LNCT University B. Tech CSE

VI SEM

Subject: Data Science Toolkits CS601

COURSE OUTCOMES:

After Completing the course student should be able to

CO1	Learn various data preprocessing techniques
CO2	.Evaluate supervised ad unsupervised techniques with python performance functions
CO3	Analyze and forecast time series data using Recurrent Neural Networks (RNNs) and ARIMA techiques.
CO4	Analyze datasets using basic statistical and graphical techniques in R to gain meaningful insights
CO5	Design version controlling utility for Data science real life problem

COURSECONTENT

UNIT -1 Introduction to Data Science – Evolution of Data Science – Data Science Roles – Stages in a Data Science Project – Applications of Data Science in various fields – Data Security Issues-Data Preprocessing overview and Techniques of Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization-Exploratory Data Analytics- Descriptive Statistics – Mean, Standard Deviation, Skewness and Kurtosis – Box Plots – Pivot Table – Heat Map

Unit II Python for Data Science Review of Numpy, Pandas and Scikit-learn.Supervised Learning Techniques packages/toolkit for regression and classification: - Ensemble Methods, Ordinary Least Squares Regression, Logistic Regression, etc. Unsupervised Learning, Optimization Using Evolutionary Techniques

Unit III Deep Learning and Time Series Analysis Basics of TensorFlow and keras,Basics of PyTorch, perform style transfer of one image to another, Perform text generation, and sentiment analysis with PyTorch. Overview of CNN, Time series regression and exploratory data analysis toolkits: ARMA/ARIMA models, model identification/estimation/linear operators, Fourier analysis, spectral estimation, and state- space models.

Unit IV R for Data Science Basic of R and RStudio. R data structures: vectors, factors, lists, arrays, matrices, and data frames. Working with data: Import data into R and visualize data. Regression and Classification with R.

Unit V Overview of Data Analytics Software Weka, Orange, Rapidminer, Minitab, PowerBI, GitHub, Google Colab. Version controlling tools for data science projects. Case studies of data science projects.

References

- Cathy O'Neil and Rachel Schutt , "Doing Data Science", O'Reilly, 2015
- Brockwell& Davis (2016) Introduction to Time Series and Forecasting, 3rd edition, Springer
- R for Data Science: Import, Tidy, Transform, Visualize, and Model Data, 1st Edition, O'reilly publication.
- Ruppert& Matteson (2016) Statistics and Data Analysis for Financial Engineering with R examples, 2nd Edition, Springer

Suggested List of Experiments

1. READING AND WRITING DIFFERENT TYPES OF DATASETS using Python

a. Reading different types of data sets (.txt, .csv) from web and disk and writing in file in specific disk location.

b. Reading Excel data sheet in python. **(C)** Reading XML dataset in python.

2 VISUALIZATIONS:

c. Find the data distributions using box and scatter plot. b. Find the outliers using plot.

d. Plot the histogram, bar chart and pie chart on sample data

e. EXPLORATORY DATA ANALYSIS (EDA): Perform EDA on Credit Card Fraud Detection Dataset (open source dataset) for analyzing the data.

f. LINEAR REGRESSION MODEL FOR PREDICTION: Apply Regression Model techniques to predict the future values of data on the open source available datasets.

g. LOGISTIC REGRESSION MODEL: Import the Red-Wine dataset from the UCI Machine Learning Repository having three qualities of wines. Apply logistic regression model for multi-class classification of the wine categories.

h. MODEL EVALUATION USING RESIDUAL PLOT: Plotting Accuracy and Error Metrics against number of iterations for evaluation of model performance

i. Perform 1,2,3,4,5,with R programming

j. Illustrate Various data analytics tools with case studies

Natural Language Processingand Gen AI CS-602

COURSE OUTCOMES

After Completing the course student should be able to

CO1	Understand linguistic concepts and deep learning models for NLP.
CO2	Design and implement NLP pipelines with real-world data and preprocessing techniques.
CO3	Analyze transformer architectures and large language models (LLMs).
CO4	Apply and evaluate text generation methods in GenAI using advanced sampling strategies.
CO5	Explore cross-modal GenAI, including image and code generation, and address risks and safety concerns in LLMs.

