

1. Foundational Understanding (Days 1–30)

Objective: Build a solid grounding in NLP, AI fundamentals, and the underlying technologies powering Generative AI.

- **Key Topics:**
 - Basics of Natural Language Processing (NLP) and text preprocessing.
 - Fundamentals of machine learning and deep learning.
 - Introduction to key tools: Python, TensorFlow, PyTorch, Hugging Face.
 - Core models: n-grams, word embeddings (Word2Vec, GloVe), TF-IDF.
 - Basics of evaluation metrics and exploratory data analysis.
 - **Outcomes:**
 - Understand the essential concepts and use cases of NLP.
 - Ability to preprocess and explore text datasets.
 - Familiarity with ML libraries and tools used for NLP and AI.
-

Phase I: Foundational Understanding (Days 1–30)

Week 1: NLP Basics and Preprocessing

- **Objective:** Gain a foundational understanding of Natural Language Processing (NLP) and text preprocessing techniques.
- **Day 1:** Introduction to NLP
 - **Topics:**
 - Overview of Natural Language Processing (NLP).
 - Real-world applications: sentiment analysis, chatbots, machine translation, and information retrieval.
 - Challenges in NLP: ambiguity, data sparsity, and context understanding.
 - **Tasks:**
 - Read introductory materials from *Speech and Language Processing* by Jurafsky and Martin.
 - Install essential libraries (nltk, spaCy, scikit-learn).
 - Perform tokenization using nltk on sample text data.

- **Day 2: Text Preprocessing Techniques**
 - **Topics:**
 - Importance of text preprocessing.
 - Techniques: tokenization, stopwords removal, stemming, and lemmatization.
 - Differences between stemming and lemmatization.
 - **Tasks:**
 - Preprocess raw text using Python libraries (nltk and spaCy).
 - Write a custom pipeline to clean text and compare the results of stemming vs. lemmatization.
- **Day 3: POS Tagging and Named Entity Recognition (NER)**
 - **Topics:**
 - Part-of-Speech tagging: assigning grammatical labels to words.
 - Named Entity Recognition: identifying proper nouns and entities (e.g., names, dates, locations).
 - Applications of POS tagging and NER in text analysis and search systems.
 - **Tasks:**
 - Use spaCy to perform POS tagging and NER on a news article.
 - Analyze the accuracy of tags and entities.
- **Day 4: Representing Text Data**
 - **Topics:**
 - Representing text as vectors: Bag of Words and TF-IDF (Term Frequency-Inverse Document Frequency).
 - Strengths and limitations of these representations.
 - **Tasks:**
 - Convert a small text corpus into a Bag of Words and TF-IDF representation using scikit-learn.
 - Visualize word frequencies and TF-IDF scores.

- **Day 5: Working with N-grams**
 - **Topics:**
 - Understanding n-grams: sequences of words.
 - Applications of n-grams in language modeling and text prediction.
 - **Tasks:**
 - Implement unigram, bigram, and trigram models with Python.
 - Compare the results on a dataset, highlighting the predictive power of different n-grams.
 - **Day 6: Exploratory Data Analysis (EDA) for Text**
 - **Topics:**
 - Importance of EDA in understanding text datasets.
 - Common EDA techniques for text: word clouds, word frequencies, and co-occurrence analysis.
 - **Tasks:**
 - Analyze a sample text dataset (e.g., tweets or product reviews).
 - Visualize the top 20 most frequent words and bigrams using matplotlib or seaborn.
 - **Day 7: Project Day**
 - **Task:**
 - Apply the concepts learned in Week 1 to preprocess, analyze, and visualize insights from a real-world text dataset (e.g., IMDb reviews).
 - **Deliverable:**
 - A report showcasing the preprocessing pipeline and key insights.
-

Week 2: Language Models and Word Embeddings

- **Objective:** Understand the evolution of language models and explore word embeddings for representing text data.
- **Day 8: Introduction to Language Models**
 - **Topics:**
 - Basics of language models: unigrams, bigrams, and n-grams.
 - Limitations of traditional models in capturing context.
 - **Tasks:**
 - Implement an n-gram language model for text prediction.
 - Analyze performance on small text datasets.

