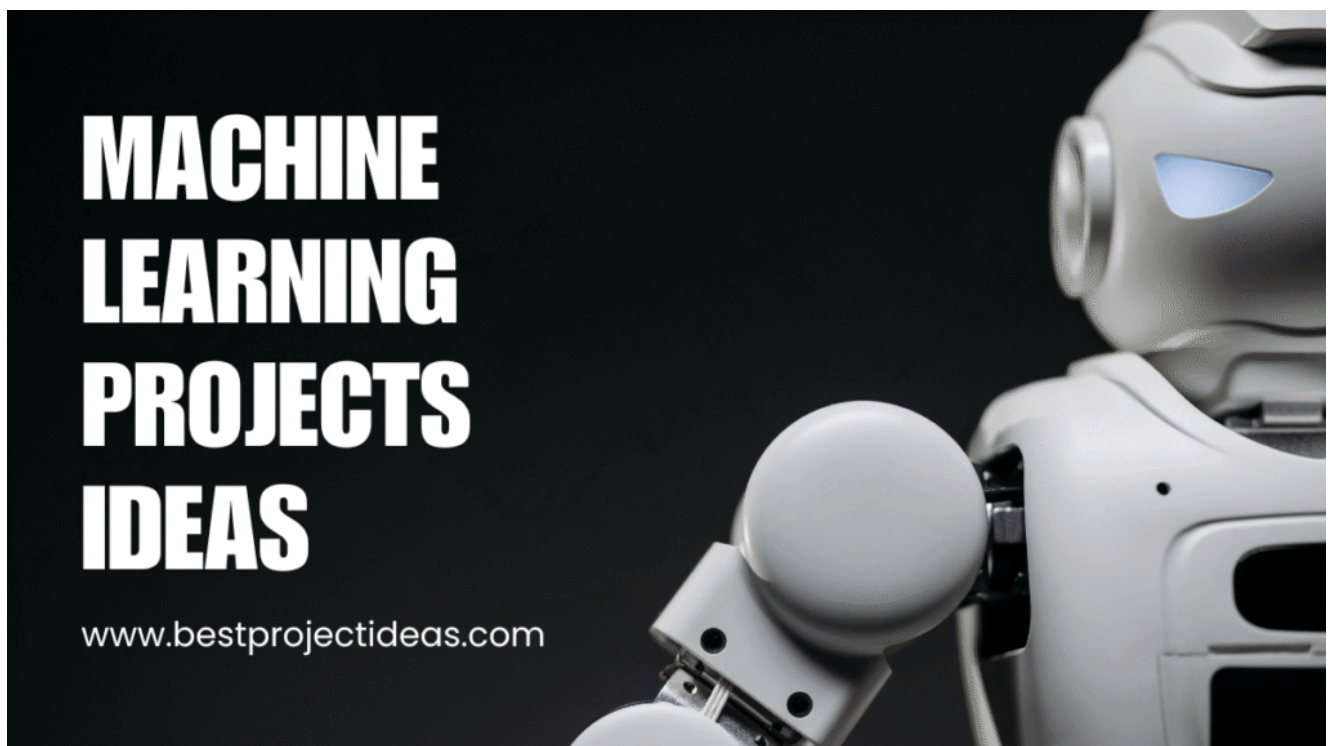


99+ Machine Learning Projects Ideas 2025-26

OCTOBER 28, 2025 | JOHN DEAR



If you are a student eager to learn machine learning, building projects is the fastest and most effective way to grow.

This article collects **100 Machine Learning Projects Ideas** designed for students at different levels — beginner, intermediate, and advanced. Each idea includes a short description, the difficulty level, suggested tools and datasets, a few key steps to implement it, and the learning outcomes you can expect.

Use these projects to practice coding, understand algorithms, prepare for interviews, and create a portfolio that shows real skills.

Before you start, remember: a good project focuses on a real problem, has a clear dataset, and shows progression — from data exploration to modeling to evaluation and deployment. Choose projects that interest you, scale complexity gradually, and document your work (readme, notebooks, short demo). Now let's dive into the 100 project ideas.

