AIDS Track

Fundamentals of Artificial Intelligence

BCO 541A	**Fundamentals of Artificial Intelligence**	**3-0-0 [3]**

OBJECTIVE:

- To provide fundamental knowledge of Artificial Intelligence concepts, goals, scope, and evaluation methods.
- To understand core AI techniques such as knowledge representation, search strategies, and heuristic problem-solving.
- To develop familiarity with logic-based reasoning approaches, including propositional logic, first-order logic, and inference mechanisms.
- To explore the design and applications of Expert Systems along with various reasoning methods like deductive, inductive, abductive, and fuzzy reasoning.
- To equip students with foundational skills in machine learning paradigms, neural networks, and deep learning techniques for solving real-world problems.

UNIT 1	Artificial Intelligence Introduction, Goals, Reasons of Boost, Applications of Artificial Intelligence in various domains, Composition of Artificial Intelligence, Advantages, Disadvantages of Artificial Intelligence, Classification of Artificial Intelligence, Weak, Evolutionary, Strong, Artificial Intelligence Agents, Types of AI Agents, PEAS in Artificial Intelligence, Classification of Environment in Artificial Intelligence, Turing Test in Artificial Intelligence with Configuration and Steps to Perform
UNIT 2	Artificial Intelligence Technique, Knowledge Representation, Search Algorithm, Breadth First Search(BFS), Depth First Search(DFS), Depth Limited Search, Uniform Cost Search, Bidirectional Search, Iterative Deepening Depth First Search, Informed Search, Heuristic Function, Heuristic Search in Artificial Intelligence, Admissible & Non-Admissible with Examples, Difference(Comparison) between Blind Search and Heuristic Search in Artificial Intelligence, Best First Search(BFS), Informed Search , A Star(A*) Search Algorithm, AO Star(AO*) Search Algorithm
UNIT 3	Knowledge in AI, Knowledge Based Agent, Knowledge Representation, Knowledge Representation Techniques, Propositional Logic, Rules of Inference, First-order Logic, Inference Rules in First Order Logic, Knowledge Engineering in FOL, Unification in First Order Logic (FOL), Resolution in First Order Logic (FOL), Forward Chaining and Backward Chaining, Backward Chaining vs Forward Chaining
UNIT 4	Expert Systems, Applications of Expert Systems, Advantages & Limitations of Expert Systems, Reasoning in Artificial Intelligence, Types of Reasoning in AI, Deductive Reasoning, Inductive Reasoning, Abductive Reasoning, Fuzzy Reasoning
UNIT 5	Types of Learning in AI, Supervised Learning, Linear Regression, Logistic Regression, Decision Trees, Semi-supervised Learning, Unsupervised Learning, Reinforcement Learning, Deep Learning, Neurons, Single Layer Perceptron, Multi-Layer Perceptron, Artificial Neural Networks (ANNs)

Course Outcome*:*

CO1: Understand the fundamental concepts, scope, and applications of Artificial Intelligence.

CO2: Apply uninformed and informed search strategies to solve AI-related problems.
CO3: Utilize propositional logic, first-order logic, and inference techniques for reasoning and knowledge representation.
CO4: Analyze and design Expert Systems using different reasoning methods such as deductive, inductive, abductive, and fuzzy reasoning.
CO5: Demonstrate the application of machine learning paradigms and neural network architectures for real-world problem solving.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	**Program Outcome**												**Program Specific Outcome**		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H								M		H		M		
CO2	H	H	M							H				H	L
CO3	M	H	H		M					H				M	M
CO4	H	M		M						H	M			H	
CO5	M		H	H						M	H			M	M

H = Highly Related; M = Medium L = Low

Textbook:

- **Stuart Russell & Peter Norvig**, *Artificial Intelligence: A Modern Approach*, 4th Edition, Pearson, 2021.
- **Elaine Rich, Kevin Knight, and Shivashankar B. Nair**, *Artificial Intelligence*, 3rd Edition, McGraw Hill, 2009.
- **Nils J. Nilsson**, *Principles of Artificial Intelligence*, Narosa Publishing House, 1980.

Reference Books:

- **George F. Luger**, *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*, 6th Edition, Pearson, 2008.
- **Patrick Henry Winston**, *Artificial Intelligence*, 3rd Edition, Addison Wesley, 1992.
- **Dan W. Patterson**, *Introduction to Artificial Intelligence and Expert Systems*, Pearson Education, 2007.

DMA022A	Mathematics for AI	3-1-0 [4]

OBJECTIVE:

- To introduce the fundamental concepts of linear algebra, probability, and calculus essential for Artificial Intelligence applications.
- To develop the ability to model and analyze data mathematically for AI-based problem-solving.
- To familiarize students with optimization and numerical methods used in AI and machine learning..
- To build a strong mathematical foundation for advanced AI, data science, and machine learning courses in higher semesters.

UNIT 1	Linear Algebra for AI: Vectors, vector spaces, and subspaces; Linear independence; Basis and dimension; Matrices – types, operations, transpose, inverse; Rank of a matrix; Systems of linear equations – Gaussian elimination; Eigenvalues and eigenvectors; Applications of linear algebra in AI (image processing, data representation).
UNIT 2	Probability and Statistics: Basics of probability theory; Conditional probability and Bayes' theorem; Random variables – discrete and continuous; Probability distributions – Bernoulli, Binomial, Gaussian; Mean, variance, standard deviation; Introduction to sampling; Law of large numbers; Applications in AI (spam detection, prediction models).
UNIT 3	Calculus for AI: Functions, limits, and continuity; Differentiation – rules and applications; Partial derivatives; Gradient and directional derivatives; Optimization basics – maxima, minima; Chain rule; Basics of integration; Applications of calculus in AI (gradient descent, curve fitting).
UNIT 4	Vectors, Matrices, and Transformations: Geometric interpretation of vectors; Dot and cross product; Norms and distances; Orthogonality; Projections; Matrix factorizations – LU, QR (conceptual); Transformations and their role in AI (rotation, scaling in data and images).
UNIT 5	Introduction to Numerical Methods and Optimization: Numerical solution of equations – bisection, Newton-Raphson method; Basics of numerical linear algebra; Optimization techniques – unconstrained and constrained optimization; Role of optimization in AI (model parameter tuning, cost function minimization).

Course Outcome*:*

CO1 – Apply linear algebra concepts to represent and manipulate data for AI applications.
CO2 – Use probability and statistics to model uncertainty and make predictions in AI systems.
CO3 – Apply calculus concepts for optimization and learning algorithms in AI.
CO4 – Interpret vector and matrix transformations in the context of AI and data analysis.
CO5 – Relate numerical methods and optimization techniques to real-world AI problem-solving.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	**Program Outcome**												**Program Specific Outcome**		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	L	L									M		L
CO2	M	M		L	L								H	M	L
CO3	H		M			L							M		L
CO4	H	M	L										M	L	
CO5	M	M	H	L									H	M	L

H = Highly Related; M = Medium L = Low

Textbook:

- **Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong, Mathematics for Machine Learning, Cambridge University Press.**
- **Gilbert Strang, Introduction to Linear Algebra, Wellesley Cambridge Press.**
- **Sheldon M. Ross, Introduction to Probability Models, Academic Press.**
- **Erwin Kreyszig, Advanced Engineering Mathematics, Wiley.**
- **E. Alpaydin, Introduction to Machine Learning, MIT Press.**

Introduction to Data Science- Core

BCO606A	Introduction to Data Science	3-0-0 [3]

OBJECTIVE:

- Understand the foundations of Data Science including its scope, importance, applications.
- Develop skills in data collection and pre-processing
- Perform exploratory data analysis (EDA), applying descriptive statistics, data visualization, and dimensionality reduction methods for data interpretation.
- Gain introductory knowledge of Machine Learning, understanding basic models, evaluation metrics, and challenges such as overfitting and underfitting.
- Explore applications, ethics, and future trends in Data Science including big data, cloud, deep learning, and responsible AI practices.

UNIT 1	**Foundations of Data Science:** Definition, scope, importance of Data Science, role of Data Scientist, skills required, Data Science life cycle, data types (structured, semi-structured, unstructured), applications of Data Science (healthcare, finance, social media, etc.), overview of tools & technologies (Python, R, SQL, Jupyter, Git)
UNIT 2	**Data Collection & Preprocessing:** Data acquisition (databases, APIs, web scraping, sensors, logs), handling missing values, handling outliers, handling duplicates, data cleaning, data preprocessing steps, feature engineering, feature scaling (normalization, standardization), data transformation, encoding techniques (categorical, ordinal, one-hot, label encoding), introduction to preprocessing pipelines.
UNIT 3	**Exploratory Data Analysis (EDA):** Descriptive statistics (mean, median, mode, variance, skewness, kurtosis), data visualization (histograms, scatter plots, box plots, heatmaps), correlation, covariance, dimensionality reduction (PCA – introduction), case study on real-world dataset (exploratory analysis)
UNIT 4	**Introduction to Machine Learning:** Machine Learning vs Traditional Programming, types of learning (supervised, unsupervised, reinforcement – overview), regression (linear regression, logistic regression), classification (k-NN, decision trees), clustering (k-means), model evaluation (accuracy, precision, recall, F1-score, confusion matrix), train-test split, cross-validation, overfitting, underfitting
UNIT 5	**Applications, Ethics & Future of Data Science :**Case studies in AI & Data Science, Big Data in Data Science, Cloud in Data Science, introduction to Deep Learning (basic concepts), data privacy, data security, ethical considerations in Data Science, emerging trends (AutoML, Explainable AI, Timeseries forecasting,Generative AI ,Agentic AI in Data Science),

Course Outcome*:*

CO1. Define and explain the concepts, scope, and applications of Data Science, and demonstrate familiarity with essential tools like Python, R, SQL, and Jupyter.

