

AIML Internship Prep



6. AI/ML Intern
 Domains: Deep Learning / NLP / Computer Vision
 • Train models, run experiments, and apply AI solutions
 • Python, NumPy, Tensor Flow/PyTorch experience preferred

Help me prepare for this role considering I have 2 weeks and also I am final year AIML graduation student and know basics of all things but yes want to brush up things.

Also make me ready for clearing the online assesment round.

Also consider adding some real life scenario based things that can test my knowledge after preparation and also add all state of the techniques, principles being used in the today's market for making me Interview ready.

Great! With 2 weeks and your AIML background, we can prepare you for the AI/ML internship by building a focused plan around:

2-Week Roadmap

Week 1 – Core Revise + Hands-on

Day	Topics	Goals	Tasks
Day 1	Python + NumPy Refresher	Brush up syntax, operations	Practice via Exercism / [LeetCode Easy Python]
Day 2	Pandas + Matplotlib	Data wrangling & visualization	Kaggle Titanic dataset EDA
Day 3	ML Fundamentals (Sklearn)	Regression, Classification	Implement Logistic Regression, Decision Tree on Sklearn datasets
Day 4	Deep Learning Basics (TensorFlow/Keras)	Feedforward NN, Activation, Loss	Build MNIST classifier
Day 5	PyTorch Intro	Tensors, Autograd, Training loop	CIFAR-10 or FashionMNIST model
Day 6	NLP Basics	Tokenization, Embeddings, RNN	Sentiment analysis with IMDB (use TorchText or HuggingFace)
Day 7	Computer Vision Basics	CNNs, Filters, Transfer Learning	Use ResNet for image classification (e.g. Dogs vs Cats)

Week 2 – Advanced + Scenario Practice

Day	Topics	Goals	Tasks
Day 8	End-to-End ML Project	Pipeline building	Build a pipeline for housing price prediction

Day	Topics	Goals	Tasks
Day 9	Real-time Projects	Vision/NLP/LLMs	Pick one real-life mini project (see below)
Day 10	Deployment Basics	Streamlit, Flask, Hugging Face Spaces	Deploy a simple model (e.g. sentiment classifier)
Day 11	Mock Assessment Practice	Timed MCQ + coding	Solve timed tests on InterviewBit , Hackerrank
Day 12	Scenario-based Case Study	Real application thinking	See below examples
Day 13	Interview Preparation	Behavioral + Tech Qs	Prepare answers to common ML intern Qs
Day 14	Final Revision & Project Presentation	Be confident	Review notes, projects, present one mini project

✓ Real-Life Scenario Practice

Use these to simulate thinking like an ML engineer:

- CV Task:** You're asked to build a defect detector in manufacturing using a ResNet-based model with limited data.
Q: How would you improve accuracy and reduce overfitting?
- NLP Task:** You're given noisy legal contracts to summarize and extract key clauses.
Q: Would you use transformers? Which model fits best?
- Data Issue:** Your model performs well on training data but fails miserably on production.
Q: What steps do you take to debug?
- Team Scenario:** You're asked to explain a deep learning model's prediction to a non-tech manager.
Q: How would you explain it?

🔍 Online Assessment Focus Areas

Expect:

- **Python/Numpy-based coding MCQs**
- **Output prediction (e.g., what does this tensor operation return?)**
- **ML/NN theory:** overfitting, activation functions, optimizers
- **Short coding problem:** matrix ops, classification score calc, etc.

Practice Platforms:

- [Kaggle Notebooks](#)
- [Stratascratch](#)
- [Machine Learning MCQs Practice](#)
- [HackerRank Python & ML](#)

Core Topics to Be Interview-Ready

Area	Key Concepts
Deep Learning	CNNs, RNNs, Transformers, Attention, Dropout, BatchNorm
NLP	BERT, GPT, Tokenization, Fine-tuning, Text classification
Computer Vision	ResNet, Transfer Learning, Image Augmentation
ML	Overfitting, Bias-Variance, Regularization, Metrics (F1, AUC)
Python/Libs	NumPy, Pandas, Matplotlib, Sklearn, Torch, TensorFlow
Project Thinking	Problem framing, pipeline steps, evaluation, deployment

Mini Project Suggestions

Pick one to build in week 2:

1. **Resume Classifier** – NLP pipeline to filter resumes by skill match.
2. **Face Mask Detector** – CV + Transfer learning using MobileNet.
3. **Legal Clause Extractor** – Use spaCy or BERT to identify legal obligations.
4. **Tweet Sentiment Dashboard** – NLP model + Streamlit + Twitter API.

Would you like me to create a GitHub-ready folder structure + boilerplate code for one mini-project from this list?