Also Read: [99+ Famous Profile Sekai Project Ideas For Students](#)

Table of Contents



How to use this list

1. Pick projects labeled **Beginner** if you are new to Python, Pandas, NumPy, and scikit-learn.
2. Choose **Intermediate** projects once you are comfortable with ML basics and want to try deep learning (TensorFlow / PyTorch).
3. Attempt **Advanced** projects when you want to build end-to-end systems, use large datasets, or deploy models to production.

Common tools and datasets across these projects:

- Languages & frameworks: Python, Jupyter Notebook, scikit-learn, TensorFlow, Keras, PyTorch, Flask/FastAPI (for deployment)
- Data handling: Pandas, NumPy, Matplotlib, Seaborn
- Natural language: NLTK, spaCy, Hugging Face transformers
- Computer vision: OpenCV, PIL
- Cloud & deployment: Heroku, AWS, GCP, Streamlit
- Datasets: UCI Machine Learning Repository, Kaggle, Google Dataset Search, OpenML

100 Machine Learning Projects Ideas 2025-26

1. Titanic Survival Prediction — *Beginner*

Description: Predict passenger survival using the Titanic dataset.

Tools / Dataset: Python, Pandas, scikit-learn, Kaggle Titanic dataset.

Key steps: data cleaning → feature engineering → model selection (logistic regression, random forest) → evaluation.

Learning outcomes: handling missing data, categorical encoding, model metrics (accuracy, ROC).

2. House Price Prediction — *Beginner*

Description: Predict house prices from structured features (area, rooms, location).

Tools / Dataset: scikit-learn, XGBoost, Kaggle House Prices.

Key steps: EDA → feature engineering → regression models → hyperparameter tuning.

Learning outcomes: regression metrics (RMSE, MAE), feature importance interpretation.

3. Iris Flower Classification — *Beginner*

Description: Classify iris species using petal and sepal measurements.

Tools / Dataset: scikit-learn (built-in iris).

Key steps: visualize data → train-test split → simple classifiers (k-NN, SVM) → confusion matrix.

Learning outcomes: multiclass classification basics, decision boundaries.

4. Spam Email Classifier — *Beginner / Intermediate*

Description: Classify emails as spam or not using text features.

Tools / Dataset: NLTK, scikit-learn, Enron or SMS Spam Collection dataset.

Key steps: text cleaning → TF-IDF vectorization → Naive Bayes / logistic regression → evaluate precision/recall.

Learning outcomes: text preprocessing, balancing precision and recall.

5. Credit Card Fraud Detection — *Intermediate*

Description: Detect fraudulent transactions in highly imbalanced data.

Tools / Dataset: scikit-learn, imbalanced-learn, Kaggle Credit Card Fraud dataset.

Key steps: data sampling strategies (SMOTE), anomaly detection models, ensemble methods.

Learning outcomes: dealing with class imbalance, ROC-AUC interpretation.

6. Customer Segmentation (Clustering) — *Beginner / Intermediate*

Description: Segment customers using transaction data for marketing.

Tools / Dataset: scikit-learn, KMeans, PCA, Mall Customers dataset.

Key steps: scale data → dimensionality reduction → clustering → interpret segments.

Learning outcomes: unsupervised learning, silhouette score, cluster profiling.

7. **Stock Price Prediction (Time Series)** — *Intermediate*

Description: Forecast future stock prices using historical data.

Tools / Dataset: pandas, statsmodels, LSTM with TensorFlow, Yahoo Finance.

Key steps: time-series decomposition → feature creation (lags) → ARIMA or LSTM → backtesting.

Learning outcomes: time-series concepts, train-test temporal split, overfitting in time-series.

8. **Handwritten Digit Recognition (MNIST)** — *Beginner / Intermediate*

Description: Classify handwritten digits using CNNs.

Tools / Dataset: TensorFlow/Keras or PyTorch, MNIST dataset.

Key steps: reshape images → build CNN → train/validate → visualize filters and misclassifications.

Learning outcomes: basics of convolutional neural networks, image preprocessing.