CO2. Collect, clean, and preprocess datasets from multiple sources (databases, APIs, web scraping, sensors, logs) and apply feature engineering, scaling, and encoding techniques.

CO3. Apply EDA techniques using descriptive statistics, data visualization, and dimensionality reduction to extract insights from real-world datasets.

CO4. Implement basic Machine Learning models (regression, classification, clustering), evaluate performance with appropriate metrics, and address model validation issues.

CO5. Analyse case studies and future trends in Data Science, critically assess ethical implications, and discuss emerging technologies such as Explainable AI, Generative AI, and Agentic AI.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	**Program Outcome**												**Program Specific Outcome**		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	L	L	M	L	-	L	-	M	-	M	H	M	M
CO2	H	H	M	M	H	-	-	-	-	M	-	M	H	H	M
CO3	H	H	M	H	H	-	-	-	-	M	-	M	H	H	M
CO4	H	H	H	H	H	-	-	-	-	M	L	M	H	H	M
CO5	M	M	L	M	M	M	M	H	M	M	L	H	M	M	H

H = Highly Related; M = Medium L = Low

Textbooks

1. Anil Maheshwari* – Data Science: Fundamentals and Practical Approaches, Springer, 2017.
2. Foster Provost & Tom Fawcett* – Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking, O'Reilly, 2013.
3. Joel Grus* – Data Science from Scratch: First Principles with Python, O'Reilly, 2019.
4. Jake VanderPlas* – Python Data Science Handbook: Essential Tools for Working with Data, O'Reilly, 2016.
5. Trevor Hastie, Robert Tibshirani, Jerome Friedman* – The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer, 2nd Edition, 2009.

Supplementary References*

1. Andreas Müller & Sarah Guido* – Introduction to Machine Learning with Python, O'Reilly, 2017.
2. Christopher Bishop* – Pattern Recognition and Machine Learning, Springer, 2006.
3. Ian Goodfellow, Yoshua Bengio, Aaron Courville* – Deep Learning, MIT Press, 2016.

Online Resources

- Python libraries: [NumPy](https://numpy.org/), [Pandas](https://pandas.pydata.org/), [Matplotlib](https://matplotlib.org/), [Seaborn](https://seaborn.pydata.org/)
- Datasets: [Kaggle](https://www.kaggle.com/), [UCI Machine Learning Repository](https://archive.ics.uci.edu/ml/index.php)
- MIT OpenCourseWare: Introduction to Computational Thinking and Data

MACHINE LEARNING

BCO636A	Machine Learning	3-0-0 [3]

OBJECTIVE:

- To understand the fundamental concepts, types, and applications of Machine Learning.
- To develop the ability to preprocess data and select appropriate features for building ML models.
- To gain proficiency in implementing supervised and unsupervised learning algorithms.
- To explore advanced techniques such as ensemble methods, neural networks, and deep learning basics.
- To apply Machine Learning models to real-world problems and evaluate their performance using standard metrics.

UNIT 1	Introduction to Machine Learning: Definition of Machine Learning, Types of Machine Learning: Supervised, Unsupervised, Reinforcement, Applications of Machine Learning, Data preprocessing: cleaning, normalization, feature selection, Overview of ML workflow, Model evaluation metrics
UNIT 2	Supervised Learning: Regression techniques: Linear Regression, Polynomial Regression, Classification techniques: Logistic Regression, k-Nearest Neighbors (k-NN), Decision Trees, Support Vector Machines (SVM), Performance evaluation: Confusion matrix, Accuracy, Precision, Recall, F1-score, ROC curve
UNIT 3	Unsupervised Learning: Clustering techniques: k-Means, Hierarchical, DBSCAN, Dimensionality reduction: Principal Component Analysis (PCA), t-SNE, Association rule learning: Apriori algorithm, FP-Growth, Applications of unsupervised learning
UNIT 4	Advanced Machine Learning Techniques: Ensemble methods: Bagging, Boosting, Random Forests, Gradient Boosting, Neural networks basics: Perceptron, Multi-Layer Perceptron (MLP), Activation functions, Introduction to Deep Learning concepts, Overfitting, underfitting, Regularization techniques
UNIT 5	Model Deployment, Tools, and Applications: Model selection, Cross-validation techniques, Popular ML tools and libraries: Python, scikit-learn, TensorFlow, Keras, Real-world applications: Image recognition, Natural Language Processing (NLP), Recommendation systems, Autonomous systems, Case studies and project discussions

Course Outcome*:*

CO1: Understand the fundamental concepts, types, and workflow of Machine Learning.

CO2: Apply data preprocessing techniques and perform feature selection for ML models.

CO3: Implement and analyze supervised and unsupervised learning algorithms for solving real-world problems.

CO4: Explore and utilize advanced ML techniques, including ensemble methods, neural networks, and deep learning basics.

CO5: Evaluate and deploy ML models using performance metrics and popular ML tools for practical applications.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	**Program Outcome**												**Program Specific Outcome**		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H			M		H		M							
CO2	H	H	M						H				H	L	
CO3	M	H	H		M				H	M			M	M	
CO4	H	M		M					H		M	L	H		
CO5	M		H	H					M		H	M	M		

H = Highly Related; M = Medium L = Low

Textbooks

1. **“Machine Learning”** by Tom M. Mitchell, McGraw-Hill, 1997.
2. **“Pattern Recognition and Machine Learning”** by Christopher M. Bishop, Springer, 2006.
3. **“Introduction to Machine Learning with Python”** by Andreas C. Müller and Sarah Guido, O’Reilly Media, 2016.
4. **“Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow”** by Aurélien Géron, 2nd Edition, O’Reilly Media, 2019.
5. **“Machine Learning: A Probabilistic Perspective”** by Kevin P. Murphy, MIT Press, 2012.

Reference Books

1. **“Deep Learning”** by Ian Goodfellow, Yoshua Bengio, and Aaron Courville, MIT Press, 2016.
2. **“Data Mining: Concepts and Techniques”** by Jiawei Han, Micheline Kamber, and Jian Pei, 3rd Edition, Morgan Kaufmann, 2011.
3. **“Python Machine Learning”** by Sebastian Raschka and Vahid Mirjalili, Packt Publishing, 3rd Edition, 2019.
4. **“Machine Learning for Dummies”** by John Paul Mueller and Luca Massaron, Wiley, 2016.
5. **“Artificial Intelligence: A Modern Approach”** by Stuart Russell and Peter Norvig, 4th Edition, Pearson, 2020.

DEEP LEARNING

BCO638A	Deep Learning	3-0-0 [3]

OBJECTIVE:

- To understand the fundamental concepts and architecture of deep neural networks.
- To develop skills in implementing and training artificial neural networks, CNNs, and RNNs.
- To explore advanced deep learning techniques such as autoencoders, GANs, and deep reinforcement learning.
- To apply deep learning frameworks like TensorFlow, Keras, and PyTorch for practical problem solving.
- To analyze and evaluate deep learning models for real-world applications in computer vision, NLP, and AI systems.

