CLEVERA
Thrive Through Cleverness

ARTIFICIAL INTELLIGENCE MACHINE LEARNING Cookbook

AI & ML Mastery Program

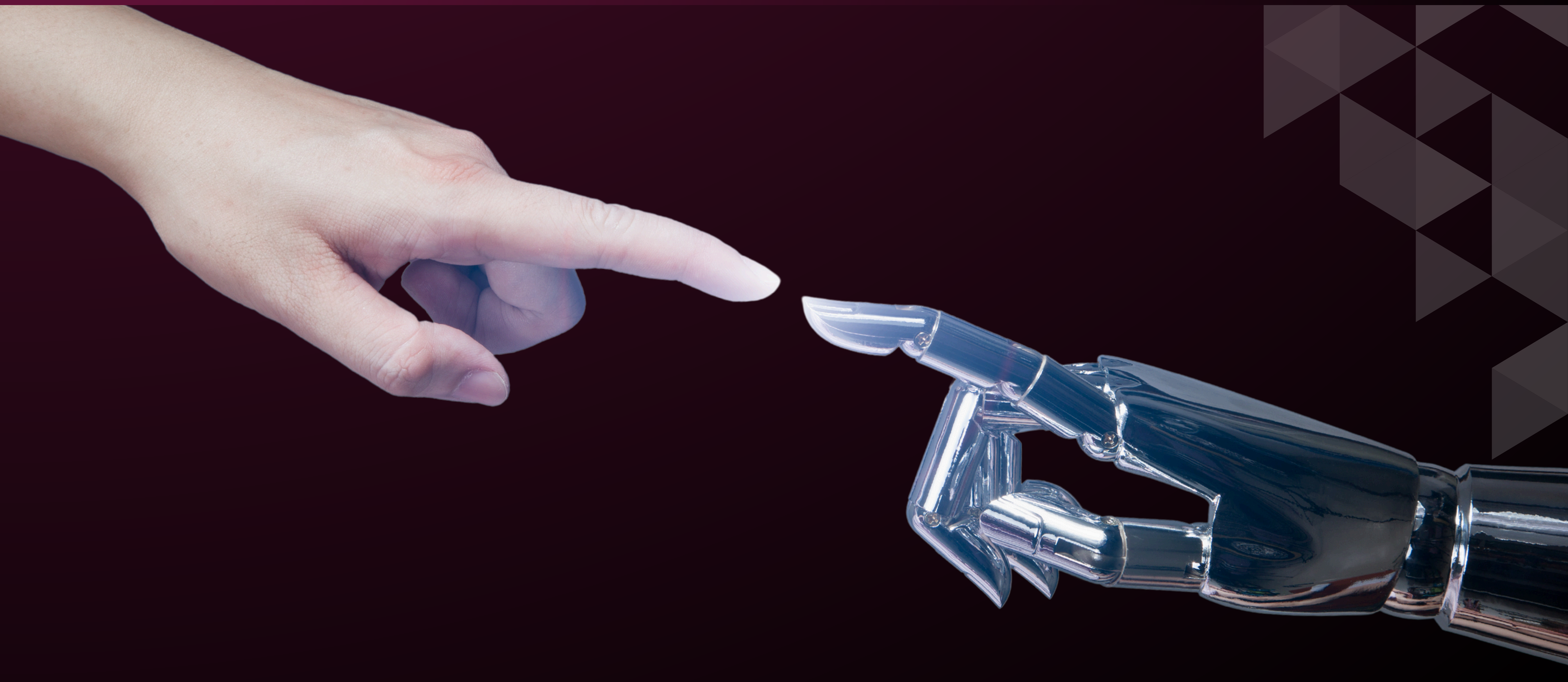

TABLE *Of Contents*

28. Advanced NLP
29. Gen AI
30. Fundamentals of Gen AI
31. Prompt Engineering
32. LLM models
33. Langchain for GenAI
34. Vector databases for GenAI
35. GenAI assignment

About Artificial Intelligence

AI is the ability of machines to mimic human cognitive functions. By crunching vast amounts of data, AI systems can identify patterns, make predictions, and solve problems – often even better than humans. This AI training piques your curiosity and sets the stage for a deeper dive into AI. It highlights the transformative potential of the technology while keeping the explanation concise and engaging.

About Software Development Industry

The global software engineering market is expected to grow to approximately USD $37.4 billion by 2022, at a CAGR of nearly 12 percent. Corporate initiatives focused on digital transformation will help drive demand for skilled software developers, which is reflected in current hiring trends and future projections. According to NASSCOM and USA Today, one million software development jobs will be added to the workforce by 2020 in India and 1.4 million jobs in the United States.