9. **Sentiment Analysis of Movie Reviews** — *Intermediate*

Description: Predict sentiment (positive/negative) from movie reviews.

Tools / Dataset: NLTK/spaCy, scikit-learn, IMDb dataset or Kaggle.

Key steps: text cleaning → embeddings (word2vec/TF-IDF) → RNN/CNN or transformer fine-tuning.

Learning outcomes: sequence modeling, embedding techniques, transfer learning basics.

10. **Recommendation System (Collaborative Filtering)** — *Intermediate*

Description: Build a movie recommender using user-item ratings.

Tools / Dataset: Surprise library, Pandas, MovieLens dataset.

Key steps: matrix factorization (SVD), evaluation (RMSE), cold-start strategies.

Learning outcomes: recommender system paradigms, evaluation metrics.

11. **Image Caption Generator** — *Advanced*

Description: Generate descriptive captions for images using CNN + RNN.

Tools / Dataset: TensorFlow, Flickr8k/Flickr30k.

Key steps: extract image features (CNN) → sequence model (LSTM) → train with paired captions → beam search for inference.

Learning outcomes: multimodal modeling, sequence-to-sequence learning.

12. **Fake News Detector** — *Intermediate*

Description: Classify news articles as fake or real using NLP techniques.

Tools / Dataset: Transformers, Kaggle FakeNews dataset.

Key steps: preprocess text → fine-tune transformer model → evaluate precision/recall/F1.

Learning outcomes: modern NLP, transfer learning, model interpretability.

13. **Traffic Sign Recognition** — *Intermediate*

Description: Identify traffic signs from images for autonomous driving tasks.

Tools / Dataset: OpenCV, TensorFlow, German Traffic Sign dataset.

Key steps: data augmentation → CNN training → real-time detection using OpenCV.

Learning outcomes: robustness to variations, deployment in real-time systems.

14. **Voice Gender Recognition** — *Beginner / Intermediate*

Description: Classify voice samples by gender using audio features.

Tools / Dataset: Librosa, scikit-learn, voice datasets.

Key steps: extract MFCCs → build classifier (SVM/NN) → evaluate.

Learning outcomes: audio feature extraction, signal processing basics.

15. **Hand Gesture Recognition** — *Intermediate*

Description: Recognize hand gestures from webcam feed for HCI applications.

Tools / Dataset: OpenCV, MediaPipe, TensorFlow.

Key steps: detect hand landmarks → extract features → classify gestures.

Learning outcomes: real-time inference, keypoint detection.

16. **Face Emotion Recognition** — *Intermediate*

Description: Detect facial emotions from images or video frames.

Tools / Dataset: FER2013 dataset, CNNs, OpenCV.

Key steps: face detection → preprocess → train CNN → evaluate per-class accuracy.

Learning outcomes: facial feature extraction, class imbalance handling.

17. **Human Activity Recognition (HAR)** — *Intermediate*

Description: Classify activities (walking, running) using smartphone sensor data.

Tools / Dataset: UCI HAR dataset, scikit-learn, LSTM.

Key steps: sliding windows → feature extraction → train time-series models.

Learning outcomes: sensor data handling, sequence classification.

18. **Object Detection with YOLO** — *Advanced*

Description: Detect objects and draw bounding boxes in images.

Tools / Dataset: Darknet/YOLOv5, COCO dataset.

Key steps: choose model variant → fine-tune on custom dataset → evaluate mAP.

Learning outcomes: real-time detection, evaluation metrics for detection.

19. **Person Re-identification** — *Advanced*

Description: Match people across camera views for surveillance analytics.

Tools / Dataset: Deep learning frameworks, Market-1501 dataset.

Key steps: feature embedding learning → triplet loss training → retrieval metrics.

Learning outcomes: metric learning, retrieval systems.

20. **Churn Prediction for Telecom** — *Intermediate*

Description: Predict which customers are likely to leave a service.

Tools / Dataset: scikit-learn, XGBoost, Telco customer churn dataset.