UNIT 1	Introduction to Deep Learning: Introduction to Artificial Neural Networks (ANN), Differences between Machine Learning and Deep Learning, History and evolution of Deep Learning, Applications of Deep Learning, Overview of Deep Learning workflow, Activation functions and their importance
UNIT 2	Neural Networks Fundamentals: Perceptron, Multi-Layer Perceptron (MLP), Forward and Backpropagation algorithms, Gradient Descent, Loss functions, Overfitting, underfitting, Regularization techniques (L1, L2, Dropout), Optimization algorithms: SGD, Adam
UNIT 3	Convolutional Neural Networks (CNN): Introduction to CNN, Convolution operation, Pooling layers, CNN architectures (LeNet, AlexNet, VGG, ResNet), Applications in image recognition and computer vision, Transfer learning using pre-trained CNN models
UNIT 4	Recurrent Neural Networks (RNN) and Variants: Introduction to RNN, Vanishing and Exploding gradient problem, Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Applications in time series prediction and Natural Language Processing (NLP), Sequence modeling
UNIT 5	Advanced Deep Learning Concepts and Tools: Autoencoders, Generative Adversarial Networks (GANs), Deep Reinforcement Learning, Introduction to frameworks: TensorFlow, Keras, PyTorch, Model evaluation and deployment, Case studies and real-world applications

Course Outcome*:*

CO1: Understand the concepts, architecture, and workflow of deep learning models.
CO2: Implement and train artificial neural networks using forward and backpropagation.
CO3: Design and apply convolutional neural networks (CNNs) for image and vision-based tasks.
CO4: Develop recurrent neural networks (RNNs), LSTM, and GRU models for sequence and NLP tasks.
CO5: Explore advanced deep learning models (autoencoders, GANs, reinforcement learning) and deploy models using modern DL frameworks.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H		M		M								H		
CO2	H	M	H		M								H	M	
CO3	H	H	H	M	H								M	H	
CO4	M	H	H		H								M	H	
CO5	H	M	H	M	H								H	H	M

H = Highly Related; M = Medium L = Low

Textbooks

1. **“Deep Learning”** by Ian Goodfellow, Yoshua Bengio, and Aaron Courville, MIT Press, 2016.
2. **“Hands-On Deep Learning with TensorFlow”** by Dan Van Boxel, Packt Publishing, 2018.
3. **“Deep Learning with Python”** by François Chollet, 2nd Edition, Manning Publications, 2021.
4. **“Python Deep Learning”** by Ivan Vasilev, Packt Publishing, 2017.
5. **“Neural Networks and Deep Learning”** by Charu C. Aggarwal, Springer, 2018.

Reference Books

1. **“Deep Learning for Computer Vision”** by Rajalingappaa Shanmugamani, Packt Publishing, 2018.
2. **“Introduction to Deep Learning Using Python”** by Ahmed Menshawy, Apress, 2019.
3. **“Machine Learning and Deep Learning for Engineers”** by Anil Maheshwari, Wiley, 2020.
4. **“Practical Deep Learning for Cloud, Mobile, and Edge”** by Anirudh Koul, O’Reilly Media, 2019.
5. **“Artificial Intelligence: A Modern Approach”** by Stuart Russell and Peter Norvig, 4th Edition, Pearson, 2020.

LARGE LANGUAGE MODELLING

BCO616A	Large Language Modelling	3-0-0 [3]

OBJECTIVE:

- To understand the fundamental concepts and evolution of Large Language Models (LLMs).
- To develop skills in Transformer architectures and attention mechanisms.
- To explore pre-training, fine-tuning, and prompt engineering techniques for LLMs.
- To study optimization, scaling, and evaluation techniques for large models.
- To apply LLMs in practical applications such as text generation, summarization, and conversational AI using modern frameworks.

UNIT 1	Introduction to Large Language Models: Overview of NLP and Language Models, Evolution from traditional models to Transformers, Introduction to Large Language Models (LLMs), Applications of LLMs in AI systems, Challenges in LLMs, Ethical and societal considerations, Tokenization methods (BPE, WordPiece, SentencePiece)
UNIT 2	Transformer Architecture: Self-attention mechanism, Multi-head attention, Positional encoding, Encoder-decoder structure, Feed-forward networks in Transformers, Layer normalization and residual connections, Comparison with RNNs and LSTMs
UNIT 3	Pre-training and Fine-tuning of LLMs: Pre-training objectives: Masked Language Modeling (MLM), Causal Language Modeling (CLM), Next Sentence Prediction, Fine-tuning on downstream tasks, Transfer learning in LLMs, Prompt engineering and prompt tuning, Instruction-tuned models
UNIT 4	Advanced LLM Techniques and Optimization: Handling long sequences (efficient attention mechanisms), Model scaling and parallelism, Knowledge distillation and model compression, Sparse vs dense models, RLHF (Reinforcement Learning from Human Feedback), Evaluation metrics for LLMs (perplexity, accuracy, BLEU, ROUGE)
UNIT 5	Applications, Tools, and Frameworks: Text generation, Summarization, Question answering, Chatbots and conversational AI, Code generation and reasoning, LLM frameworks and libraries: Hugging Face Transformers, OpenAI API, DeepSpeed, Accelerate, Case studies of GPT, BERT, LLaMA, BLOOM models

Course Outcome*:*

CO1: Understand the architecture, principles, and evolution of Large Language Models.
CO2: Analyze and implement Transformer-based models and attention mechanisms.
CO3: Apply pre-training, fine-tuning, and prompt engineering techniques for downstream NLP tasks.
CO4: Explore advanced LLM optimization techniques, scaling strategies, and model evaluation methods.
CO5: Develop and deploy LLM-based applications using popular frameworks such as Hugging Face Transformers, DeepSpeed, and OpenAI APIs.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	**Program Outcome**												**Program Specific Outcome**		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H		M		M								H		
CO2	H	M	H		H								H	M	
CO3	H	H	H	M	H								M	H	
CO4	M	H	H		H								M	H	
CO5	H	M	H	M	H								H	H	M

H = Highly Related; M = Medium L = Low

Textbooks

1. **"Natural Language Processing with Transformers"** by Lewis Tunstall, Leandro von Werra, Thomas Wolf, O'Reilly Media, 2022.
2. **"Transformers for Natural Language Processing"** by Denis Rothman, Packt Publishing, 2021.
3. **"Deep Learning for NLP"** by Palash Goyal, Sumit Pandey, Karan Jain, Apress, 2018.
4. **"Neural Network Methods in Natural Language Processing"** by Yoav Goldberg, Morgan & Claypool Publishers, 2017.
5. **"GPT-3: Building Innovative NLP Products Using Large Language Models"** by Sandra Kublik, Packt Publishing, 2023.

Reference Books

1. **"Applied Natural Language Processing with Python"** by Taweh Beysolow II, Apress, 2018.
2. **"Hands-On Transformers with Python"** by Sudharsan Ravichandiran, Packt Publishing, 2021.
3. **"Deep Learning for Natural Language Processing"** by Palash Goyal, Sumit Pandey, Karan Jain, Apress, 2018.
4. **"Generative Deep Learning"** by David Foster, O'Reilly Media, 2019.
5. **OpenAI, Hugging Face, and DeepSpeed Documentation and Tutorials (Online Resources)**

NATURAL LANGUAGE PROCESSING

BCO640A	Natural Language Processing	3-0-0 [3]

OBJECTIVE:

- To understand the fundamental concepts, techniques, and applications of Natural Language Processing.
- To develop skills in text preprocessing, language modeling, and syntactic analysis.
- To explore semantic analysis, embeddings, and advanced NLP models using neural networks.
- To implement and evaluate NLP applications such as machine translation, sentiment analysis, and chatbots.
- To gain hands-on experience with popular NLP tools and frameworks like NLTK, SpaCy, and Hugging Face Transformers.

UNIT 1	Introduction to NLP: Definition and scope of NLP, History and evolution of NLP, Applications of NLP, Overview of NLP pipeline, Challenges in NLP, Regular expressions, Text preprocessing: tokenization, stemming, lemmatization, stopword removal, POS tagging
UNIT 2	Language Modeling and Syntax: N-gram language models, Probability and smoothing techniques, Part-of-Speech (POS) tagging methods, Context-free grammar (CFG), Parsing techniques: Top-down, Bottom-up, Dependency parsing, Syntax trees, Applications of syntactic analysis
UNIT 3	Semantic Analysis: Word sense disambiguation, Semantic similarity and relatedness, Named Entity Recognition (NER), Semantic role labeling, Lexical semantics: WordNet, Distributional semantics, Introduction to embeddings: Word2Vec, GloVe, FastText
UNIT 4	Advanced NLP Techniques: Machine learning approaches in NLP: Naive Bayes, HMM, CRF, Introduction to neural network-based NLP, Recurrent Neural Networks (RNNs), LSTM and GRU for sequence modeling, Attention mechanism, Transformer architecture, BERT, GPT models
UNIT 5	Applications and Tools in NLP: Text classification, Sentiment analysis, Machine translation, Question answering systems, Chatbots, Speech recognition, NLP libraries and frameworks: NLTK, SpaCy, Hugging Face Transformers, Evaluation metrics: BLEU, ROUGE, Case studies and practical applications

Course Outcome*:*

CO1: Understand the core concepts, techniques, and challenges in Natural Language Processing.
CO2: Perform text preprocessing, tokenization, stemming, lemmatization, POS tagging, and syntactic analysis.
CO3: Apply semantic analysis, word embeddings, and word sense disambiguation in NLP applications.
CO4: Implement advanced NLP models using neural networks, transformers, and attention mechanisms.
CO5: Develop, evaluate, and deploy NLP applications using modern frameworks and tools.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	**Program Outcome**												**Program Specific Outcome**		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H		M		M								H		
CO2	H	M	H		M								H	M	
CO3	H	H	H	M	H								M	H	
CO4	M	H	H		H								M	H	
CO5	H	M	H	M	H								H	H	M

H = Highly Related; M = Medium L = Low

Textbooks

1. **"Speech and Language Processing"** by Daniel Jurafsky and James H. Martin, 3rd Edition, Pearson, 2021.
   "Natural Language Processing with Python" by Steven Bird, Ewan Klein, and Edward Loper, O'Reilly Media, 2009.
2. **"Deep Learning for Natural Language Processing"** by Palash Goyal, Sumit Pandey, and Karan Jain, Apress, 2018.
3. **"Foundations of Statistical Natural Language Processing"** by Christopher D. Manning and Hinrich Schütze, MIT Press, 1999.
4. **"Natural Language Processing in Action"** by Hobson Lane, Cole Howard, Hannes Hapke, Manning Publications, 2019.