Many companies prefer to hire multi-skilled technology professionals such as automation test engineers. The average annual salary for an automation test engineer is USD $94,270 (ZipRecruiter).

Additional facts about the state of the software development industry:

- Junior developers are getting massive starting salaries compared to those of the last 20 years.
- Upskilling is one of the most important priorities for developers today (Stackoverflow 2019)
- The phenomenal rise of consumer applications in both web and mobile is driven by the availability of open-source projects and libraries
- Smaller, quicker releases—which results in better productivity—are becoming crucial for software product success. Automation engineers are well-positioned to empower this trend.

Key Features

Comprehensive Blended Learning program

120 hours of self-paced learning

Choose from 4 industry- aligned capstone projects

270 hours of in-depth training

20+ in-demand tools and skills

Job-assist program included

150 hours of instructor- led training

10 lesson-end & 4 phase - end projects

Along with this, 24x7 Support Available.

Top Skills and Tools Covered

Top Skills Covered:

- Python Syntax and Basics
- Statistical Analysis
- Data Manipulation
- Machine Learning
- Predictive Modeling
- Exploratory Data Visualization

Top Tools Covered:

- Jupyter Notebook
- PyCharm
- NumPy
- Pandas
- Scikit-learn
- Matplotlib
- Seaborn

Topic 1

Introduction to Python

Unlock the gateway to programming with our Introduction to Python course. Python, renowned for its simplicity and versatility, is the ideal language for both novice learners and seasoned developers. This module lays a robust foundation in Python programming, covering fundamental concepts such as variables, data types, control structures, and functions. Through interactive sessions and practical exercises, participants will familiarize themselves with industry-standard tools like Jupyter Notebook and PyCharm.

Frameworks and Tools Covered:

- Jupyter Notebook: Interactive computing environment for writing and executing Python code.
- PyCharm: Integrated Development Environment (IDE) offering features such as code completion, debugging, and version control.

Top Skills Covered:

- Python Syntax and Basics: Understanding the syntax and structure of Python programming.
- Data Types and Variables: Learning about different data types like integers, floats, strings, lists, tuples, and dictionaries.
- Control Structures: Mastering flow control with concepts like if statements, loops, and functions.
- Hands-on Experience: Gaining practical skills through coding exercises and projects.

Topic 2

Python for Data Science

Delve into the world of data science with Python as your primary tool. This module equips participants with essential skills in data manipulation, analysis, and modeling using libraries like NumPy, Pandas, and Scikit-learn. Through hands-on projects, participants will learn to preprocess data, perform statistical analysis, and build predictive models, setting the stage for a successful career in data science.

Frameworks and Tools Covered:

- NumPy: Library for numerical computing, offering support for arrays and mathematical functions.
- Pandas: Data manipulation library providing data structures and operations for structured data.
- Scikit-learn: Machine learning library featuring algorithms for classification, regression, clustering, and dimensionality reduction.

Top Skills Covered:

- Data Manipulation: Preprocessing and cleaning datasets, handling missing values, and transforming data for analysis.
- Statistical Analysis: Exploring data distributions, identifying patterns, and deriving insights using statistical methods.
- Machine Learning Basics: Understanding supervised and unsupervised learning, model training, and evaluation.
- Predictive Modeling: Building and assessing predictive models for regression and classification tasks.

Topic 3

Data Visualization using Python

Master the art of data visualization and transform complex datasets into insightful visuals. In this module, participants will harness the power of Python's visualization libraries, including Matplotlib and Seaborn. They will learn to create a wide range of plots and charts, customize visualizations, and communicate findings effectively through data visualization.

Frameworks and Tools Covered:

- Matplotlib: Versatile plotting library for creating static, interactive, and animated visualizations.
- Seaborn: Statistical data visualization library offering high-level interfaces for creating attractive and informative plots.

Top Skills Covered:

- Plotting Basics: Creating basic plots and customizing visual properties such as colors, markers, and labels.
- Exploratory Data Visualization: Visualizing data distributions, identifying trends, and uncovering patterns in data.
- Customization and Annotation: Enhancing visualizations with annotations, legends, and titles to convey insights effectively.
- Dashboarding and Reporting: Creating interactive dashboards and reports to present findings to stakeholders.

Topic 4

Exploratory Data Analysis

Unlock hidden insights within datasets through Exploratory Data Analysis (EDA). This module empowers participants to explore, visualize, and summarize data using Python's data manipulation and visualization libraries. Participants will gain proficiency in descriptive statistics, data distributions, correlation analysis, and outlier detection methods, enabling them to derive meaningful insights from data.