Key steps: feature creation → handle categorical data → class imbalance → model interpretation.

Learning outcomes: business-oriented modeling, feature importance.

21. **Loan Default Prediction** — *Intermediate*

Description: Predict loan repayment default to help lenders.

Tools / Dataset: LendingClub dataset, scikit-learn.

Key steps: preprocess financial data → build risk scoring model → validate performance.

Learning outcomes: credit scoring, regulatory-aware modeling.

22. **Medical Image Segmentation (U-Net)** — *Advanced*

Description: Segment organs or tumors in medical images.

Tools / Dataset: TensorFlow/PyTorch, U-Net architecture, medical datasets (ISIC for skin).

Key steps: prepare masks → train U-Net → evaluate Dice coefficient.

Learning outcomes: segmentation tasks, working with medical annotations.

23. **Predicting Diabetes (Healthcare)** — *Beginner / Intermediate*

Description: Use clinical data to predict diabetes risk.

Tools / Dataset: Pima Indians Diabetes dataset, scikit-learn.

Key steps: EDA → feature selection → logistic regression / tree-based models.

Learning outcomes: understanding medical data, model explainability.

24. **Personality Prediction from Text** — *Intermediate*

Description: Infer personality traits from social media posts.

Tools / Dataset: NLP tools, Kaggle personality datasets.

Key steps: text preprocessing → feature extraction → multi-label classification.

Learning outcomes: multi-label tasks, ethical considerations in profiling.

25. **Traffic Flow Prediction (City Planning)** — *Intermediate*

Description: Forecast traffic intensity using sensor/time-series data.

Tools / Dataset: LSTM/GRU, city traffic datasets.

Key steps: spatio-temporal modeling → train recurrent networks → evaluate forecasting errors.

Learning outcomes: spatio-temporal models, urban data applications.

26. **Plant Disease Detection** — *Intermediate*

Description: Detect disease on plant leaves using images.

Tools / Dataset: MobileNet or EfficientNet, PlantVillage dataset.

Key steps: data augmentation → train CNN → mobile deployment.

Learning outcomes: practical agriculture applications, model optimization for edge devices.

27. **Optical Character Recognition (OCR)** — *Intermediate*

Description: Convert images of text into machine-encoded text.

Tools / Dataset: Tesseract, deep learning OCR pipelines.

Key steps: text detection → recognition → post-processing.

Learning outcomes: image-to-text pipelines, sequence decoding.

28. **Language Translation (Seq2Seq)** — *Advanced*

Description: Build a simple neural machine translation system.

Tools / Dataset: Seq2Seq with attention, TensorFlow, parallel corpora (e.g., English-French).

Key steps: build encoder-decoder → train on parallel sentences → evaluate

BLEU score.

Learning outcomes: sequence modeling, attention mechanisms.

29. **Traffic Sign Detection for Cyclists** — *Intermediate*

Description: Detect and notify about traffic signs using a helmet-mounted camera.

Tools / Dataset: OpenCV, lightweight CNNs, custom dataset.

Key steps: real-time detection → low-power model selection → prototype app.

Learning outcomes: model optimization, embedded inference.

30. **Resume Screening Automation** — *Intermediate*

Description: Rank resumes for job matching using NLP features.

Tools / Dataset: spaCy, transformers, custom resume dataset.

Key steps: parse resumes → extract skills → build ranking classifier.

Learning outcomes: entity extraction, building real-world HR tools.

31. **Music Genre Classification** — *Intermediate*

Description: Predict music genre from audio snippets.

Tools / Dataset: Librosa, GTZAN dataset, CNNs on spectrograms.

Key steps: convert audio to spectrograms → train CNN → evaluate accuracy.

Learning outcomes: audio processing, feature engineering from sound.

32. **Predicting Student Performance** — *Beginner / Intermediate*

Description: Predict student grades using demographic and study data.

Tools / Dataset: UCI student performance dataset.

Key steps: feature selection → regression/classification → actionable insights for educators.

Learning outcomes: educational data mining, fairness considerations.

33. **Product Review Summarizer** — *Advanced*

Description: Summarize customer reviews into concise highlights.