Course contents

UNIT I –Basics ofNatural Language Processing &Deep Learning Approaches Introduction to NLP and applications: assistants, chatbots, translation. Key NLP tasks: sentiment analysis, NER, POS tagging, summarization. Linguistic foundations: syntax, semantics, ambiguity, code-switching. ML and Deep Learning for NLP: Naive Bayes, SVM, RNNs, LSTMs, CNNs. Transformers (BERT, GPT), Transfer Learning, Limitations

UNIT II – NLP Project Pipeline & Data Processing NLP Workflow: Define, Collect, Preprocess, Model, Evaluate, Deploy. Data Sources: Public datasets, Web scraping (BeautifulSoup, newspaper3k). Data Preprocessing: Cleaning, Tokenization, Stopwords, Normalization. Text Augmentation and Representation: BoW, TF-IDF, Word2Vec, GloVe, FastText. Contextual Embeddings: BERT, ELMo, Subword Models (BPE, WordPiece). Language Models: N-gram, MLM, CLM

UNIT III - Transformers and LLM Architectures Transformer Core: Self-Attention, Multi-Head Attention, Encoder/Decoder. Evolution: Transformer →BERT →GPT. Architectures: BERT (MLM), GPT (CLM). Fine-tuning vs Pretraining, LoRA, PEFT. LLM Evaluation: Perplexity, BLEU, ROUGE. Challenges: Hallucination, Bias, Computational Costs.

Unit IV - Text Generation and Evaluation in Generative AI Text Generation Techniques: Autoregressive, Conditional Generation. Sampling Strategies: Greedy, Beam, Top-k, Top-p. Evaluation: Fluency, BLEU, ROUGE, Human Assessment. Prompt Engineering, Conditioning, Safe Decoding. Mitigating Risks: Bias, Hallucinations, Toxicity

Unit V: Cross-Modal GenAI and Safety in LLMs Cross-Modal GenAI: Text-to-Image (DALL-E, Stable Diffusion), Text-to-Code (Codex, Gemini). Intro to Audio/Video Generation, Prompt Chaining, RAG. Risks: Deepfakes, Synthetic Media, Misuse. Safety Measures: RLHF, Guardrails, Filtering. Policies: EU AI Act, NITI Aayog Guidelines.

Reference Books-

1. Natural Language Processing with Python by Steven Bird, Ewan Klein and Edward Loper.
2. Foundations of Statistical Natural Language Processing by Christophe

LNCT University B. Tech CSE

List of Experiment

1. Write a program to perform text preprocessing (tokenization, stopword removal, stemming, lemmatization) using NLTK and spaCy.
2. Write a program to perform sentiment analysis on a movie review dataset using a Naive Bayes classifier.
3. Write a program to implement POS tagging and Named Entity Recognition (NER) using spaCy.
4. Write a program to implement text vectorization using Bag-of-Words and TF-IDF on a corpus of documents.
5. Write a program to train a Word2Vec model using Gensim and visualize word embeddings using PCA or t-SNE.
6. Write a program to compare contextual embeddings from BERT with traditional embeddings (e.g., Word2Vec).
7. Write a program to implement a text classification pipeline using Scikit-learn (e.g., classify news headlines).
8. Write a program to implement a simple RNN or LSTM for text generation using Keras or PyTorch.
9. Write a program to fine-tune a pre-trained BERT model using Hugging Face Transformers for a classification task.
10. Write a program to implement a chatbot using rule-based and retrieval-based methods.
11. Write a program to apply transformer-based summarization (e.g., using T5 or BART on an article).
12. Write a program to implement greedy, top-k, and top-p sampling for text generation using GPT-2.
13. Write a program to evaluate generated text using BLEU and ROUGE metrics.
14. Write a program to generate text-to-image prompts using DALL·E API or similar models.
15. Write a program to convert natural language to code using OpenAI Codex or Gemini APIs.
16. Write a program to demonstrate Retrieval-Augmented Generation (RAG) using a simple custom knowledge base.
17. Write a program to apply prompt engineering techniques for a text completion task using LLMs.
18. Write a program to demonstrate LoRA or PEFT fine-tuning on a lightweight transformer model.
19. Write a program to identify and mitigate hallucinations and bias in LLM-generated responses.
20. Write a program to analyze AI-generated output for ethical risks and apply filtering/guardrails.

LNCT University B. Tech CSE

CS603(a) Cloud Computing-II

COURSE OUTCOMES:After Completing the course student should be able to

CO1	Understand the basics of DevOps Infrastrusture.
CO2	Understand and configure devops framework.
CO3	Deploy with DevOps integration.
CO4	Design Applications with Docker and Kubernetes
CO5	Evaluate various DevOps Tools with case studies.