Frameworks and Tools Covered:

- Pandas: Data manipulation library for exploring and analyzing structured data.
- Matplotlib and Seaborn: Visualization libraries for creating informative plots and charts.
- NumPy: Mathematical library for performing numerical computations and statistical analysis.

Top Skills Covered:

- Descriptive Statistics: Calculating and interpreting summary statistics such as mean, median, mode, variance, and standard deviation.
- Data Distribution Analysis: Understanding the distribution of data through histograms, density plots, and box plots.
- Correlation Analysis: Exploring relationships between variables using correlation matrices and scatter plots.
- Outlier Detection: Identifying and handling outliers using statistical methods and visualization techniques.

Topic 5

Inferential Stats

Master the art of inferential statistics and make robust inferences about populations from sample data. In this module, participants will learn about probability distributions, sampling methods, confidence intervals, and hypothesis testing using Python's scientific computing and statistical analysis libraries. Through practical exercises, participants will develop the skills to apply inferential statistics to real-world datasets confidently.

Frameworks and Tools Covered:

- SciPy: Scientific computing library for performing mathematical functions and statistical tests.
- StatsModels: Statistical modeling library for exploring data, estimating statistical models, and performing hypothesis tests.

Top Skills Covered:

- Probability Distributions: Understanding common probability distributions such as normal, binomial, and Poisson distributions.
- Sampling Methods: Exploring techniques such as simple random sampling, stratified sampling, and cluster sampling.
- Confidence Intervals: Estimating population parameters and calculating confidence intervals for population means and proportions.
- Hypothesis Testing: Formulating null and alternative hypotheses, choosing appropriate test statistics, and interpreting

Topic 6

Hypothesis Testing

Learn to validate assumptions and make data-driven decisions with hypothesis testing. This module dives deep into hypothesis testing methods and procedures using Python's scientific computing and statistical analysis libraries. Participants will explore parametric and nonparametric tests, one-sample and two-sample tests, and chi-square tests, gaining the skills to conduct rigorous hypothesis tests.

Frameworks and Tools Covered:

- SciPy: Library for scientific computing and statistical analysis in Python, providing functions for hypothesis testing and statistical modeling.
- StatsModels: Library for exploring data, estimating statistical models, and performing hypothesis tests.

Top Skills Covered:

- Null and Alternative Hypotheses: Formulating research hypotheses and defining null and alternative hypotheses.
- Test Statistics: Choosing appropriate test statistics based on the type of data and research question.
- P-values and Significance Levels: Interpreting p-values and determining statistical significance based on significance levels.
- Type I and Type II Errors: Understanding the concepts of Type I and Type II errors and their implications for hypothesis testing.

Topic 7

Linear Regression

Master the foundational technique of linear regression for predictive modeling. In this module, participants will learn about simple and multiple linear regression, model assumptions, parameter estimation, and model evaluation metrics using Python's machine learning libraries. Through practical exercises, participants will develop the skills to build and evaluate linear regression models effectively.

Frameworks and Tools Covered:

- Scikit-learn: Machine learning library for building and evaluating predictive models in Python, including linear regression.

Top Skills Covered:

- Simple Linear Regression: Modeling the relationship between two variables using a straight line.
- Multiple Linear Regression: Extending linear regression to multiple independent variables to capture complex relationships.
- Model Assumptions: Understanding assumptions such as linearity, independence, homoscedasticity, and normality.
- Model Evaluation Metrics: Assessing model performance using metrics such as R-squared, adjusted R-squared, and RMSE.

Topic 8

Logistic Regression

Unravel the mysteries of logistic regression for binary classification tasks. This module introduces participants to logistic regression concepts, including the logistic function, odds ratio, model interpretation, and evaluation metrics. Participants will learn to build and evaluate logistic regression models using Python's machine learning libraries, paving the way for effective classification modeling.

Frameworks and Tools Covered:

- Scikit-learn: Machine learning library for building and evaluating predictive models in Python, including logistic regression.

Top Skills Covered:

- Binary Classification: Understanding the principles of binary classification and logistic regression modeling.
- Logistic Function: Modeling the probability of a binary outcome using the logistic function.
- Model Interpretation: Interpreting logistic regression coefficients and odds ratios to understand the relationship between variables.
- Model Evaluation Metrics: Assessing classification model performance using metrics such as accuracy, precision, recall, F1 score, and ROC-AUC.

Topic 9

Naive Bayes

Unlock the power of Naive Bayes for probabilistic classification. In this module, participants will explore different types of Naive Bayes classifiers, including Gaussian Naive Bayes, Multinomial Naive Bayes, and Bernoulli Naive Bayes. Through practical exercises, participants will learn to build and evaluate Naive Bayes models for classification tasks such as spam detection and sentiment analysis.