- **Day 9: Word Embeddings (Word2Vec and GloVe)**
 - **Topics:**
 - Dense vector representations for words.
 - Key ideas: capturing semantic similarity using embeddings.
 - **Tasks:**
 - Train Word2Vec on a small dataset using gensim.
 - Visualize word embeddings using dimensionality reduction techniques like PCA.
- **Day 10: Sentence Embeddings**
 - **Topics:**
 - Moving beyond word embeddings to sentence-level representations.
 - Applications in text similarity and clustering.
 - **Tasks:**
 - Use sentence-transformers to compute embeddings for text documents.
 - Compare the similarity between sentences using cosine similarity.
- **Day 11: Measuring Text Similarity**
 - **Topics:**
 - Metrics for comparing text representations: cosine similarity, Euclidean distance.
 - **Tasks:**
 - Build a text similarity tool to compare user queries with a set of pre-existing FAQs.
- **Day 12: Feature Engineering for Text Classification**
 - **Topics:**
 - Combining statistical and embedding-based features for text classification tasks.
 - Selecting meaningful features to improve model performance.
 - **Tasks:**
 - Train a simple text classification model using TF-IDF features combined with custom text features (e.g., length of text).

- **Day 13: Case Study: Building a Simple Chatbot**
 - **Scenario:**
 - Create a rule-based chatbot using pre-trained embeddings for matching user queries to pre-defined responses.
 - **Tasks:**
 - Implement a simple chatbot capable of retrieving the best match for user questions.
 - **Day 14: Project Day**
 - **Task:**
 - Deliver a working prototype of a basic chatbot or FAQ retrieval tool using sentence embeddings.
 - **Deliverable:**
 - Demonstrate chatbot capabilities with example user queries.
-

Week 3: Machine Learning Basics for NLP

- **Objective:** Learn foundational machine learning techniques applied to NLP tasks.
- **Day 15: Introduction to Text Classification**
 - **Topics:**
 - Common NLP classification tasks: spam detection, sentiment analysis.
 - Workflow: preprocessing, feature extraction, model training.
 - **Tasks:**
 - Train a Naive Bayes classifier for spam detection on an email dataset using scikit-learn.
- **Day 16: Evaluation Metrics**
 - **Topics:**
 - Evaluating model performance: precision, recall, F1-score.
 - Trade-offs between precision and recall.
 - **Tasks:**
 - Evaluate the spam classifier using classification metrics.
 - Interpret precision-recall curves.

- **Day 17: Linear Models for Text Classification**
 - **Topics:**
 - Using Logistic Regression and Support Vector Machines (SVMs) for text classification.
 - Pros and cons of linear models in NLP.
 - **Tasks:**
 - Train a logistic regression model to classify movie reviews into positive or negative categories.
- **Day 18: Feature Selection and Dimensionality Reduction**
 - **Topics:**
 - Identifying relevant features to enhance model performance.
 - Techniques: chi-square tests, PCA.
 - **Tasks:**
 - Apply feature selection techniques on a text dataset.
- **Day 19: Hyperparameter Tuning**
 - **Topics:**
 - Optimizing ML models with grid search and random search.
 - Tuning hyperparameters for text classification models.
 - **Tasks:**
 - Perform hyperparameter tuning for an SVM classifier.
- **Day 20: Real-World Application**
 - **Task:**
 - Build a complete classification pipeline for a real-world dataset (e.g., Yelp reviews).
 - **Deliverable:**
 - Report summarizing model performance and key insights.
- **Day 21: Project Day**
 - **Task:**
 - Consolidate Week 3 concepts into an end-to-end text classification system.
 - **Deliverable:**
 - A polished classification pipeline with evaluation results.

Week 4: Introduction to Deep Learning for NLP

- **Objective:** Explore deep learning techniques and their applications in NLP.
- **Day 22: Basics of Neural Networks**
 - **Topics:**
 - Components of a neural network: layers, activations, loss functions.
 - How neural networks learn from data.
 - **Tasks:**
 - Implement a simple neural network for text classification using PyTorch or TensorFlow.
- **Day 23: Recurrent Neural Networks (RNNs)**
 - **Topics:**
 - Sequential data and the architecture of RNNs.
 - Applications in text generation and sequence modeling.
 - **Tasks:**
 - Train an RNN to generate text from a small corpus.
- **Day 24: Long Short-Term Memory (LSTM) Networks**
 - **Topics:**
 - Addressing vanishing gradients with LSTMs.
 - Applications in NLP tasks requiring long-term dependencies.
 - **Tasks:**
 - Train an LSTM for sentiment analysis on the IMDB dataset.
- **Day 25: Sequence-to-Sequence (Seq2Seq) Models**
 - **Topics:**
 - Architectures for Seq2Seq models in machine translation and summarization.
 - **Tasks:**
 - Implement a Seq2Seq model for text summarization.
- **Day 26: Attention Mechanisms**
 - **Topics:**
 - Enhancing Seq2Seq models with attention mechanisms.
 - How attention improves performance by focusing on relevant parts of the input.
 - **Tasks:**
 - Add an attention layer to a Seq2Seq model.
- **Day 27: Transformers: The New Standard**