Reference Books

1. **"Python Natural Language Processing"** by Jalaj Thanaki, Packt Publishing, 2017.
2. **"Practical Natural Language Processing"** by Sowmya Vajjala, Bodhisattwa Majumder, Anuj Gupta, Harshit Surana, O'Reilly Media, 2020.
3. **"Transformers for Natural Language Processing"** by Denis Rothman, Packt Publishing, 2021.
4. **"Neural Network Methods in Natural Language Processing"** by Yoav Goldberg, Morgan & Claypool Publishers, 2017.
5. **"Applied Natural Language Processing"** by Taweh Beysolow II, Apress, 2018.

COMPUTER VISION AND IMAGE PROCESSING

BCO671A	Computer Vision and Image Processing	3-0-0 [3]

OBJECTIVE:

- To understand the fundamentals of digital image representation, acquisition, and processing.
- To learn techniques for image enhancement, restoration, segmentation, and feature extraction.
- To study object detection, recognition, and tracking methods using classical and deep learning approaches.
- To explore advanced applications of computer vision such as face recognition, motion analysis, and 3D vision.
- To gain practical experience with popular CV and IP tools and frameworks (OpenCV, TensorFlow, PyTorch).

UNIT 1	Introduction to Computer Vision and Image Processing: Overview of computer vision and image processing, Applications in real-world systems, Digital image fundamentals, Image formation and acquisition, Image representation and models, Image sampling and quantization, Basic operations on images (point, neighborhood, geometric transformations).
UNIT 2	Image Enhancement and Restoration: Image enhancement in spatial domain: contrast stretching, histogram equalization, smoothing and sharpening filters, Edge detection techniques (Sobel, Prewitt, Canny), Noise models and types, Image restoration: inverse filtering, Wiener filtering, Regularization techniques.
UNIT 3	Image Segmentation and Feature Extraction: Thresholding techniques, Region growing and splitting, Watershed segmentation, Clustering-based segmentation (k-means, mean-shift), Morphological image processing, Feature extraction: corners, blobs, SIFT, SURF, HOG descriptors.
UNIT 4	Object Detection, Recognition, and Tracking: Template matching, Bag of Visual Words (BoVW), Object detection using traditional methods (Haar cascades, HOG+SVM), Introduction to deep learning in vision (CNNs for vision tasks), Modern object detection models (R-CNN, Fast R-CNN, YOLO, SSD), Object tracking methods (Kalman filter, Mean-shift, CAMShift).
UNIT 5	Advanced Topics and Applications: Face detection and recognition, Motion analysis, Image compression techniques (JPEG, MPEG), 3D vision basics (stereo vision, depth estimation), Medical image analysis, Computer vision in autonomous systems, Popular libraries and frameworks (OpenCV, TensorFlow, PyTorch, Keras), Case studies and project discussions.

Course Outcome:

CO1: Understand the fundamental concepts of computer vision and digital image processing.
CO2: Apply image enhancement, filtering, and restoration techniques to improve image quality.
CO3: Implement image segmentation and feature extraction methods for vision applications.
CO4: Design object detection, recognition, and tracking solutions using traditional and deep learning approaches.
CO5: Explore advanced computer vision applications and use modern tools/frameworks for real-world problem solving.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	**Program Outcome**												**Program Specific Outcome**		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M			M								H		
CO2	H	H	M	M	H								H	M	
CO3	M	H	H	H	H								M	H	
CO4	M	H	H	H	H								H	H	M
CO5	H	M	H	H	H		M						H	H	M

H = Highly Related; M = Medium L = Low

Textbooks

1. **“Digital Image Processing”** by Rafael C. Gonzalez and Richard E. Woods, 4th Edition, Pearson, 2018.
2. **“Computer Vision: Algorithms and Applications”** by Richard Szeliski, Springer, 2022.
3. **“Learning OpenCV 4: Computer Vision with Python”** by Adrian Kaehler and Gary Bradski, O’Reilly Media, 2019.
4. **“Deep Learning for Vision Systems”** by Mohamed Elgendy, Manning Publications, 2020.
5. **“Computer Vision: Models, Learning, and Inference”** by Simon J.D. Prince, Cambridge University Press, 2012.

Reference Books

1. **“Multiple View Geometry in Computer Vision”** by Richard Hartley and Andrew Zisserman, Cambridge University Press, 2003.
2. **“Programming Computer Vision with Python”** by Jan Erik Solem, O’Reilly Media, 2012.
3. **“Practical Python and OpenCV”** by Adrian Rosebrock, PyImageSearch, 2016.
4. **“Handbook of Mathematical Models in Computer Vision”** by Nikos Paragios, Yunmei Chen, Olivier Faugeras, Springer, 2006.
5. Research papers and online documentation for **OpenCV, TensorFlow, PyTorch, and Keras**.

BIG DATA ENGINEERING

BCO673A	Big Data Engineering	3-0-0 [3]

OBJECTIVE:

- To understand the concepts, characteristics, and applications of Big Data systems.
- To learn distributed data storage and parallel processing frameworks such as Hadoop and Spark.
- To explore data ingestion, integration, and real-time streaming tools.
- To gain proficiency in NoSQL databases and data management techniques.
- To apply Big Data analytics for real-world applications using modern tools and cloud platforms.

UNIT 1	Introduction to Big Data: Definition of Big Data, Characteristics of Big Data (Volume, Velocity, Variety, Veracity, Value), Traditional vs Big Data systems, Big Data applications in business, healthcare, social media, and IoT, Big Data ecosystem overview.
UNIT 2	Big Data Storage and Processing Frameworks: Distributed file systems (HDFS concepts, architecture, operations), Introduction to Hadoop ecosystem, MapReduce programming model, YARN resource management, Apache Spark basics (RDDs, DataFrames, SparkSQL).
UNIT 3	Data Ingestion, Integration, and Streaming: Data ingestion tools (Sqoop, Flume, Kafka), Data integration workflows, ETL vs ELT, Real-time streaming data, Spark Streaming, Structured Streaming, Introduction to Apache Storm and Flink.
UNIT 4	NoSQL Databases and Data Management: Types of NoSQL databases (Key-value, Document, Columnar, Graph), MongoDB, Cassandra, HBase fundamentals, Querying and indexing in NoSQL, CAP theorem, Consistency models, Data governance, Data quality, and Data security.
UNIT 5	Big Data Analytics and Applications: Machine Learning with Spark MLlib, Data visualization tools (Tableau, PowerBI), Big Data in Cloud platforms (AWS, Azure, GCP), Case studies in e-commerce, healthcare, finance, smart cities, Future trends in Big Data (Data Lakes, Data Mesh, Edge Analytics).

Course Outcome*:*

CO1: Understand the fundamentals and ecosystem of Big Data technologies.
CO2: Implement distributed storage and processing using Hadoop and Spark frameworks.
CO3: Apply data ingestion, integration, and real-time streaming solutions in Big Data pipelines.
CO4: Work with NoSQL databases for efficient data management and retrieval.
CO5: Analyze, visualize, and apply Big Data analytics techniques for solving real-world problems.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M			M								H		
CO2	H	H	M	M	H								H	M	
CO3	M	H	H	H	H								M	H	
CO4	M	H	H	H	H								H	H	M
CO5	H	M	H	H	H		M						H	H	M

H = Highly Related; M = Medium L = Low

Textbooks

1. **Tom White**, *Hadoop: The Definitive Guide*, O'Reilly Media, 4th Edition, 2015.
2. **Rajkumar Buyya, Rodrigo N. Calheiros, Amir Vahid Dastjerdi**, *Big Data: Principles and Paradigms*, Elsevier, 2016.
3. **Jules S. Damji, Brooke Wenig, Tathagata Das, Denny Lee**, *Learning Spark: Lightning-Fast Big Data Analysis*, O'Reilly, 2020.
4. **Viktor Mayer-Schönberger, Kenneth Cukier**, *Big Data: A Revolution That Will Transform How We Live, Work, and Think*, Eamon Dolan/Mariner Books, 2014.

Reference Books

1. **Nataraj Dasgupta**, *Practical Big Data Analytics: Hands-on Techniques to Implement Enterprise Analytics and Machine Learning Using Hadoop, Spark, NoSQL, and R*, Packt, 2018.
2. **Chuck Lam**, *Hadoop in Action*, Manning Publications, 2010.
3. **Saurabh Gupta**, *Implementing Splunk Big Data Analytics*, Packt, 2016.
4. Latest **Apache documentation (Hadoop, Spark, Kafka, Flume, Cassandra, MongoDB)**.