Course Contents

UNIT – I DevOps Infrastructure: What is DevOps, Implement Continuous Integration (CI), Continuous Delivery (CD), and Continuous Delivery (CD), understand Infrastructure as Code (IaC) practices, Business drivers for DevOps adoption, data explosion, cloud computing, Big data, data science and machine learning, planning DevOps strategy, benefits of DevOps, devops lifecycle,devops vs traditional operations

UNIT – II DevOps Framework: DevOps process, Source code management, code review, configuration management, build management, Artifacts repository management, release management, test automation, routine automation, DevOps maturity life cycle, DevOps Maturity Map, DevOps progression framework, DevOps Maturity checklists, Agile framework, Cloud as the Foundation

UNIT – III DevOps as a Service – Continuous Integration, Delivery and Deployment: Best Practices for CI/CD, Jenkins setup, Git (SCM) integration with Jenkins, Integrating GitHub with Jenkins, Maven (Build) tool integration with Jenkins, Building Jobs with Jenkins, Source Code Review – Gerrit, Installation of Gerrit, Repository Management, Testing with Jenkins

UNIT – IV Continuous Delivery – Build Pipeline, DevOps continuous Deployment, Chef landscape components, features of Chef, Chef Automate workflow, Features of Ansible, Ansible CMDB, Playbooks, Modules, Inventory, Plugins, Ansible Tower, Ansible Vault, Ansible Galaxy, Monitoring, Aplunk, Nagios Monitoring Tool, Devops ad cloud security issues

UNIT-V Containerized Applications with Docker and Kubernetes: Installing Docker, Creating Dockerfile, Building and running a container on a local machine, pushing an image to Docker Hub, managing containers with Kubernetes, Technical requirements of Kubernetes, Kubernetes architecture overview, Installing Kubernetes on a local machine, Installing Kubernetes Dashboard, Using HELM as package manager, ecosystem

References

- The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in
- Technology Organizations, Gene Kim, Jez Humble, Patrick Debois, John Willis,TMH

Suggested List of Experiments

- Study and set up DevOps Infrastructure CO1
- Implement Continuous Integration (CI), Continuous Delivery (CD) with it CO1
- Configure DevOps Framework CO2
- Illustrate DevOps process for Source code management, code review, CO2
- Illustrate DevOps process for build management, Artifacts repository management, CO3
- Illustrate it for release management, test automation CO3
- Integrate Agile framework with DevOps CO4
- Build Pipeline and Automate workflow with tools CO4
- Install docker and create Dockerfile CO5
- Install Kubernetes and review its architecture CO5

LNCT University B. Tech CSE

Salesforce (CS 603 (b))

Syllabus Objectives: Skill Development: Gain expertise in Apex programming and Salesforce development. Practical Application: Implement real-world business logic using Apex, SOQL, triggers, and batch classes. Certification Readiness: Prepare for Salesforce certifications, such as Salesforce Platform Developer I and II. Deployment Proficiency: Learn best practices for testing and deploying Salesforce solutions

COURSE OUTCOMES:

After Completing the course student should be able to:

CO1	Understand and implement Salesforce triggers with best practices and trigger design patterns
CO2	Develop and test asynchronous Apex using Batch classes and test frameworks.
CO3	Build and manage Lightning Web Components using Salesforce DX and modern web standards
CO4	Integrate Salesforce with external systems using REST/SOAP APIs and authentication methods
CO5	Configure Agentforce and navigate core Salesforce features for agent productivity and support

Unit 1: Triggers in Salesforce Understanding Triggers: What Are Triggers, Trigger Events (Before and After Triggers), Trigger Context Variables. Trigger Syntax and Structure: Writing Basic Triggers, Trigger Helper Classes for Best Practices. Common Use Cases of Triggers: Automating Field Updates, Managing Related Records, Custom Business Logic. Trigger Design Patterns: Bulkifying Triggers, Avoiding Recursive Triggers. Hands-On Practice:

Unit 2: Batch Apex and Test Class Introduction to Asynchronous Processing: Importance of Asynchronous Apex, Overview of Batch Apex. Batch Apex Components: Writing Batch Classes (Start, Execute, Finish Methods), Database.QueryLocator and Iterable Interfaces. Scheduling and Monitoring Batch Jobs: Scheduling Batch Jobs Using Apex Scheduler, Viewing and Debugging Batch Jobs Introduction to Salesforce Testing: Importance of Test Classes, Apex Testing Framework. Writing Effective Test Classes: Best Practices for Test Classes, Using Test Methods and @isTest Annotation, Generating Test Data Using Test Setup