- **Topics:**
 - Revolutionary transformer architecture and its impact on NLP.
 - Use cases of pre-trained transformers like BERT and GPT.
 - **Tasks:**
 - Explore Hugging Face's transformers library to fine-tune a BERT model.
 - **Day 28: Fine-Tuning Pre-Trained Models**
 - **Topics:**
 - Transfer learning and domain-specific fine-tuning.
 - **Tasks:**
 - Fine-tune a BERT model for text classification.
 - **Day 29: Ethics in AI**
 - **Topics:**
 - Addressing bias, fairness, and ethical concerns in AI systems.
 - **Tasks:**
 - Evaluate a pre-trained model for biases.
 - Discuss strategies for mitigating fairness issues.
 - **Day 30: Recap and Review**
 - **Task:**
 - Summarize Phase I concepts.
 - Refactor and improve previous projects for scalability and performance.
-

Outcomes of Phase I

By the end of Phase I, you will:

- Understand the foundational concepts of NLP and machine learning.
- Be proficient in preprocessing and analyzing text data.
- Build and evaluate machine learning models for text classification.
- Gain exposure to deep learning techniques and transformers for NLP.
- Develop small-scale projects showcasing your skills.

2. Intermediate Skills and Model Training (Days 31–60)

Objective: Develop practical skills in building and fine-tuning models, leveraging transformers, and understanding generative models.

- **Key Topics:**
 - Recurrent Neural Networks (RNNs), LSTMs, GRUs, and sequence models.
 - Introduction to transformers: architecture and applications (BERT, GPT, etc.).
 - Fine-tuning and transfer learning with pretrained models.
 - Generative models: VAEs, GANs, and their role in text/image synthesis.
 - Advanced NLP tasks: summarization, question answering, sentiment analysis.
 - Fundamentals of model evaluation and hyperparameter tuning.
 - **Outcomes:**
 - Ability to train and fine-tune deep learning models for NLP tasks.
 - Practical understanding of transformer-based models and their customization.
 - Exposure to generative models and their potential in creating synthetic content.
-

Week 5: Deep Dive into Advanced Language Models (Days 31–37)

- **Day 31: Fundamentals of RNN Extensions**
 - **Topics:**
 - Understanding GRUs (Gated Recurrent Units) and their differences from LSTMs.
 - Applications of GRUs in NLP tasks.
 - **Tasks:**
 - Train a GRU-based model for sequence prediction tasks (e.g., stock price prediction or time-series text tasks).
- **Day 32: Advanced Sequence Models**
 - **Topics:**
 - Revisiting Seq2Seq models and their applications in translation and text generation.
 - Sequence evaluation metrics: BLEU and ROUGE scores.
 - **Tasks:**
 - Train a Seq2Seq model for English-to-French translation using TensorFlow.
 - Evaluate the model using BLEU scores.

- **Day 33: Introduction to Attention Mechanisms**
 - **Topics:**
 - Deep dive into the concept of attention and self-attention.
 - Importance of attention mechanisms in improving model performance.
 - **Tasks:**
 - Add attention layers to a Seq2Seq model and analyze improvements in translation quality.
- **Day 34: Transformers: Revolutionizing NLP**
 - **Topics:**
 - Transformer architecture explained: encoder-decoder structure, multi-head attention, positional encodings.
 - How transformers differ from RNNs/LSTMs.
 - **Tasks:**
 - Read the "Attention Is All You Need" paper and summarize its key contributions.
 - Build a basic transformer from scratch using PyTorch.
- **Day 35: Introduction to Pretrained Transformers (BERT and GPT)**
 - **Topics:**
 - BERT: masked language modeling, token classification, fine-tuning.
 - GPT: autoregressive language modeling, text generation.
 - **Tasks:**
 - Use Hugging Face's transformers library to fine-tune BERT for a sentiment analysis task.
- **Day 36: Fine-Tuning GPT-2**
 - **Topics:**
 - Understanding GPT-2's architecture and pretraining objectives.
 - Use cases: text generation, creative writing, summarization.
 - **Tasks:**
 - Fine-tune GPT-2 on a dataset of product descriptions to generate marketing copy.