Tools / Dataset: Transformers (BART, T5), Amazon reviews dataset.

Key steps: fine-tune summarization model → evaluate ROUGE scores → build UI.

Learning outcomes: abstractive summarization, model deployment.

34. **Topic Modeling on News Articles** — *Beginner / Intermediate*

Description: Discover topics in a news corpus using LDA or NMF.

Tools / Dataset: Gensim, news datasets.

Key steps: text cleaning → vectorization → fit LDA → visualize topics.

Learning outcomes: unsupervised text analysis, topic coherence.

35. **Real-time Object Tracking** — *Advanced*

Description: Track moving objects across video frames (e.g., people or cars).

Tools / Dataset: OpenCV (SORT, Deep SORT), video datasets.

Key steps: detection → data association → track management.

Learning outcomes: multi-object tracking, performance in real-time.

36. **Automated Essay Scoring** — *Advanced*

Description: Score essays automatically by content and quality.

Tools / Dataset: NLP models, ASAP-AES dataset.

Key steps: feature extraction (cohesion, grammar) → regression or ordinal classification.

Learning outcomes: regression on subjective labels, fairness and bias considerations.

37. **Predicting Heart Disease** — *Intermediate*

Description: Build a classifier for heart disease risk from medical records.

Tools / Dataset: UCI Heart Disease dataset.

Key steps: data preprocessing → model training → interpret results with SHAP/LIME.

Learning outcomes: explainable AI in healthcare, model validation.

38. **Bike Sharing Demand Forecasting** — *Intermediate*

Description: Forecast bike rentals per hour/day for sharing systems.

Tools / Dataset: Kaggle Bike Sharing dataset.

Key steps: time-series features → regression models → error analysis.

Learning outcomes: operational forecasting, seasonal patterns.

39. **Wine Quality Prediction** — *Beginner*

Description: Predict wine quality scores from physicochemical tests.

Tools / Dataset: UCI Wine Quality dataset.

Key steps: regression/classification → feature importance → cross-validation.

Learning outcomes: simple regression pipelines, model validation.

40. **AI Chatbot for FAQs** — *Intermediate*

Description: Build a rule-based + ML chatbot to answer FAQs.

Tools / Dataset: Rasa, transformers, domain FAQ dataset.

Key steps: intent recognition → response retrieval/generation → integrate

with web.

Learning outcomes: conversational AI basics, user intent classification.

41. **Style Transfer for Images** — *Advanced*

Description: Apply artistic style from one image to another using neural style transfer.

Tools / Dataset: PyTorch/TensorFlow implementations.

Key steps: compute content and style losses → iterative optimization → evaluate visually.

Learning outcomes: optimization-based generation, perceptual losses.

42. **Face Recognition System** — *Advanced*

Description: Identify known individuals in images using embeddings.

Tools / Dataset: FaceNet, pre-trained models, custom face datasets.

Key steps: detect faces → generate embeddings → build a matching database.

Learning outcomes: privacy and ethics, model calibration.

43. **Document Classification (Legal/Medical)** — *Intermediate*

Description: Classify documents into categories like contract types or medical reports.

Tools / Dataset: Transformers, domain corpora.

Key steps: fine-tune models → evaluate classification metrics → deploy classifier.

Learning outcomes: domain adaptation, labeling strategies.

44. **Autonomous Drone Navigation (Simulated)** — *Advanced*

Description: Train a drone agent to navigate in simulation using RL/**ML**.

Tools / Dataset: AirSim, OpenAI Gym, reinforcement learning libraries.

Key steps: define reward → train policy → test in sim.

Learning outcomes: reinforcement learning basics, sim-to-real challenges.

45. **Optical Music Recognition** — *Advanced*

Description: Convert sheet music images into machine-readable music notation.

Tools / Dataset: Computer vision methods, specialized music datasets.

Key steps: symbol detection → sequence reconstruction → export MIDI.

Learning outcomes: niche OCR tasks, sequence reconstruction.