Frameworks and Tools Covered:

- Scikit-learn: Machine learning library for building and evaluating predictive models in Python, including Naive Bayes classifiers.

Top Skills Covered:

- Probabilistic Classification: Understanding the principles of probabilistic classification and the Naive Bayes algorithm.
- Types of Naive Bayes Classifiers: Exploring variants based on the distribution of feature variables.
- Model Evaluation: Assessing classification model performance using metrics such as accuracy, precision, recall, F1 score, and ROC-AUC.
- Text Classification: Applying Naive Bayes classifiers to tasks such as spam detection, sentiment analysis, and document categorization.

Topic 10

Advanced Regression

Master advanced regression techniques to model complex relationships in data. In this module, participants will explore polynomial regression, regularization methods like Ridge and Lasso regression, and model evaluation techniques using Python's machine learning libraries. Through hands-on exercises, participants will develop the skills to build and evaluate advanced regression models effectively.

Frameworks and Tools Covered:

- Scikit-learn: Machine learning library for building and evaluating predictive models in Python, including advanced regression techniques.

Top Skills Covered:

- Polynomial Regression: Modeling nonlinear relationships between variables using polynomial regression.
- Regularization: Preventing overfitting and improving model generalization using Ridge and Lasso regression.
- Model Evaluation: Assessing regression model performance using metrics such as R-squared, adjusted R-squared, RMSE, and MAE.
- Hyperparameter Tuning: Optimizing model performance by tuning hyperparameters such as regularization strength and polynomial degree.

Topic 11

Tree Models

Explore decision tree algorithms and ensemble methods for predictive modeling. This module covers decision tree algorithms, including classification and regression trees (CART), random forests, and gradient boosting machines (GBM). Through practical exercises, participants will learn to build and evaluate decision tree models effectively.

Frameworks and Tools Covered:

- Scikit-learn: Machine learning library for building and evaluating predictive models in Python, including decision tree algorithms.

Top Skills Covered:

- Decision Tree Basics: Understanding decision tree structure, splitting criteria, and pruning techniques.
- Ensemble Learning: Exploring bagging and boosting methods for combining multiple decision trees.
- Random Forests: Building robust and scalable predictive models using random forest algorithms.
- Gradient Boosting Machines: Leveraging gradient boosting algorithms like XGBoost and LightGBM for improved predictive performance.

Topic 12

Bagging

Master bagging, an ensemble learning technique for improving model performance. This module covers bootstrap resampling, bagged decision trees, and bagged ensembles. Through practical exercises, participants will learn to build and evaluate bagging models effectively.

Frameworks and Tools Covered:

- Scikit-learn: Machine learning library for building and evaluating predictive models in Python, including bagging algorithms.

Top Skills Covered:

- Bootstrap Resampling: Generating multiple training datasets by sampling with replacement from the original dataset.
- Bagged Decision Trees: Building ensemble models by training multiple decision trees on different bootstrap samples and aggregating their predictions.
- Model Evaluation: Assessing bagging model performance using metrics such as accuracy, precision, recall, F1 score, and ROC-AUC.
- Overfitting Reduction: Reducing overfitting and improving model generalization by averaging predictions from multiple models.

Topic 13

Boosting

Unleash the power of boosting, an ensemble learning technique for creating strong predictive models. This module covers AdaBoost, gradient boosting machines (GBM), and extreme gradient boosting (XGBoost). Through practical exercises, participants will learn to build and evaluate boosting models effectively.

Frameworks and Tools Covered:

- Scikit-learn: Machine learning library for building and evaluating predictive models in Python, including boosting algorithms.
- XGBoost: Scalable and efficient implementation of gradient boosting algorithms for classification and regression tasks.

Top Skills Covered:

- Weak Learners: Understanding the concept of weak learners and their role in boosting algorithms.
- AdaBoost: Training multiple weak learners sequentially and adjusting weights to focus on misclassified data points.
- Gradient Boosting Machines: Building ensemble models by sequentially adding decision trees to correct errors made by previous models.
- Model Evaluation: Assessing boosting model performance using metrics such as accuracy, precision, recall, F1 score, and ROC-AUC.

Topic 14

Clustering

Unlock patterns within data and discover hidden structures through clustering. This module covers K-means, hierarchical clustering, and DBSCAN algorithms. Through practical exercises, participants will learn to apply clustering techniques effectively.

Frameworks and Tools Covered:

- Scikit-learn: Machine learning library for building and evaluating predictive models in Python, including clustering algorithms.