CLOUD COMPUTING FOR DATA SCIENCE

BCO674A	**Cloud Computing For Data Science**	**3-0-0 [3]**

OBJECTIVE:

- To introduce the fundamentals of cloud computing and its relevance in Data Science.
- To learn cloud service and deployment models with hands-on exposure to leading cloud platforms.
- To explore cloud-based data engineering, pipelines, and storage solutions.
- To understand cloud security, governance, and cost optimization strategies.
- To apply cloud-native tools for building and deploying Data Science and AI solutions.

UNIT 1	Introduction to Cloud Computing: Definition and characteristics of cloud computing, Cloud service models (IaaS, PaaS, SaaS), Deployment models (Public, Private, Hybrid, Community), Virtualization concepts, Containerization (Docker, Kubernetes), Benefits and challenges of cloud adoption in Data Science.
UNIT 2	Cloud Platforms and Services for Data Science: Overview of major cloud providers (AWS, Azure, GCP), Cloud storage systems (Amazon S3, Azure Blob, Google Cloud Storage), Distributed computing frameworks on cloud (Hadoop, Spark), Cloud-based machine learning services (AWS SageMaker, Azure ML Studio, Google Vertex AI).
UNIT 3	Data Engineering on the Cloud: Cloud data ingestion tools and pipelines (AWS Kinesis, Google Pub/Sub, Azure Event Hub), Data integration services, ETL on the cloud, Cloud data warehouses (Amazon Redshift, Google BigQuery, Azure Synapse), Real-time streaming analytics.
UNIT 4	Security, Governance, and Cost Optimization: Cloud security fundamentals, Identity and Access Management (IAM), Data privacy and compliance (GDPR, HIPAA), Resource monitoring and optimization, Cost estimation models, Scalability, fault tolerance, Service Level Agreements (SLAs).
UNIT 5	Applications of Cloud Computing in Data Science: Building and deploying ML/DL models on the cloud, Cloud-based data visualization (Tableau Cloud, PowerBI Service), Case studies in healthcare, e-commerce, finance, and IoT, Edge and serverless computing for data science applications, Future trends in cloud-native AI and big data solutions.

Course Outcome:

CO1: Understand the fundamentals, service models, and architectures of cloud computing.
CO2: Apply cloud platforms and services for large-scale data storage and distributed computation.
CO3: Design and manage cloud-based data engineering pipelines for batch and real-time processing.
CO4: Evaluate and implement cloud security, governance, and cost management practices.
CO5: Deploy Data Science and AI solutions using cloud-native tools and frameworks for real-world applications.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M			M								H		
CO2	H	H	H	M	H								H	M	
CO3	M	H	H	H	H								M	H	
CO4	M	H	M	H	H	M	M						H	H	M
CO5	H	M	H	H	H		M						H	H	M

H = Highly Related; M = Medium L = Low

Textbooks

1. **Rajkumar Buyya, James Broberg, Andrzej Goscinski**, *Cloud Computing: Principles and Paradigms*, Wiley, 2011.
2. **Thomas Erl, Zaigham Mahmood, Ricardo Puttini**, *Cloud Computing: Concepts, Technology & Architecture*, Pearson, 2013.
3. **Chris Fregly, Antje Barth**, *Data Science on AWS: Implementing End-to-End, Continuous AI and Machine Learning Pipelines*, O'Reilly, 2021.
4. **Pradeep Kulkarni**, *Cloud Computing for Data-Intensive Applications*, Springer, 2020.

Reference Books

1. **Gaurav Aroraa, Rashi Mehrotra**, *Hands-on Cloud Analytics with Microsoft Azure*, Packt, 2018.
2. **Kevin L. Jackson, Scott Goessling**, *Architecting Cloud Computing Solutions*, Packt, 2018.
3. **Saurabh Shrivastava, Smruti Ranjan Sarangi**, *Building Cloud Computing Solutions for Data Scientists*, Wiley, 2019.
4. Documentation from **AWS, Azure, Google Cloud** for latest cloud-native Data Science services.

B. Tech. - IV Semester
Contact Hours (L-T-P): 0-0-1

BCO637A	**MACHINE LEARNING LAB**	**Total Credits: 1**

List of Experiments

Students are required to perform any ten experiments out of the following list of experiments.

1	**Data Preprocessing**: Perform data cleaning (handle missing values, duplicates, outliers) and normalization on a real-world dataset (e.g., Titanic / Diabetes dataset).
2	**Feature Selection**: Apply feature selection techniques (correlation, variance threshold, chi-square test) and explain their effect on model performance.
3	**Evaluation Metrics Demo**: Implement accuracy, precision, recall, F1-score, and ROC curve on a sample classification dataset.
4	**Linear Regression**: Predict house prices/student scores using single-variable regression.
5	**Polynomial Regression**: Implement polynomial regression on a dataset with nonlinear patterns.
6	**Logistic Regression**: Classify survival on Titanic dataset (binary classification).
7	**k-Nearest Neighbors (k-NN)**: Classify Iris dataset flowers using different k values and compare accuracy.
8	**Decision Tree Classifier**: Build a decision tree for student performance prediction and visualize the tree.
9	**Support Vector Machine (SVM)**: Implement SVM on a dataset (e.g., Breast Cancer dataset) and visualize the hyperplane.
10	**k-Means Clustering**: Apply k-Means clustering for customer segmentation and visualize clusters.
11	**Hierarchical Clustering**: Perform hierarchical clustering on a dataset (e.g., Mall Customers dataset) and plot dendrogram.
12	**Association Rule Mining**: Implement Apriori algorithm on a transactional dataset (Market Basket Analysis).
13	**Dimensionality Reduction**: Apply PCA and t-SNE on high-dimensional data and visualize reduced features.
14	**Ensemble Models**: Compare Bagging, Random Forest, and Gradient Boosting on the same dataset and evaluate results.

15	**Neural Network Basics**: Implement a simple **Multi-Layer Perceptron (MLP)** for digit recognition (MNIST dataset) using TensorFlow/Keras.

Course Outcomes-

At the end of the course, students will be able to:

CO1: *Apply data preprocessing, normalization, feature engineering, and visualization techniques to prepare datasets for ML models.*

CO2: *Implement supervised learning algorithms (Regression, Classification, SVM, Decision Trees, k-NN) and analyze their performance.*

CO3: *Apply unsupervised learning techniques (Clustering, PCA, t-SNE, Association Rule Mining) and interpret their results.*

CO4: *Evaluate and compare advanced ML techniques such as Ensemble methods and Neural Networks for predictive tasks.*

CO5: *Design and develop mini-projects applying ML models to real-world problems using standard libraries and frameworks.*

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	M	H	M					L		L	H	M	
CO2	H	H	M	H	M					L		L	H	M	M
CO3	H	M	M	H	M					L		L	H	M	M
CO4	H	H	M	H	H					M		M	H	H	M
CO5	H	H	H	H	H	L	L		M	H	M	H	H	H	H

H = Highly Related; M = Medium L = Low

B. Tech. - V Semester
Contact Hours (L-T-P): 0-0-1

BCO639A	**DEEP LEARNING LAB**	**Total Credits: 1**

List of Experiments

Students are required to perform any ten experiments out of the following list of experiments.

1	Implement and visualize different **activation functions** (Sigmoid, Tanh, ReLU, Leaky ReLU, Softmax) and compare outputs.
2	Build a **single-layer perceptron** for solving simple binary classification (e.g., AND/OR gate).
3	Train a **multi-layer perceptron (MLP)** on the Iris dataset and analyze accuracy.
4	Implement **forward propagation and backpropagation** manually (without frameworks).
5	Train a simple **MLP on MNIST dataset** for handwritten digit classification using TensorFlow/Keras.
6	Compare performance of different **optimization algorithms (SGD, Adam)** on the same dataset.
7	Implement **regularization techniques (L1, L2, Dropout)** and compare model performance with/without them.
8	Implement a **basic CNN** model on MNIST or CIFAR-10 dataset.
9	Build and compare performance of **LeNet and VGG-like CNN models**.
10	Apply **Transfer Learning** using a pre-trained CNN (e.g., ResNet or VGG16) for image classification tasks.
11	Implement a **basic RNN** for text sequence prediction.
12	Train an **LSTM model** for time series forecasting (e.g., stock price prediction).
13	Implement a **GRU model** for sentiment analysis on IMDB movie review dataset.
14	Build an **Autoencoder** for image reconstruction and denoising.
15	Implement a **Generative Adversarial Network (GAN)** for generating synthetic images (MNIST/Fashion-MNIST).

