoBERTa, and DistilBERT - Applications in domain-specific tasks - Evaluating and optimizing fine-tuned models.

Unit 6 Use cases: Domain-specific text generation, summarization, and rewriting : Advanced Topics in Generative AI; Reinforcement Learning with Human Feedback (RLHF) - Understanding RLHF in generative AI - Training with feedback loops for improved outputs - Applications in conversational AI; Ethics and Bias in Generative AI - Identifying and mitigating biases in language models - Building responsible and ethical generative AI applications - Governance and regulatory compliance ; Scaling and Deploying LLMs - Infrastructure requirements for large-scale LLMs - Efficient inference techniques: quantization and distillation - Deployment strategies and performance monitoring; Future Directions in Generative AI - Emerging trends in generative AI - Open research challenges in LLMs and multimodal AI - Preparing for innovations in AI technology.

References:

1. OpenAI (2023). "GPT-4 Technical Report." Available at OpenAI website.
2. Jurafsky, D., & Martin, J. H. (2021). "Speech and Language Processing." Pearson.
3. Mitchell, M. (2019). "Artificial Intelligence: A Guide for Thinking Humans." Farrar, Straus and Giroux.
4. Russell, S., & Norvig, P. (2021). "Artificial Intelligence: A Modern Approach." Pearson.