- **Day 37: Project Day**
 - **Task:**
 - Consolidate Week 5 concepts by building a custom text translation system using a transformer-based architecture.
 - **Deliverable:**
 - A detailed report including model architecture, performance evaluation, and example outputs.

Week 6: Generative Models and Applications (Days 38–44)

- **Day 38: Introduction to Generative Models**
 - **Topics:**
 - Overview of generative models in NLP.
 - Variational Autoencoders (VAEs) and their applications in text generation.
 - **Tasks:**
 - Implement a VAE to generate text snippets from a dataset.
- **Day 39: Generative Adversarial Networks (GANs) in NLP**
 - **Topics:**
 - How GANs work: generator and discriminator.
 - Challenges of using GANs for text.
 - **Tasks:**
 - Implement a basic GAN for text generation using PyTorch.
- **Day 40: Hybrid Approaches: GANs + Transformers**
 - **Topics:**
 - Exploring how GANs and transformers can be combined to improve generative tasks.
 - **Tasks:**
 - Research case studies on GAN-based text augmentation and build a simple implementation.

- **Day 41: Real-World Generative AI Applications**
 - **Topics:**
 - Use cases: automated report generation, text summarization, conversational AI.
 - **Tasks:**
 - Build a text summarization system using GPT and evaluate its performance.
 - **Day 42: Introduction to Multimodal Generative Models**
 - **Topics:**
 - Overview of models like CLIP and DALL-E.
 - Generating text-to-image content.
 - **Tasks:**
 - Use OpenAI's DALL-E or similar APIs to generate visual content from text.
 - **Day 43: Building End-to-End Generative AI Systems**
 - **Topics:**
 - Combining multiple generative tasks into a cohesive pipeline.
 - Use case: AI for automated content creation and publishing.
 - **Tasks:**
 - Build a pipeline to generate, summarize, and classify text using transformers.
 - **Day 44: Project Day**
 - **Task:**
 - Create an end-to-end generative AI system that summarizes articles and generates blog posts or creative text.
 - **Deliverable:**
 - A demonstration-ready generative system with documentation.
-

Week 7: Exploring Large Language Models (Days 45–51)

- **Day 45: Introduction to Large Language Models (LLMs)**
 - **Topics:**
 - Understanding LLMs like GPT-3, GPT-4, and their training methodologies.
 - Differences between traditional NLP models and LLMs.
 - **Tasks:**
 - Explore OpenAI's API for GPT-3 and generate text using predefined prompts.

- **Day 46: Fine-Tuning LLMs**
 - **Topics:**
 - Techniques for fine-tuning large models on specific tasks.
 - **Tasks:**
 - Fine-tune GPT-3 on a custom dataset for customer support queries.
- **Day 47: LLM Applications in Business**
 - **Topics:**
 - How LLMs are used in industries like healthcare, finance, and e-commerce.
 - Ethical considerations when deploying LLMs.
 - **Tasks:**
 - Research case studies of GPT-3 applications in real-world scenarios.
- **Day 48: Advanced Prompt Engineering**
 - **Topics:**
 - Crafting effective prompts for LLMs to perform complex tasks.
 - **Tasks:**
 - Experiment with multi-turn conversations using OpenAI's API to simulate intelligent dialogue.
- **Day 49: Case Study: GPT for Creative Writing**
 - **Scenario:**
 - Use GPT-3 to write creative pieces such as poetry, short stories, or marketing copy.
 - **Tasks:**
 - Build a tool that generates creative text based on user input themes.
- **Day 50: Deploying LLMs in Production**
 - **Topics:**
 - Deploying LLMs via APIs and managing costs.
 - **Tasks:**
 - Deploy a fine-tuned GPT model using a cloud service like AWS or Azure.
- **Day 51: Project Day**
 - **Task:**
 - Develop a creative writing or conversational AI system powered by GPT-3.
 - **Deliverable:**
 - A production-ready LLM-based application with deployment documentation.