Top Skills Covered:

- K-means Clustering: Partitioning data into K clusters by minimizing within-cluster variance.
- Hierarchical Clustering: Building a hierarchy of clusters by recursively merging or splitting clusters based on proximity.
- DBSCAN: Density-based clustering algorithm that identifies clusters based on regions of high density separated by areas of low density.
- Cluster Evaluation: Assessing clustering quality using metrics such as silhouette score, Davies-Bouldin index, and Calinski-Harabasz index.

Topic 15

Principle Component Analysis

Uncover the underlying structure of high-dimensional data through dimensionality reduction with Principle Component Analysis (PCA). This module covers eigenvalues, eigenvectors, and the singular value decomposition (SVD) algorithm. Through practical exercises, participants will learn to apply PCA effectively.

Frameworks and Tools Covered:

- Scikit-learn: Machine learning library for building and evaluating predictive models in Python, including PCA implementation.

Top Skills Covered:

- Dimensionality Reduction: Reducing the number of features in a dataset while preserving essential structure and minimizing information loss.
- Eigenvalues and Eigenvectors: Understanding mathematical concepts behind PCA and the eigendecomposition process.
- Singular Value Decomposition (SVD): Decomposing a matrix into orthogonal matrices and diagonal matrices to perform PCA.
- Visualization: Visualizing high-dimensional data in lower-dimensional space using PCA and scatter plots.

Topic 16

ML Assignment

Put your machine learning skills to the test with our hands-on machine learning assignment. In this module, participants will apply machine learning concepts and techniques to real-world datasets. They will work on supervised and unsupervised learning tasks, including classification, regression, and clustering. Through practical exercises and projects, participants will gain valuable experience in solving machine learning problems and analyzing results.

Frameworks and Tools Covered:

- Scikit-learn: Machine learning library for building and evaluating predictive models in Python, including various machine learning algorithms.

Top Skills Covered:

- Problem Formulation: Defining machine learning tasks and selecting appropriate algorithms based on problem characteristics.
- Data Preprocessing: Cleaning, transforming, and scaling datasets to prepare them for model training.
- Model Selection: Choosing the best machine learning algorithm and hyperparameters for a given task using cross-validation and grid search.
- Performance Evaluation: Assessing model performance using appropriate evaluation metrics and interpreting results.

Topic 17

Neural networks

Delve into the realm of deep learning with our Neural Networks module. Participants will explore neural network architectures, activation functions, and training techniques using TensorFlow and Keras. Through hands-on exercises, participants will learn to build and train neural networks for various tasks, including image classification and regression.

Frameworks and Tools Covered:

- TensorFlow: Open-source machine learning framework developed by Google for building and training neural networks and other machine learning models.
- Keras: High-level neural networks API for building and training deep learning models, built on top of TensorFlow.

Top Skills Covered:

- Neural Network Basics: Understanding the architecture & components of artificial neural networks, including neurons, layers, & activation functions.
- Backpropagation: Learning the mathematical principles behind backpropagation, the algorithm used to train neural networks by adjusting weights and biases.
- Optimization Techniques: Exploring optimization algorithms such as gradient descent, stochastic gradient descent, and Adam optimization for training neural networks.
- Model Training and Evaluation: Building, training, and evaluating neural network models for various tasks, including classification and regression.

Topic 18

CNN

Dive into the world of computer vision with Convolutional Neural Networks (CNNs). This module covers CNN architectures, convolutional layers, and pooling operations using TensorFlow and Keras. Through hands-on exercises, participants will learn to build and train CNNs for image classification tasks.

Frameworks and Tools Covered:

- TensorFlow: Open-source machine learning framework developed by Google for building and training neural networks and other machine learning models.
- Keras: High-level neural networks API for building and training deep learning models, built on top of TensorFlow.

Top Skills Covered:

- Convolutional Layers: Understanding convolutional filters, feature maps, and stride and padding operations in convolutional neural networks.
- Pooling Layers: Reducing spatial dimensions and extracting dominant features using pooling operations such as max pooling & average pooling.
- Popular CNN Architectures: Exploring state-of-the-art CNN architectures such as LeNet, AlexNet, and VGGNet for image classification tasks.
- Image Classification: Building and training CNN models to classify images into predefined categories, such as object recognition and facial recognition.

Topic 19

RNN

Unlock the power of Recurrent Neural Networks (RNNs) for sequential data analysis. This module covers RNN architectures, recurrent layers, and sequence modeling techniques using TensorFlow and Keras. Through hands-on exercises, participants will learn to build and train RNNs for tasks such as natural language processing and time series analysis.

Frameworks and Tools Covered:

- TensorFlow: Open-source machine learning framework developed by Google for building and training neural networks and other machine learning models.
- Keras: High-level neural networks API for building and training deep learning models, built on top of TensorFlow.