Unit 3 Introduction to Salesforce Lightning Web Components (LWC) Overview of LWC: Evolution of Lightning Framework, Benefits of Lightning Web Components, LWC vs Aura Components. LWC Basics: Anatomy of an LWC (HTML, JavaScript, and CSS Files), Setting Up Salesforce DX and Scratch Orgs for LWC Development, Lightning Web Component Bundles. Core Concepts: Component Lifecycle Hooks, Data Binding and State Management, Handling Events in LWC. Hands-On Practice.

Unit 4: Salesforce Integration Basics Introduction to Salesforce Integration: Overview of APIs and Integration Approaches, Understanding REST and SOAP APIs in Salesforce. Outbound Integrations: Callouts from Apex Using REST and SOAP, Configuring Named Credentials for Secure Integration. Inbound Integrations: Creating RESTful Services in Salesforce Authentication Mechanisms (OAuth 2.0, JWT). Integration Tools: Postman for API Testing, Salesforce Workbench for API Exploration. Hands-On Practice:

Unit 5: Introduction to Agentforce and Salesforce Fundamentals Understanding Agentforce: Overview of Agentforce and Its Role in Salesforce, Key Features and Use Cases of Agentforce.Salesforce Basics: Introduction to Salesforce Platform, Navigating Salesforce UI and Apps, Standard and Custom Objects in Salesforce. Configuring Agentforce : Creating Custom Agent Profiles and Roles, Configuring Communication Channels, Setting Up Agent Availability and Skills Omni-Channel and Productivity Tools

Reference Books for Salesforce Development

1. "Advanced Apex Programming for Salesforce Developers" by Dan Appleman A detailed guide on Apex programming with a focus on advanced techniques and best practices.
2. "Salesforce Platform Developer I Certification Guide" by Jan Vandevelde and Gaurav Kheterpal A comprehensive resource for mastering Salesforce development concepts and preparing for the PD-1 certification.
3. "Learning Salesforce Development with Apex" by Paul Battisson A beginner-friendly book to understand Apex, SOQL, triggers, and Salesforce development fundamentals.
4. "Mastering Salesforce DevOps" by Andrew Davis Covers deployment strategies, CI/CD pipelines, and best practices for Salesforce development and testin

Suggested List of Experiment

1. Write a Salesforce Apex Trigger to automatically update a custom field on Account whenever a new Contact is inserted.
2. Develop a Trigger with Helper Class to prevent recursive updates and implement bulkification when inserting multiple Opportunity records.
3. Write a Trigger to maintain a count of related Contacts on the Account object whenever Contacts are inserted or deleted.
4. Create a Batch Apex Class that processes all Accounts and updates a custom "Last Processed Date" field using Database.QueryLocator.
5. Write a Batch Apex Class using Iterable interface to process large data records and schedule it using Apex Scheduler.
6. Create a Test Class for an existing Apex Trigger to achieve at least 75% code coverage using @isTest annotation and Test Setup methods.
7. Create a basic Lightning Web Component (LWC) that displays a list of Accounts fetched via Apex class.
8. Develop an LWC with input fields to create new Contact records and display confirmation messages on successful creation.
9. Write an LWC to handle component events and parent-child communication for Account and related Contacts display.
10. Build a RESTful Apex web service that exposes Account data as a JSON response and test it using Postman.
11. Develop an Apex Callout to consume an external REST API using Named Credentials for authentication.
12. Create and configure an OAuth 2.0 connected app in Salesforce and test the authentication process using Workbench.
13. Create Agentforce custom agent profiles and roles, configure Omni-Channel presence statuses and assign skills to agents.
14. Configure Omni-Channel routing for incoming cases to the available agents using the configured Agentforce settings
15. Create custom Reports and Dashboards to monitor Agentforce productivity metrics such as case response time and resolution time

LNCT University B. Tech CSE

LNCT University B.TECH-AIML

CS604 Design Thinking and Innovation

Course objectives To expose the student with state of the art perspectives, ideas, concepts, and solutions related to the design and execution of projects using design thinking principles To prepare the mindset and discipline of systemic inspiration driven by a desire to identify new sources of ideas, and new models especially outside their regular working atmosphere To propose a concrete, feasible, viable and relevant innovation project/challenge

COURSE OUTCOMES:

After Completing the course student should be able to

CO1	Learn basic terminology design thinking.
CO2	Apply empathizing techniques for real life design problems.
CO3	Develop maps as design thinking tools for primary and secondary research
CO4	Implement story telling and scenario planning to strengthen innovation.
CO5	Create business model for engineering domain real life challenges.