Week 8: Ethics, Scalability, and Future Trends (Days 52–60)

- **Day 52: Ethics in Generative AI**
 - **Topics:**
 - Bias, fairness, and misuse of generative models.
 - Mitigation strategies for responsible AI development.
 - **Tasks:**
 - Analyze potential biases in an LLM-generated dataset and suggest solutions.
- **Day 53: Scalability in Generative AI Systems**
 - **Topics:**
 - Challenges in training and deploying large-scale models.
 - Distributed computing and optimization techniques.
 - **Tasks:**
 - Use TensorFlow Distributed Strategy to train a model on multiple GPUs.
- **Day 54: AI Alignment and Safety**
 - **Topics:**
 - Ensuring alignment between AI systems and human values.
 - **Tasks:**
 - Research current advancements in AI alignment techniques.
- **Day 55: Emerging Trends in Generative AI**
 - **Topics:**
 - Foundation models and their growing influence.
 - Human-AI collaboration.
 - **Tasks:**
 - Explore papers on foundation models and their applications across modalities.
- **Day 56: Future of Agentic AI Systems**
 - **Topics:**
 - Transition from task-driven AI to autonomous, goal-oriented AI agents.
 - **Tasks:**
 - Implement a simple AI agent capable of multi-step reasoning using GPT and reinforcement learning.

- **Day 57: Case Study: Multimodal AI Systems**
 - **Scenario:**
 - Combine LLMs with vision models (e.g., CLIP) for multimodal tasks.
 - **Tasks:**
 - Build a system that generates captions for images using CLIP and GPT.
- **Day 58: Building a Portfolio of Projects**
 - **Topics:**
 - Documenting and presenting AI projects for career advancement.
 - **Tasks:**
 - Prepare a GitHub repository with comprehensive documentation for Phase II projects.
- **Day 59: Summary of Phase II**
 - **Task:**
 - Review and refine all Phase II projects.
 - Deliver a presentation-ready portfolio of projects.
- **Day 60: Final Capstone Project**

Task:

- Develop a comprehensive generative AI application incorporating everything learned in Phase II.

- **Deliverable:**
 - A polished capstone project with a detailed report, presentation slides, and deployment-ready code.

3. Advanced Applications and System Design (Days 61–80)

Objective: Explore large-scale applications of Generative AI and Agentic AI, focusing on integrating, deploying, and building intelligent systems.

- **Key Topics:**
 - Deep dive into Large Language Models (LLMs): GPT-3, GPT-4, and beyond.
 - Architectures for end-to-end AI systems: integrating NLP pipelines.
 - Exploring agentic AI systems: autonomous reasoning, task-solving agents.
 - Real-world applications: chatbots, content generation, multimodal systems.
 - Scalability: training LLMs on large datasets, distributed computing.
 - Ethics and bias in AI systems.
 - **Outcomes:**
 - Proficiency in deploying Generative AI solutions for complex tasks.
 - Understanding of agentic AI frameworks and their applications.
 - Awareness of ethical considerations in creating AI-driven systems.
-

Week 9: Large Language Models (Days 61–67)

- **Day 61: Understanding LLM Architectures**
 - **Topics:**
 - Evolution of Large Language Models: GPT-3, GPT-4, and beyond.
 - Challenges in training LLMs: compute requirements, data availability, and biases.
 - **Tasks:**
 - Research OpenAI's GPT-4 architecture and Google's PaLM model.
 - Explore Hugging Face's large-scale model repositories and training pipelines.
- **Day 62: Fine-Tuning and Adapting LLMs**
 - **Topics:**
 - Adapting LLMs to specific domains and tasks through fine-tuning and in-context learning.
 - **Tasks:**
 - Fine-tune GPT-3 on a domain-specific dataset (e.g., legal or healthcare documents).
 - Use zero-shot or few-shot learning for a niche NLP task (e.g., financial sentiment analysis).

- **Day 63: Memory-Augmented LLMs**
 - **Topics:**
 - Adding memory capabilities to LLMs for multi-turn conversations.
 - Techniques for knowledge grounding and retrieval-augmented generation (RAG).
 - **Tasks:**
 - Build a chatbot capable of retrieving facts from a knowledge base (e.g., Wikipedia) using LLMs.
- **Day 64: Using Reinforcement Learning for Fine-Tuning**
 - **Topics:**
 - Reinforcement Learning with Human Feedback (RLHF) for aligning LLM behavior with user expectations.
 - **Tasks:**
 - Read OpenAI's paper on RLHF and implement a basic RLHF pipeline for text generation.
- **Day 65: Case Study: Business Applications of LLMs**
 - **Scenario:**
 - Explore applications like automated customer support, document summarization, and conversational assistants.
 - **Tasks:**
 - Develop a prototype chatbot that uses GPT-3 to assist with customer queries and FAQs.
- **Day 66: Evaluation Metrics for LLMs**
 - **Topics:**
 - Metrics for assessing LLM performance: perplexity, BLEU, ROUGE, and human evaluation.
 - **Tasks:**
 - Compare the output quality of GPT-3 vs. a fine-tuned smaller model on a summarization task.
- **Day 67: Project Day**
 - **Task:**
 - Build a domain-specific conversational agent using GPT-3 or GPT-4 for a business use case (e.g., e-commerce or travel).
 - **Deliverable:**
 - A chatbot with robust, context-aware capabilities for handling multi-turn dialogues.