Top Skills Covered:

- Recurrent Layers: Understanding recurrent neural network architectures and mechanisms for modeling sequential data.
- Long Short-Term Memory (LSTM): Learning the architecture & functionality of LSTM cells for capturing long-term dependencies in sequential data.
- Gated Recurrent Unit (GRU): Exploring the simpler variant of LSTM cells.
- Model Training and Evaluation: Building and training RNN models for various sequential data analysis tasks.

Topic 20

DL Assignment

Challenge your deep learning skills with our hands-on assignment. In this module, participants will apply deep learning concepts and techniques to real-world datasets. They will work on tasks such as image classification, natural language processing, and time series analysis. Through practical exercises and projects, participants will gain valuable experience in solving deep learning problems and analyzing results.

Frameworks and Tools Covered:

- TensorFlow: Open-source machine learning framework developed by Google for building and training neural networks and other machine learning models.
- Keras: High-level neural networks API for building and training deep learning models, built on top of TensorFlow.

Top Skills Covered:

- Deep Learning Applications: Applying deep learning techniques to various domains such as computer vision, natural language processing, and time series analysis.
- Model Architectures: Designing and implementing deep learning architectures such as CNNs, RNNs, and autoencoders.
- Hyperparameter Tuning: Optimizing model performance by tuning hyperparameters such as learning rate, batch size, & network architecture.
- Performance Evaluation: Assessing deep learning model performance using appropriate evaluation metrics and interpreting results.

Topic 21

NLP - Lexical processing

Unlock the power of Natural Language Processing (NLP) with our Lexical Processing module. Participants will explore lexical analysis techniques such as tokenization, stemming, and lemmatization using Python's NLP libraries. Through hands-on exercises, participants will learn to preprocess text data effectively for downstream NLP tasks.

Frameworks and Tools Covered:

- NLTK (Natural Language Toolkit): Library for building NLP applications, providing tools for text processing, tokenization, stemming, and lemmatization.

Top Skills Covered:

- Tokenization: Breaking text into individual words or tokens for further analysis.
- Stemming and Lemmatization: Reducing words to their root forms to normalize text data.
- Stopword Removal: Filtering out common words that carry little semantic meaning from text data.
- Text Preprocessing Pipelines: Building preprocessing pipelines to clean and transform text data efficiently.

Topic 22

NLP - Syntactic Processing

Dive deeper into Natural Language Processing (NLP) with our Syntactic Processing module. Participants will explore syntactic analysis techniques such as part-of-speech tagging, parsing, and dependency parsing using Python's NLP libraries. Through hands-on exercises, participants will learn to extract syntactic structures and relationships from text data.

Frameworks and Tools Covered:

- NLTK (Natural Language Toolkit): Library for building NLP applications, providing tools for part-of-speech tagging, parsing, and dependency parsing.

Top Skills Covered:

- Part-of-Speech Tagging: Assigning grammatical tags to words based on their syntactic roles in sentences.
- Parsing: Analyzing the grammatical structure of sentences to identify syntactic relationships between words.
- Dependency Parsing: Parsing sentences to represent syntactic dependencies between words in a tree structure.
- Syntactic Analysis Pipelines: Building syntactic analysis pipelines to extract syntactic features from text data.

Topic 23

NLP - Semantic processing

Unlock the semantics of text data with our Semantic Processing module. Participants will explore semantic analysis techniques such as named entity recognition, word embeddings, and semantic similarity using Python's NLP libraries. Through hands-on exercises, participants will learn to extract semantic information and meanings from text data.

Frameworks and Tools Covered:

- NLTK (Natural Language Toolkit): Library for building NLP applications, providing tools for named entity recognition, word embeddings, and semantic similarity.

Top Skills Covered:

- Named Entity Recognition: Identifying and classifying named entities such as persons, organizations, and locations in text data.
- Word Embeddings: Representing words as dense vectors in continuous vector spaces to capture semantic similarities between words.
- Semantic Similarity: Measuring the similarity between words or sentences based on their semantic meanings.
- Semantic Analysis Pipelines: Building semantic analysis pipelines to extract semantic features from text data.

Topic 24

NLP Assignment

Put your Natural Language Processing (NLP) skills to the test with our hands-on assignment. In this module, participants will apply NLP concepts and techniques to real-world text data. They will work on tasks such as text classification, named entity recognition, sentiment analysis, and semantic similarity. Through practical exercises and projects, participants will gain valuable experience in solving NLP problems and analyzing results.

Frameworks and Tools Covered:

- NLTK (Natural Language Toolkit): Library for building NLP applications, providing tools for text processing, tokenization, syntactic analysis, and semantic analysis.