Course Outcomes-

At the end of the course, students will be able to:

CO1: *Apply fundamental concepts of deep learning including activation functions, perceptron models, and optimization techniques.*

CO2: *Implement and train feedforward neural networks (MLPs) and evaluate their performance using appropriate metrics.*

CO3: *Apply convolutional neural networks (CNNs) and transfer learning techniques for image recognition and computer vision tasks.*

CO4: *Implement recurrent neural networks (RNNs), LSTMs, and GRUs for sequence modeling and NLP applications.*

CO5: *Design and develop advanced deep learning models such as autoencoders and GANs, and apply frameworks (TensorFlow/Keras/PyTorch) for real-world problem solving.*

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	M	H	M					L		L	H	M	
CO2	H	H	M	H	M					L		L	H	M	M
CO3	H	M	M	H	M					L		L	H	M	M
CO4	H	H	M	H	H					M		M	H	H	M
CO5	H	H	H	H	H	L	L		M	H	M	H	H	H	H

H = Highly Related; M = Medium L = Low

B. Tech. - V Semester

Contact Hours (L-T-P): 0-0-1

BCO617A	LARGE LANGUAGE MODELS LAB	Total Credits: 1

List of Experiments

Students are required to perform any ten experiments out of the following list of experiments.

1	**Tokenization Techniques:** Implement Byte Pair Encoding (BPE) and SentencePiece tokenization on a sample dataset and analyze vocabulary size vs. performance.
2	**Word Embeddings Visualization:** Train Word2Vec/Glove embeddings and visualize semantic similarity using PCA or t-SNE.
3	**Attention Mechanism Demo:** Implement self-attention on a small sequence and visualize attention weights.
4	**Mini Transformer Model:** Build a simple transformer model (encoder-decoder) for sequence-to-sequence tasks like translation.
5	**Pre-trained LLM Usage:** Use Hugging Face Transformers to load GPT-2 or BERT and generate text on custom prompts.
6	**Fine-Tuning Small LLM:** Fine-tune DistilBERT or GPT-2 on a domain-specific dataset (e.g., news or product reviews).
7	**Prompt Engineering Techniques:** Compare zero-shot, few-shot, and chain-of-thought prompting using OpenAI or Hugging Face APIs.
8	**Retrieval-Augmented Generation (RAG):** Implement document retrieval + LLM pipeline to answer questions from a custom corpus.
9	**LLM Evaluation Metrics:** Evaluate model outputs using BLEU, ROUGE, and perplexity on generated text.
10	**Bias Detection in LLMs:** Analyze outputs of a pre-trained LLM for potential bias using predefined prompts.
11	**LoRA Fine-Tuning:** Implement Low-Rank Adaptation (LoRA) on a small transformer model for efficient fine-tuning.
12	**Multi-Modal LLM Demo:** Use CLIP or LLaVA to generate image captions and compare with text-only models.
13	**Model Compression:** Apply quantization or pruning on a small LLM and evaluate inference speed and accuracy trade-offs.
14	**Safety and Alignment Testing:** Evaluate model responses for harmful content and propose prompt-level mitigations.
15	**Deploying LLM on Cloud:** Host a fine-tuned model using Hugging Face Spaces or Azure ML endpoint.
16	**AutoGPT/Agentic Demo:** Create a simple multi-step autonomous agent using LLM APIs to complete a defined goal.

Course Outcomes-

While graduating, students of the Large Language Models Lab program would be able to:

CO1: *Demonstrate understanding of LLM components (tokenization, attention, transformer blocks) through hands-on experiments.*

CO2: *Fine-tune and evaluate pre-trained LLMs for specific tasks using modern tools and frameworks.*

CO3: *Apply prompt engineering, RAG, and parameter-efficient methods to optimize performance.*

CO4: *Analyze ethical and safety aspects in LLM outputs and propose mitigation strategies.*

CO5: *Deploy and integrate LLM-based applications in real-world scenarios using cloud and open-source platforms.*

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M		L	M							M	H	M	M
CO2	M	H	M	H	H							M	H	H	M
CO3	M	H	H	M	H	L		M		M		M	H	H	H
CO4	M	M	L	M	H	H	M	H		M		H	M	M	H
CO5	H	M	H	M	H	M	L	L	M	M	L	M	H	H	H

H = Highly Related; M = Medium L = Low

B. Tech. - VI Semester
Contact Hours (L-T-P): 0-0-1

BCO641A	**NATURAL LANGUAGE PROCESSING LAB**	**Total Credits: 1**

List of Experiments

Students are required to perform any ten experiments out of the following list of experiments.

1	Implement **text preprocessing pipeline**: tokenization, stopword removal, stemming, lemmatization.
2	Perform **POS tagging** on text using **NLTK and SpaCy**.
3	Apply **regular expressions** for text extraction (e.g., emails, phone numbers, hashtags from raw text).
4	Implement **unigram, bigram, trigram language models** and calculate sentence probabilities.
5	Apply **smoothing techniques** (Laplace, Add-k) on n-gram models.
6	Build and visualize a **syntax tree** using context-free grammar (CFG) parsing.
7	Implement **dependency parsing** and analyze grammatical relations in text.
8	Perform **Named Entity Recognition (NER)** using SpaCy/Hugging Face Transformers.
9	Calculate **semantic similarity** between words/sentences using WordNet and embeddings (Word2Vec/GloVe).
10	Implement **Word Sense Disambiguation (WSD)** using the Lesk algorithm.
11	Implement **Naïve Bayes classifier** for text classification (e.g., spam detection).
12	Train an **LSTM/GRU model** for sentiment analysis on IMDB dataset.
13	Fine-tune a **pre-trained BERT model** for text classification (e.g., sentiment/emotion detection).
14	Build a **chatbot** using rule-based methods and extend with transformer-based models (DialoGPT/ChatGPT API).
15	Implement **machine translation** (English–French or English–Hindi) using a seq2seq or Transformer model; evaluate with BLEU/ROUGE scores.

Course Outcomes-

At the end of the NLP lab, students will be able to:

CO1: *Apply text preprocessing techniques (tokenization, stemming, lemmatization, POS tagging) and regular expressions for structured text analysis.*

CO2: *Implement statistical language models, parsing techniques, and syntactic analysis for natural language processing tasks.*

CO3: *Perform semantic analysis including WSD, NER, semantic similarity, and embeddings to derive meaning from text.*

CO4: *Apply machine learning and neural network-based NLP techniques (Naïve Bayes, LSTM, Transformer, BERT) for text classification and sequence modeling.*

CO5: *Design and develop NLP applications (chatbots, sentiment analysis, machine translation) using standard frameworks (NLTK, SpaCy, Hugging Face) and evaluate them with appropriate metrics.*

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	M	H	M					L		L	H	M	
CO2	H	H	M	H	M					L		L	H	M	M
CO3	H	M	M	H	H					M		M	H	H	M
CO4	H	H	M	H	H					M		M	H	H	H
CO5	H	H	H	H	H	L	L		M	H	M	H	H	H	H

OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

H = Highly Related; M = Medium L = Low

B. Tech. - VI Semester
Contact Hours (L-T-P): 0-0-1

BCO672A	COMPUTER VISION AND IMAGE PROCESSING LAB	Total Credits: 1

List of Experiments

Students are required to perform any ten experiments out of the following list of experiments.

1	Implement image acquisition and representation: loading, displaying, and saving images using **OpenCV**.
2	Perform **image sampling and quantization** at different resolutions and analyze the effect.
3	Apply **geometric transformations** (scaling, rotation, translation, flipping) on images.
4	Implement **contrast stretching and histogram equalization** for image enhancement.
5	Apply **smoothing and sharpening filters** (mean, Gaussian, Laplacian).
6	Perform **edge detection** using Sobel, Prewitt, and Canny operators.
7	Implement **image restoration** using inverse filtering and Wiener filtering.
8	Perform **thresholding techniques** (global, adaptive, Otsu’s method).
9	Implement **region growing, splitting & merging** for segmentation.
10	Apply **watershed segmentation** and visualize results.
11	Extract **features** (corners, blobs, SIFT, SURF, HOG descriptors) from images.
12	Implement **template matching and HOG+SVM classifier** for object detection.
13	Detects faces using **Haar cascade classifiers**.
14	Implement a **YOLO/SSD model** for real-time object detection using a pre-trained CNN.
15	Implement **object tracking** using Kalman filter or CAMShift on video sequences.

Course Outcomes-

CO1: *Apply fundamental operations of image representation, sampling, quantization, and transformations.*

CO2: *Implement image enhancement and restoration techniques for improving image quality.*

CO3: *Perform image segmentation and feature extraction for meaningful representation of images.*

CO4: *Develop object detection, recognition, and tracking models using traditional and deep learning approaches.*

CO5: *Design and evaluate computer vision applications using OpenCV, TensorFlow, and PyTorch frameworks for real-world problems.*

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	M	H	M					L		L	H	M	
CO2	H	H	M	H	M					L		L	H	M	M
CO3	H	M	M	H	H					M		M	H	H	M
CO4	H	H	M	H	H					M		M	H	H	H
CO5	H	H	H	H	H	L	L		M	H	M	H	H	H	H

OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

H = Highly Related; M = Medium L = Low

B. Tech. - VII Semester
Contact Hours (L-T-P): 0-0-1

BCO675A	BIG DATA ENGINEERING LAB	Credit: 1

List of Experiments

Students are required to perform any ten experiments out of the following list of experiments.