Week 10: Multimodal AI Systems (Days 68–74)

- **Day 68: Introduction to Multimodal AI**
 - **Topics:**
 - Combining vision, text, and audio modalities.
 - Overview of models like CLIP, DALL-E, and Flamingo.
 - **Tasks:**
 - Generate text-based image captions using OpenAI's CLIP.
- **Day 69: Text-to-Image Generation**
 - **Topics:**
 - Techniques for text-to-image synthesis: GANs, diffusion models, and DALL-E.
 - **Tasks:**
 - Use DALL-E to generate images from descriptive text prompts.
- **Day 70: Image Captioning with Multimodal AI**
 - **Topics:**
 - How image-text alignment works in models like CLIP.
 - **Tasks:**
 - Build a system that generates captions for uploaded images using pre-trained models.
- **Day 71: Multimodal Search Engines**
 - **Topics:**
 - Integrating text and image search in a single pipeline.
 - **Tasks:**
 - Develop a search engine that retrieves relevant images based on textual input using CLIP.
- **Day 72: Case Study: Multimodal Applications in Healthcare**
 - **Scenario:**
 - Explore how multimodal AI is used for diagnosing diseases, summarizing medical images, or integrating text and visuals in reports.
 - **Tasks:**
 - Design a prototype that matches medical images with diagnostic descriptions using multimodal AI.

- **Day 73: Ethics in Multimodal AI**
 - **Topics:**
 - Challenges of bias and fairness in multimodal systems.
 - Use cases that raise ethical concerns: deepfakes, misinformation.
 - **Tasks:**
 - Analyze the ethical implications of a multimodal project and suggest mitigation strategies.
- **Day 74: Project Day**
 - **Task:**
 - Build a multimodal application, such as a visual search tool or image-to-text pipeline for accessibility.
 - **Deliverable:**
 - A working multimodal AI system with documentation and an ethical assessment.

Week 11: Scalability and Deployment of Generative AI Systems (Days 75–80)

- **Day 75: Challenges in Scaling Generative AI**
 - **Topics:**
 - Compute and storage requirements for training and deploying large models.
 - Distributed systems for model training and inference.
 - **Tasks:**
 - Use TensorFlow or PyTorch to train a model using distributed computing.
- **Day 76: Deploying Generative AI Models**
 - **Topics:**
 - Serving AI models via REST APIs and cloud platforms.
 - Scaling inference for high-traffic applications.
 - **Tasks:**
 - Deploy a fine-tuned GPT model on AWS or GCP.

- **Day 77: Optimization Techniques for Large Models**
 - **Topics:**
 - Reducing memory and compute overhead with techniques like quantization and pruning.
 - **Tasks:**
 - Optimize a pre-trained transformer model using model quantization and compare its performance.
- **Day 78: Real-World Deployment Challenges**
 - **Topics:**
 - Handling user privacy, data compliance, and security in generative AI systems.
 - **Tasks:**
 - Design a secure and compliant architecture for deploying a chatbot handling sensitive user data.
- **Day 79: Case Study: Scalable Chatbot Deployment**
 - **Scenario:**
 - Deploy a scalable, interactive chatbot for a customer service use case.
 - **Tasks:**
 - Monitor chatbot performance under different loads and fine-tune it for speed and accuracy.
- **Day 80: Final Capstone Project for Phase III**
 - **Task:**
 - Develop and deploy a multimodal AI system capable of generating text, images, and handling multi-turn conversations.
 - **Deliverable:**
 - A scalable, deployed system with detailed documentation, source code, and a live demo link.

4. Leadership and Innovation in Generative AI (Days 81–90)

Objective: Transition from practitioner to thought leader, mastering the art of innovation and strategic decision-making in Generative AI.