Top Skills Covered:

- Text Classification: Classifying text documents into predefined categories or labels using machine learning algorithms.
- Named Entity Recognition: Identifying and extracting named entities such as persons, organizations, and locations from text data.
- Sentiment Analysis: Analyzing the sentiment or emotion expressed in text documents using lexicon-based or machine learning approaches.
- Semantic Similarity: Measuring the semantic similarity between words or sentences based on their meanings using word embeddings or semantic models.

Topic 25

Advanced NLP

Take your Natural Language Processing (NLP) skills to the next level with our Advanced NLP module. Participants will explore advanced NLP techniques such as topic modeling, text summarization, and machine translation using Python's NLP libraries. Through hands-on exercises, participants will learn to tackle complex NLP tasks and challenges.

Frameworks and Tools Covered:

- NLTK (Natural Language Toolkit): Library for building NLP applications, providing tools for text processing, syntactic analysis, semantic analysis, and advanced NLP techniques.

Top Skills Covered:

- Topic Modeling: Discovering hidden topics or themes in text collections using probabilistic models such as Latent Dirichlet Allocation (LDA).
- Text Summarization: Generating concise summaries of long documents or articles using extractive or abstractive methods.
- Machine Translation: Translating text from one language to another using neural machine translation models such as sequence-to-sequence models.
- Advanced NLP Applications: Applying advanced NLP techniques to tasks such as information extraction, question answering, and dialogue systems.

About Gen AI

Generative AI (Gen AI) is a revolutionary branch of artificial intelligence pushing the boundaries of what machines can do. It's not just about recognizing patterns; it's about enabling machines to generate entirely new content. Imagine AI composing music that rivals the greats, crafting paintings that captivate the art world, or even writing code that automates complex tasks. Gen AI is blurring the lines between human and machine creativity, opening doors to a future of endless possibilities.

Topic 26

Fundamentals of Gen AI

Embark on a journey into the realm of Generative Artificial Intelligence (Gen AI) with our Fundamentals module. Participants will explore the principles, architectures, and applications of generative models such as Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs). Through theoretical insights and practical exercises, participants will gain a foundational understanding of Gen AI.

Frameworks and Tools Covered:

- TensorFlow: Open-source machine learning framework developed by Google for building and training neural networks and other machine learning models.
- Keras: High-level neural networks API for building and training deep learning models, built on top of TensorFlow.

Top Skills Covered:

- Generative Models: Understanding the principles and architectures of generative models for synthesizing new data samples.
- Generative Adversarial Networks (GANs): Learning concepts & components of GANs, including generators, discriminators, and adversarial training.
- Variational Autoencoders (VAEs): Exploring the theory and applications of VAEs for learning latent representations and generating new data samples.
- Applications of Gen AI: Discovering real-world applications of generative models in domains such as image generation, text generation, and data augmentation.

Topic 27

Prompt Engineering

Uncover the art and science of prompt engineering with our dedicated module. Participants will learn to craft effective prompts for Generative Artificial Intelligence (Gen AI) models, guiding model behavior and output generation. Through practical exercises and case studies, participants will develop the skills to design prompts that produce desired outcomes and control model behavior.

Frameworks and Tools Covered:

- OpenAI Codex: Large-scale language model developed by OpenAI for natural language understanding and generation tasks.
- GPT-3: State-of-the-art language model developed by OpenAI for generating human-like text based on input prompts.

Top Skills Covered:

- Prompt Formulation: Crafting clear, concise, and effective prompts to guide Gen AI model behavior and output generation.
- Prompt Design Strategies: Exploring different prompt design strategies such as completion prompts, conditional prompts, and fine-tuning prompts.
- Controlling Model Output: Learning techniques to control and manipulate Gen AI model output using prompt engineering.
- Ethical Considerations: Understanding ethical implications and considerations in prompt design and model use.

Topic 28

LLM models

Delve into the world of Large Language Models (LLMs) with our dedicated module. Participants will explore the architecture, capabilities, and applications of LLMs such as GPT-3. Through theoretical insights and practical exercises, participants will gain a deep understanding of LLMs and their potential to transform various industries and domains.

Frameworks and Tools Covered:

- OpenAI GPT-3: State-of-the-art large language model developed by OpenAI for generating human-like text based on input prompts.

Top Skills Covered:

- Large Language Models: Understanding the architecture and capabilities of large language models such as GPT-3.
- Text Generation: Exploring the capabilities of LLMs for generating human-like text across various domains and topics.
- Fine-Tuning: Learning techniques to fine-tune pre-trained LLMs for specific tasks and domains using domain-specific data.
- Applications of LLMs: Discovering real-world applications of LLMs in domains such as natural language understanding, text generation, and conversational AI.