1	Install and configure a **Hadoop environment** (single-node setup).
2	Explore **Big Data ecosystem** tools and demonstrate a simple word count program.
3	**Case study**: Identify Big Data applications in healthcare/social media and analyze datasets (descriptive analytics).
4	Implement basic **HDFS operations**: upload, retrieve, and delete files.
5	Develop a **MapReduce program** for word count, inverted index, and sorting.
6	Implement a data transformation pipeline using **Spark RDDs and DataFrames**.
7	Perform SQL queries on large datasets using **SparkSQL**.
8	Import relational database data into Hadoop using **Sqoop**.
9	Stream data from log files into HDFS using **Flume**.
10	Implement a real-time producer-consumer pipeline using **Kafka**.
11	Perform **real-time data processing** using **Spark Streaming** (e.g., word count from live tweets or logs).
12	Perform CRUD operations in **MongoDB** (insert, query, update, delete).
13	Design and query a **Cassandra database** for time-series data.
14	Implement indexing and query optimization in **HBase**.
15	Build a **machine learning pipeline in Spark MLlib** (classification/regression on a real dataset) and visualize results in **Tableau/PowerBI**.

Course Outcomes-

At the end of this lab, students will be able to:

CO1: *Demonstrate understanding of Big Data ecosystem tools and perform basic HDFS and Hadoop operations.*

CO2: *Implement distributed data processing using MapReduce and Apache Spark frameworks.*

CO3: *Apply data ingestion and streaming techniques using Sqoop, Flume, and Kafka for real-time data integration.*

CO4: *Design and manage data using NoSQL databases (MongoDB, Cassandra, HBase) ensuring scalability and consistency.*

CO5: *Develop analytics solutions using Spark MLlib and visualization tools to address real-world Big Data applications.*

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	M	H	M					L		L	H	M	
CO2	H	H	M	H	M					L		L	H	M	M
CO3	H	M	M	H	H					M		M	H	H	M
CO4	H	H	M	H	H					M		M	H	H	H
CO5	H	H	H	H	H	L	L		M	H	M	H	H	H	H

OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

H = Highly Related; M = Medium L = Low

B. Tech. - IV Semester
Contact Hours (L-T-P): 0-0-1

BCO637A	MACHINE LEARNING LAB	Total Credits: 1

List of Experiments

Students are required to perform any ten experiments out of the following list of experiments.

1	**Data Preprocessing**: Perform data cleaning (handle missing values, duplicates, outliers) and normalization on a real-world dataset (e.g., Titanic / Diabetes dataset).
2	**Feature Selection**: Apply feature selection techniques (correlation, variance threshold, chi-square test) and explain their effect on model performance.
3	**Evaluation Metrics Demo**: Implement accuracy, precision, recall, F1-score, and ROC curve on a sample classification dataset.
4	**Linear Regression**: Predict house prices/student scores using single-variable regression.
5	**Polynomial Regression**: Implement polynomial regression on a dataset with nonlinear patterns.
6	**Logistic Regression**: Classify survival on Titanic dataset (binary classification).
7	**k-Nearest Neighbors (k-NN)**: Classify Iris dataset flowers using different k values and compare accuracy.
8	**Decision Tree Classifier**: Build a decision tree for student performance prediction and visualize the tree.
9	**Support Vector Machine (SVM)**: Implement SVM on a dataset (e.g., Breast Cancer dataset) and visualize the hyperplane.
10	**k-Means Clustering**: Apply k-Means clustering for customer segmentation and visualize clusters.
11	**Hierarchical Clustering**: Perform hierarchical clustering on a dataset (e.g., Mall Customers dataset) and plot dendrogram.
12	**Association Rule Mining**: Implement Apriori algorithm on a transactional dataset (Market Basket Analysis).
13	**Dimensionality Reduction**: Apply PCA and t-SNE on high-dimensional data and visualize reduced features.
14	**Ensemble Models**: Compare Bagging, Random Forest, and Gradient Boosting on the same dataset and evaluate results.
15	**Neural Network Basics**: Implement a simple **Multi-Layer Perceptron (MLP)** for digit recognition (MNIST dataset) using TensorFlow/Keras.

Course Outcomes-

At the end of the course, students will be able to:

CO1: *Apply data preprocessing, normalization, feature engineering, and visualization techniques to prepare datasets for ML models.*

CO2: *Implement supervised learning algorithms (Regression, Classification, SVM, Decision Trees, k-NN) and analyze their performance.*

CO3: *Apply unsupervised learning techniques (Clustering, PCA, t-SNE, Association Rule Mining) and interpret their results.*

CO4: *Evaluate and compare advanced ML techniques such as Ensemble methods and Neural Networks for predictive tasks.*

CO5: *Design and develop mini-projects applying ML models to real-world problems using standard libraries and frameworks.*

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	M	H	M					L		L	H	M	
CO2	H	H	M	H	M					L		L	H	M	M
CO3	H	M	M	H	M					L		L	H	M	M
CO4	H	H	M	H	H					M		M	H	H	M
CO5	H	H	H	H	H	L	L		M	H	M	H	H	H	H

H = Highly Related; M = Medium L = Low

B. Tech. - V Semester
Contact Hours (L-T-P): 0-0-1

BCO639A	**DEEP LEARNING LAB**	**Total Credits: 1**

List of Experiments

Students are required to perform any ten experiments out of the following list of experiments.

1	Implement and visualize different **activation functions** (Sigmoid, Tanh, ReLU, Leaky ReLU, Softmax) and compare outputs.
2	Build a **single-layer perceptron** for solving simple binary classification (e.g., AND/OR gate).
3	Train a **multi-layer perceptron (MLP)** on the Iris dataset and analyze accuracy.
4	Implement **forward propagation and backpropagation** manually (without frameworks).
5	Train a simple **MLP on MNIST dataset** for handwritten digit classification using TensorFlow/Keras.
6	Compare performance of different **optimization algorithms (SGD, Adam)** on the same dataset.
7	Implement **regularization techniques (L1, L2, Dropout)** and compare model performance with/without them.
8	Implement a **basic CNN** model on MNIST or CIFAR-10 dataset.
9	Build and compare performance of **LeNet and VGG-like CNN models**.
10	Apply **Transfer Learning** using a pre-trained CNN (e.g., ResNet or VGG16) for image classification tasks.
11	Implement a **basic RNN** for text sequence prediction.
12	Train an **LSTM model** for time series forecasting (e.g., stock price prediction).
13	Implement a **GRU model** for sentiment analysis on IMDB movie review dataset.
14	Build an **Autoencoder** for image reconstruction and denoising.
15	Implement a **Generative Adversarial Network (GAN)** for generating synthetic images (MNIST/Fashion-MNIST).

Course Outcomes-

At the end of the course, students will be able to:

CO1: *Apply fundamental concepts of deep learning including activation functions, perceptron models, and optimization techniques.*

CO2: *Implement and train feedforward neural networks (MLPs) and evaluate their performance using appropriate metrics.*

CO3: *Apply convolutional neural networks (CNNs) and transfer learning techniques for image recognition and computer vision tasks.*

CO4: *Implement recurrent neural networks (RNNs), LSTMs, and GRUs for sequence modeling and NLP applications.*

CO5: *Design and develop advanced deep learning models such as autoencoders and GANs, and apply frameworks (TensorFlow/Keras/PyTorch) for real-world problem solving.*

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	M	H	M					L		L	H	M	
CO2	H	H	M	H	M					L		L	H	M	M
CO3	H	M	M	H	M					L		L	H	M	M
CO4	H	H	M	H	H					M		M	H	H	M
CO5	H	H	H	H	H	L	L		M	H	M	H	H	H	H

H = Highly Related; M = Medium L = Low

B. Tech. - V Semester

Contact Hours (L-T-P): 0-0-1

BCO617A	LARGE LANGUAGE MODELS LAB	Total Credits: 1

List of Experiments

Students are required to perform any ten experiments out of the following list of experiments.

1	**Tokenization Techniques:** Implement Byte Pair Encoding (BPE) and SentencePiece tokenization on a sample dataset and analyze vocabulary size vs. performance.
2	**Word Embeddings Visualization:** Train Word2Vec/Glove embeddings and visualize semantic similarity using PCA or t-SNE.
3	**Attention Mechanism Demo:** Implement self-attention on a small sequence and visualize attention weights.
4	**Mini Transformer Model:** Build a simple transformer model (encoder-decoder) for sequence-to-sequence tasks like translation.
5	**Pre-trained LLM Usage:** Use Hugging Face Transformers to load GPT-2 or BERT and generate text on custom prompts.
6	**Fine-Tuning Small LLM:** Fine-tune DistilBERT or GPT-2 on a domain-specific dataset (e.g., news or product reviews).
7	**Prompt Engineering Techniques:** Compare zero-shot, few-shot, and chain-of-thought prompting using OpenAI or Hugging Face APIs.
8	**Retrieval-Augmented Generation (RAG):** Implement document retrieval + LLM pipeline to answer questions from a custom corpus.
9	**LLM Evaluation Metrics:** Evaluate model outputs using BLEU, ROUGE, and perplexity on generated text.
10	**Bias Detection in LLMs:** Analyze outputs of a pre-trained LLM for potential bias using predefined prompts.
11	**LoRA Fine-Tuning:** Implement Low-Rank Adaptation (LoRA) on a small transformer model for efficient fine-tuning.
12	**Multi-Modal LLM Demo:** Use CLIP or LLaVA to generate image captions and compare with text-only models.
13	**Model Compression:** Apply quantization or pruning on a small LLM and evaluate inference speed and accuracy trade-offs.
14	**Safety and Alignment Testing:** Evaluate model responses for harmful content and propose prompt-level mitigations.
15	**Deploying LLM on Cloud:** Host a fine-tuned model using Hugging Face Spaces or Azure ML endpoint.
16	**AutoGPT/Agentic Demo:** Create a simple multi-step autonomous agent using LLM APIs to complete a defined goal.