- **Key Topics:**
 - Advanced topics: multimodal AI (e.g., DALL-E, CLIP), reinforcement learning in AI agents.
 - Designing and leading AI projects: project management, stakeholder alignment.
 - Emerging trends: foundation models, AI alignment, human-AI collaboration.
 - Practical insights into productizing Generative AI systems.
 - Networking and collaboration: participating in AI communities and research.
 - **Outcomes:**
 - Ability to lead AI-driven projects from ideation to deployment.
 - Strategic understanding of Generative AI's role in business and technology.
 - A portfolio of projects showcasing expertise in GenAI and Agentic AI.
-

Week 12: Mastery in Generative AI and Agentic Systems (Days 81–85)

- **Day 81: Designing Agentic AI Systems**
 - **Topics:**
 - Introduction to agentic AI: autonomy, multi-agent systems, and task orchestration.
 - Building goal-driven agents using reinforcement learning and LLMs.
 - **Tasks:**
 - Implement a task-solving agent capable of multi-step reasoning using GPT-4 and reinforcement learning.
 - Use libraries like LangChain or Haystack to create agents for task orchestration.
- **Day 82: Building AI-Driven Business Strategies**
 - **Topics:**
 - Role of Generative AI in transforming business processes.
 - Identifying key areas for AI deployment: customer support, content generation, operational efficiency.
 - **Tasks:**
 - Research real-world case studies of AI-driven business success.
 - Design a strategy document for using generative AI to enhance a specific industry (e.g., finance, e-commerce, or healthcare).

- **Day 83: Trends in Generative and Agentic AI**
 - **Topics:**
 - Future advancements: foundation models, multimodal agents, AI-human collaboration.
 - Open challenges in scaling, alignment, and regulation of AI systems.
 - **Tasks:**
 - Read cutting-edge research papers (e.g., OpenAI's GPT-4 technical report or Google's Pathways model).
 - Identify emerging opportunities for AI applications in your area of expertise.
- **Day 84: Ethics, Compliance, and AI Governance**
 - **Topics:**
 - Ethical considerations for large-scale AI deployment: bias mitigation, data privacy, and compliance.
 - Role of AI governance in ensuring responsible use.
 - **Tasks:**
 - Develop an ethical checklist for deploying a generative AI system.
 - Create a compliance strategy aligning with GDPR, CCPA, or other regulations.
- **Day 85: Portfolio Development and Thought Leadership**
 - **Topics:**
 - Creating a professional portfolio: documenting projects, writing blogs, and presenting work.
 - Engaging in the AI community through conferences, meetups, and open-source contributions.
 - **Tasks:**
 - Publish a detailed blog post on your generative AI capstone project.
 - Refine your GitHub repository with comprehensive documentation for all previous projects.

Week 13: Capstone and Professional Showcase (Days 86–90)

- **Day 86: Defining the Final Capstone Project**
 - **Task:**
 - Scope out an advanced, real-world project that integrates generative AI, agentic systems, and multimodal capabilities.
 - Example Project: “**Autonomous Multimodal Assistant for Business Optimization**” – A system that combines GPT-4 for conversation, CLIP for image processing, and a task orchestration agent for operational insights.
- **Day 87: Building the Capstone Project**
 - **Task:**
 - Begin implementing the capstone project:
 - Train/fine-tune models as required.
 - Integrate multimodal capabilities and task automation.
- **Day 88: Deployment and Evaluation**
 - **Task:**
 - Deploy the capstone project on a cloud platform with robust APIs for interaction.
 - Test the system extensively and evaluate its performance using predefined metrics.
- **Day 89: Finalizing the Capstone Project**
 - **Tasks:**
 - Prepare documentation, user guides, and presentation slides for your capstone project.
 - Record a demo showcasing the system’s functionality and key features.
- **Day 90: Showcase and Presentation**
 - **Task:**
 - Deliver a presentation summarizing your 90-day journey, with a deep dive into your final capstone project.
 - Share your portfolio with peers, mentors, or prospective employers.

Use Cases

Phase I: Foundational Understanding (Days 1–30)

Focus on mastering basic NLP concepts, preprocessing, and foundational machine learning.

Use Cases:

1. Sentiment Analysis on Product Reviews:

- Analyze customer reviews to determine sentiment (positive, negative, or neutral).
- Tools: nltk, scikit-learn, and spaCy.
- Deliverable: Sentiment classification model and dashboard for insights.

2. FAQ Bot for Customer Support:

- Build a simple chatbot that matches user queries to predefined answers using TF-IDF and cosine similarity.
- Tools: Python, sentence-transformers.
- Deliverable: A prototype chatbot for a specific domain like retail or banking.

3. Text Summarization for News Articles:

- Implement a rule-based or basic machine learning model to generate summaries of news articles.
- Tools: nltk and spaCy.
- Deliverable: An automated summarization tool for headlines.