Topic 29

Langchain for GenAI

Unleash the power of Language Chains for Generative Artificial Intelligence (Gen AI) with our dedicated module. Participants will learn to construct and train Language Chains, a novel architecture for guiding and controlling text generation. Through theoretical insights and practical exercises, participants will gain the skills to design and implement Language Chains for diverse applications.

Frameworks and Tools Covered:

- Python: Programming language used for implementing Language Chains and training Gen AI models.
- TensorFlow: Open-source machine learning framework developed by Google for building and training neural networks and other machine learning models.

Top Skills Covered:

- Language Chains: Understanding the architecture and principles of Language Chains for guiding and controlling text generation.
- Chain Construction: Learning techniques to construct and train Language Chains using Python and TensorFlow.
- Model Integration: Integrating Language Chains with existing Generative Artificial Intelligence (Gen AI) models for enhanced control and guidance.
- Applications of Language Chains: Exploring diverse applications of Language Chains in domains such as creative writing, content generation, and conversational AI.

Topic 30

Vector databases for GenAI

Unlock the potential of Vector Databases for Generative Artificial Intelligence (Gen AI) with our dedicated module. Participants will learn to leverage Vector Databases for storing, querying, and manipulating high-dimensional vectors representing textual data. Through theoretical insights and practical exercises, participants will gain the skills to design and implement Vector Databases for Gen AI applications.

Frameworks and Tools Covered:

- Faiss: Efficient similarity search and clustering library developed by Facebook AI Research for handling large-scale high-dimensional vectors.
- Annoy: Approximate nearest neighbor library for high-dimensional vector indexing and search.

Top Skills Covered:

- Vector Databases: Understanding the architecture and capabilities of Vector Databases for storing and querying high-dimensional vectors.
- Similarity Search: Learning techniques to perform similarity search and retrieval of vectors based on their similarity to a query vector.
- Indexing and Querying: Exploring methods to index and query large-scale vector databases efficiently for Gen AI applications.
- Applications of Vector Databases: Leveraging Vector Databases for diverse applications such as text search, recommendation systems, and content retrieval.

Topic 31

GenAI Assignment

Challenge your skills and creativity with our hands-on GenAI assignment. In this module, participants will apply advanced concepts and techniques in Generative Artificial Intelligence (Gen AI) to real-world tasks and scenarios. They will work on tasks such as text generation, image generation, and creative writing using state-of-the-art Gen AI models. Through practical exercises and projects, participants will gain valuable experience in designing and implementing Gen AI solutions.

Frameworks and Tools Covered:

- OpenAI GPT-3: State-of-the-art large language model developed by OpenAI for generating human-like text based on input prompts.
- TensorFlow: Open-source machine learning framework developed by Google for building and training neural networks and other machine learning models.

Top Skills Covered:

- Creative Problem Solving: Applying creative thinking and problem-solving skills to design and implement Gen AI solutions for real-world tasks.
- Model Fine-Tuning: Fine-tuning pre-trained Gen AI models for specific tasks and domains using domain-specific data and prompts.
- Ethical Considerations: Understanding ethical implications & considerations in designing and deploying Gen AI solutions for diverse applications.
- Presentation and Communication: Effectively presenting and communicating Gen AI solutions to stakeholders and end-users through written reports, presentations, and demos.

CLEVERA
THRIVE THROUGH CLEVERNESS

About Clevera

Clevera: Where Innovation Meets Expertise. Your one-stop platform for empowering your business through technology and expertise. We are a passionate team of trailblazers, dedicated to empowering businesses and individuals to thrive in the dynamic digital landscape. In 2024, Clevera emerged with a bold vision: to bridge the ever-widening gap between the skills demanded by corporations and the training provided by traditional institutions. Recognizing the rapid evolution of the digital landscape and the constant need for individuals to upskill, we set out to create a training paradigm that seamlessly connects these two worlds.

Each module is designed to provide comprehensive coverage of its respective topic, equipping participants with practical skills and theoretical knowledge essential for success in today's competitive landscape. Whether you're a beginner looking to explore new technologies or an experienced professional seeking to enhance your expertise, our courses offer something for everyone. Join us on a journey of learning and discovery, and unlock your full potential with Clevera Software Training Institute.

Ready to Code The World?

CLEVERA
THRIVE THROUGH CLEVERNESS

Bangalore office

Clevera

438/E, 3rd Floor, HMT Main Road
(Above Prestige Showroom, Opp to
Tender Chicken) MATHIKERE,
Karnataka, Bangalore, India 560054

Call us at: **7406 98 94 94**

www.clevera.in

7406 97 94 94