Course Contents

Module 1 What is Different About Design thinking? Design Thinking Skills ,Principles of Design Thinking, The Basis for Design Thinking, Process,tools,project and case study, models of design thinking

Module 2 Listening and Empathizing Techniques – observation – structured open ended approach - , Analysis, Design Thinking Frameworks, Ideation tools – brainstorming, innovation heuristics, behaviour models, overcoming cognitive fixedness – Exercises and case based discussions

Module 3 Use of Diagrams and Maps in Design Thinking – Empathy map. Affinity diagram, mind map, journey map, combining ideas into complex innovation concepts. Secondary research and primary research, contextual inquirt

Module 4 Story telling – improvisation, scenario planning, development of scenarios, evaluation tools, frog design and prototyping – soft,medium,final – apply frameworks to strengthen communication – sustain a culture of innovation, usability studies

Module 5 Engineering aspect of design, Electrical, Mechanical, Design, Material, Aspect, Safety and Reliability aspect , Creating business model, Introduction of Startup with entrepreneurship approach: What is entrepreneurship, being an entrepreneurship, Challenges and possibilities of Entrepreneurship? How to Start up, Start- up Fundamental, Being Successful

References

1. Roger Martin, "The Design of Business: Why Design Thinking is the Next Competitive Advantage",Harvard Business press , 2009.
2. Hasso Plattner, Christoph Meinel and Larry Leifer (eds), "Design Thinking: Understand – Improve– Apply", Springer, 2011
3. Idris Mootee, "Design Thinking for Strategic Innovation: What They Can't Teach You at Business or Design School", John Wiley & Sons 2013
4. Jeanne Liedtka , Andrew King, Kevin Bennett , “Book - Solving Problems with Design Thinking - Ten Stories of What Works” (Columbia Business School Publishing), 2013
5. Maurício Vianna, Ysmar Vianna, Isabel K. Adler, Brenda Lucena, Beatriz Russo, “Design thinking: Business Innovation” MJV Press, 2011
6. Burgelman, Christensen, and Wheelwright, “Strategic Management of Technology and Innovation”5th Edition, McGraw Hill Publications,

www.dsource.in

DT&I, Case Studies, Courses, Tools, and Resources

https://dsource.in/dti https://dsource.in/case-study https://dsource.in/course https://dsource.in/tools https://dsource.in/resource

\

List of experiments
Case study Jellow Communication . Practice Project selection and brain storming

1. Mindmapping of selected project Examples – Smaran design for elderly ,GOGO app for elderly
2. Perform user participating mapping for the project , ex. Smart agriculture, Solar powered Pesticide Sprayer
3. Design Cue cards for sample project for ex. Lapcrate
4. Perform storytelling for E-commerce websites ::Identify project artifacts, activity, spatial mappings for the same
5. Complete analysis of Project Geolight and apply tools brain storming, idea sketching, scamper
6. Perform usability study for any one project

SYLLABUS FINAL YEAR
CS -701

The subject comprises of tutorial/assignment/internal lab work evaluation for Industry Modules Suggested by Training and Placement/Academic/Industry MOU (Samatrix)/NPTEL MOOC Course certificate for the emerging subjects not covered in curriculum

CS-702CS-801

IT will be report and seminar delivery by the student after completing Industry Internship as suggested by Training and Placement cell or under Industry MOU

CS-703

In 7th semester it is expected by student to complete Detailed Planning and Architectural design as per SDLC for the major project in the domain of web application/IoT project/AI based technology. Student will maintain Agile board/dashboard to show the progress. There will be monthly progress report and at the end of the semester, project presentation and end sem evaluation will be done through internal and external viva voce.

CS-802

In 8th semester it is expected that student will continue with the major project and complete development as per SDLC for the major project in the domain of web application/IoT project/AI based technology. Student will maintain Agile board/dashboard to show the progress and sprint review. There will be monthly progress report and at the end of the semester, project demonstration and illustration will be evaluated in end sem external viva voce along with report submission.