4. Spam Detection for Emails:

- Train a Naive Bayes classifier to distinguish spam emails from legitimate ones using word frequencies.
- Tools: scikit-learn.
- Deliverable: Spam detection pipeline with evaluation metrics.

5. Text Similarity for Legal Document Matching:

- Create a system to find similar legal clauses across contracts using embeddings.
- Tools: sentence-transformers.
- Deliverable: A similarity tool that ranks similar clauses.

Phase II: Intermediate Skills and Model Training (Days 31–60)

Dive into advanced models like transformers, fine-tuning, and generative techniques.

Use Cases:

1. Domain-Specific Chatbot with GPT-2:

- Fine-tune GPT-2 to handle specific tasks, such as answering questions in healthcare or education.
- Tools: Hugging Face transformers.
- Deliverable: Chatbot capable of answering domain-specific queries with high accuracy.

2. Text-to-Image Generation for Marketing:

- Use DALL-E to generate images based on textual descriptions, such as ad visuals or product mockups.
- Tools: OpenAI API.
- Deliverable: A text-to-image tool for marketing campaigns.

3. Summarization for Financial Reports:

- Fine-tune BERT or GPT models to extract summaries from financial statements.
- Tools: Hugging Face transformers.
- Deliverable: A summarization model that produces concise, domain-specific insights.

4. Multimodal Caption Generator:

- Build a system that generates captions for uploaded images using models like CLIP or Vision Transformers.
- Tools: Hugging Face, OpenAI APIs.
- Deliverable: An image-captioning pipeline for accessibility or e-commerce.

5. Generative Adversarial Network (GAN) for Text Augmentation:

- Use GANs to generate synthetic data for NLP tasks, such as creating new tweets or customer reviews.
- Tools: PyTorch.
- Deliverable: Synthetic dataset generation pipeline to augment training data.

Phase III: Advanced Applications and System Design (Days 61–80)

Integrate large-scale applications, multimodal systems, and agentic AI features.

Use Cases:

1. Conversational Agent for E-Commerce:

- Develop a chatbot that answers product-related queries and recommends items based on user input.
- Tools: GPT-4, retrieval-augmented generation (RAG).
- Deliverable: Deployed chatbot for e-commerce platforms.

2. Image-Based Product Search:

- Build a multimodal system that retrieves product details based on an uploaded image and textual queries.
- Tools: CLIP.
- Deliverable: Visual search engine for an e-commerce catalog.

3. AI-Powered Content Generator:

- Create a generative system to produce blog posts, newsletters, or marketing content based on topic inputs.
- Tools: GPT-3 or GPT-4.
- Deliverable: A content creation tool for marketing teams.

4. Knowledge-Enhanced Chatbot for Healthcare:

- Integrate a knowledge base with GPT-4 to answer healthcare-related queries.
- Tools: GPT, LangChain.
- Deliverable: A chatbot that combines conversational and knowledge retrieval capabilities.

5. Task Automation Agent for HR:

- Develop an AI agent that automates repetitive HR tasks, such as resume screening and candidate matching.
- Tools: GPT-4, reinforcement learning.
- Deliverable: A workflow automation tool for HR processes.

Phase IV: Leadership and Innovation in Generative AI (Days 81–90)

Lead and strategize AI solutions that impact businesses and society while showcasing your expertise.

Use Cases:

1. Autonomous Multimodal Assistant for Business Analytics:

- Combine GPT-4 for conversational insights, CLIP for analyzing visual dashboards, and task orchestration for real-time data reporting.
- Tools: GPT-4, CLIP, LangChain.
- Deliverable: A multimodal assistant that supports C-suite decision-making.

2. Creative AI for Personalized Marketing Campaigns:

- Build a system that generates personalized ad copy and visuals based on customer profiles and preferences.
- Tools: GPT-4, DALL-E.
- Deliverable: A marketing assistant for automated campaign creation.

3. Ethical AI Audit Tool:

- Develop a system that evaluates AI models for fairness, bias, and compliance with GDPR or CCPA regulations.
- Tools: Fairlearn, AI Fairness 360.
- Deliverable: An audit tool for ethical AI deployment.

4. Generative AI-Powered Education Platform:

- Create a platform that generates personalized learning content (text, videos, quizzes) for students based on their needs.
- Tools: GPT-4, multimodal models.
- Deliverable: A personalized learning assistant.

5. Agentic AI System for Supply Chain Optimization:

- Design an autonomous agent that predicts demand, optimizes inventory, and suggests logistics improvements.
- Tools: Reinforcement learning, LLMs.
- Deliverable: A goal-driven system for real-time supply chain management.