Course Outcomes-

While graduating, students of the Large Language Models Lab program would be able to:

CO1: *Demonstrate understanding of LLM components (tokenization, attention, transformer blocks) through hands-on experiments.*

CO2: *Fine-tune and evaluate pre-trained LLMs for specific tasks using modern tools and frameworks.*

CO3: *Apply prompt engineering, RAG, and parameter-efficient methods to optimize performance.*

CO4: *Analyze ethical and safety aspects in LLM outputs and propose mitigation strategies.*

CO5: *Deploy and integrate LLM-based applications in real-world scenarios using cloud and open-source platforms.*

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M		L	M							M	H	M	M
CO2	M	H	M	H	H							M	H	H	M
CO3	M	H	H	M	H	L		M		M		M	H	H	H
CO4	M	M	L	M	H	H	M	H		M		H	M	M	H
CO5	H	M	H	M	H	M	L	L	M	M	L	M	H	H	H

H = Highly Related; M = Medium L = Low

B. Tech. - VI Semester
Contact Hours (L-T-P): 0-0-1

BCO641A	NATURAL LANGUAGE PROCESSING LAB	Total Credits: 1

List of Experiments

Students are required to perform any ten experiments out of the following list of experiments.

1	Implement **text preprocessing pipeline**: tokenization, stopword removal, stemming, lemmatization.
2	Perform **POS tagging** on text using **NLTK and SpaCy**.
3	Apply **regular expressions** for text extraction (e.g., emails, phone numbers, hashtags from raw text).
4	Implement **unigram, bigram, trigram language models** and calculate sentence probabilities.
5	Apply **smoothing techniques** (Laplace, Add-k) on n-gram models.
6	Build and visualize a **syntax tree** using context-free grammar (CFG) parsing.
7	Implement **dependency parsing** and analyze grammatical relations in text.
8	Perform **Named Entity Recognition (NER)** using SpaCy/Hugging Face Transformers.
9	Calculate **semantic similarity** between words/sentences using WordNet and embeddings (Word2Vec/GloVe).
10	Implement **Word Sense Disambiguation (WSD)** using the Lesk algorithm.
11	Implement **Naïve Bayes classifier** for text classification (e.g., spam detection).
12	Train an **LSTM/GRU model** for sentiment analysis on IMDB dataset.
13	Fine-tune a **pre-trained BERT model** for text classification (e.g., sentiment/emotion detection).
14	Build a **chatbot** using rule-based methods and extend with transformer-based models (DialoGPT/ChatGPT API).
15	Implement **machine translation** (English–French or English–Hindi) using a seq2seq or Transformer model; evaluate with BLEU/ROUGE scores.

Course Outcomes-

At the end of the NLP lab, students will be able to:

CO1: *Apply text preprocessing techniques (tokenization, stemming, lemmatization, POS tagging) and regular expressions for structured text analysis.*

CO2: *Implement statistical language models, parsing techniques, and syntactic analysis for natural language processing tasks.*

CO3: *Perform semantic analysis including WSD, NER, semantic similarity, and embeddings to derive meaning from text.*

CO4: *Apply machine learning and neural network-based NLP techniques (Naïve Bayes, LSTM, Transformer, BERT) for text classification and sequence modeling.*

CO5: *Design and develop NLP applications (chatbots, sentiment analysis, machine translation) using standard frameworks (NLTK, SpaCy, Hugging Face) and evaluate them with appropriate metrics.*

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	M	H	M					L		L	H	M	
CO2	H	H	M	H	M					L		L	H	M	M
CO3	H	M	M	H	H					M		M	H	H	M
CO4	H	H	M	H	H					M		M	H	H	H
CO5	H	H	H	H	H	L	L		M	H	M	H	H	H	H

OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

H = Highly Related; M = Medium L = Low

B. Tech. - VI Semester
Contact Hours (L-T-P): 0-0-1

BCO672A	COMPUTER VISION AND IMAGE PROCESSING LAB	Total Credits: 1

List of Experiments

Students are required to perform any ten experiments out of the following list of experiments.

1	Implement image acquisition and representation: loading, displaying, and saving images using **OpenCV**.
2	Perform **image sampling and quantization** at different resolutions and analyze the effect.
3	Apply **geometric transformations** (scaling, rotation, translation, flipping) on images.
4	Implement **contrast stretching and histogram equalization** for image enhancement.
5	Apply **smoothing and sharpening filters** (mean, Gaussian, Laplacian).
6	Perform **edge detection** using Sobel, Prewitt, and Canny operators.
7	Implement **image restoration** using inverse filtering and Wiener filtering.
8	Perform **thresholding techniques** (global, adaptive, Otsu's method).
9	Implement **region growing, splitting & merging** for segmentation.
10	Apply **watershed segmentation** and visualize results.
11	Extract **features** (corners, blobs, SIFT, SURF, HOG descriptors) from images.
12	Implement **template matching and HOG+SVM classifier** for object detection.
13	Detects faces using **Haar cascade classifiers**.
14	Implement a **YOLO/SSD model** for real-time object detection using a pre-trained CNN.
15	Implement **object tracking** using Kalman filter or CAMShift on video sequences.

Course Outcomes-

CO1: *Apply fundamental operations of image representation, sampling, quantization, and transformations.*

CO2: *Implement image enhancement and restoration techniques for improving image quality.*

CO3: *Perform image segmentation and feature extraction for meaningful representation of images.*

CO4: *Develop object detection, recognition, and tracking models using traditional and deep learning approaches.*

CO5: *Design and evaluate computer vision applications using OpenCV, TensorFlow, and PyTorch frameworks for real-world problems.*

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	M	H	M					L		L	H	M	
CO2	H	H	M	H	M					L		L	H	M	M
CO3	H	M	M	H	H					M		M	H	H	M
CO4	H	H	M	H	H					M		M	H	H	H
CO5	H	H	H	H	H	L	L		M	H	M	H	H	H	H

OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

H = Highly Related; M = Medium L = Low

B. Tech. - VII Semester
Contact Hours (L-T-P): 0-0-1

BCO675A	BIG DATA ENGINEERING LAB	Credit: 1

List of Experiments

Students are required to perform any ten experiments out of the following list of experiments.

1	Install and configure a **Hadoop environment** (single-node setup).
2	Explore **Big Data ecosystem** tools and demonstrate a simple word count program.
3	**Case study**: Identify Big Data applications in healthcare/social media and analyze datasets (descriptive analytics).
4	Implement basic **HDFS operations**: upload, retrieve, and delete files.
5	Develop a **MapReduce program** for word count, inverted index, and sorting.
6	Implement a data transformation pipeline using **Spark RDDs and DataFrames**.
7	Perform SQL queries on large datasets using **SparkSQL**.
8	Import relational database data into Hadoop using **Sqoop**.
9	Stream data from log files into HDFS using **Flume**.
10	Implement a real-time producer-consumer pipeline using **Kafka**.
11	Perform **real-time data processing** using **Spark Streaming** (e.g., word count from live tweets or logs).
12	Perform CRUD operations in **MongoDB** (insert, query, update, delete).
13	Design and query a **Cassandra database** for time-series data.
14	Implement indexing and query optimization in **HBase**.
15	Build a **machine learning pipeline in Spark MLlib** (classification/regression on a real dataset) and visualize results in **Tableau/PowerBI**.

Course Outcomes-

At the end of this lab, students will be able to:

CO1: *Demonstrate understanding of Big Data ecosystem tools and perform basic HDFS and Hadoop operations.*

CO2: *Implement distributed data processing using MapReduce and Apache Spark frameworks.*

CO3: *Apply data ingestion and streaming techniques using Sqoop, Flume, and Kafka for real-time data integration.*

CO4: *Design and manage data using NoSQL databases (MongoDB, Cassandra, HBase) ensuring scalability and consistency.*

CO5: *Develop analytics solutions using Spark MLlib and visualization tools to address real-world Big Data applications.*

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	M	H	M					L		L	H	M	
CO2	H	H	M	H	M					L		L	H	M	M
CO3	H	M	M	H	H					M		M	H	H	M
CO4	H	H	M	H	H					M		M	H	H	H
CO5	H	H	H	H	H	L	L		M	H	M	H	H	H	H

OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

H = Highly Related; M = Medium L = Low