46. **Energy Consumption Forecasting** — *Intermediate*

Description: Predict electricity usage for buildings or grids.

Tools / Dataset: time-series tools, energy datasets.

Key steps: feature engineering for seasonality → train model → scenario analysis.

Learning outcomes: forecasting for resource planning, anomaly detection.

47. **Pose Estimation for Fitness Apps** — *Intermediate / Advanced*

Description: Detect body keypoints to evaluate exercise form.

Tools / Dataset: MediaPipe, OpenPose, TensorFlow.

Key steps: keypoint detection → rule-based correctness checks → feedback UI.

Learning outcomes: pose estimation, real-time feedback systems.

48. **Named Entity Recognition (NER) System** — *Intermediate*

Description: Extract entities (names, locations) from text.

Tools / Dataset: spaCy, transformers, CoNLL-2003 dataset.

Key steps: tokenization → model fine-tuning → evaluation (F1).

Learning outcomes: sequence labeling, dataset preparation.

49. **Household Appliance Fault Detection** — *Intermediate*

Description: Detect anomalies in appliance sensor data to predict faults.

Tools / Dataset: anomaly detection methods, sensor datasets.

Key steps: unsupervised anomaly detection → thresholding → alert system.

Learning outcomes: anomaly detection, IoT data handling.

50. **Predicting Movie Box Office Success** — *Intermediate*

Description: Predict opening weekend revenue using cast, genre, and marketing features.

Tools / Dataset: web-scraped movie features, regression models.

Key steps: feature collection → model training → interpret influential factors.

Learning outcomes: feature engineering from varied sources, business analytics.

51. **Automatic Speech Recognition (ASR)** — *Advanced*

Description: Build a system that converts speech to text.

Tools / Dataset: Kaldi, DeepSpeech, Librispeech.

Key steps: acoustic modeling → language modeling → evaluate WER.

Learning outcomes: speech modeling, sequence-to-sequence systems.

52. **Emotion Detection from Voice** — *Intermediate*

Description: Detect emotions (happy, sad) from audio features.

Tools / Dataset: Librosa, RAVDESS dataset.

Key steps: extract prosodic features → train classifier → evaluate confusion matrix.

Learning outcomes: paralinguistic feature extraction, model generalization.

53. **Forecasting Air Quality Index (AQI)** — *Intermediate*

Description: Predict future AQI values for cities using environmental data.

Tools / Dataset: time-series models, environmental datasets.

Key steps: collect weather + pollutant data → train forecasting model → validate.

Learning outcomes: environmental data modeling, public health implications.

54. **Scene Text Detection in Natural Images** — *Advanced*

Description: Detect and read text in complex scenes (storefronts, signs).

Tools / Dataset: EAST/CRAFT text detectors, ICDAR datasets.

Key steps: detection → recognition → post-processing.

Learning outcomes: computer vision pipelines for text in the wild.

55. **Automated Code Commenting (Code Summarization)** — *Advanced*

Description: Generate comments or summaries for source code.

Tools / Dataset: transformers, CodeSearchNet dataset.

Key steps: tokenization of code → fine-tune seq2seq models → evaluate BLEU/ROUGE.

Learning outcomes: code understanding, specialized tokenization.

56. **Multi-label Image Classification** — *Intermediate*

Description: Predict multiple labels per image (e.g., objects present).

Tools / Dataset: TensorFlow, Pascal VOC, MS-COCO.

Key steps: adapt loss to multi-label (sigmoid + BCE) → threshold selection.

Learning outcomes: multi-label paradigms, evaluation with mAP.

57. **Fake Product Review Generator (and Detection)** — *Advanced (ethical caution)*

Description: Generate synthetic reviews and build detectors to spot them.

Tools / Dataset: text generation models, real review datasets.

Key steps: train generator → create detector → discuss ethics.

Learning outcomes: adversarial generation, importance of ethics and misuse prevention.

58. **Personal Expense Categorizer** — *Beginner*

Description: Automatically categorize expense descriptions into categories.

Tools / Dataset: scikit-learn, simple labeled transactions.

Key steps: text preprocessing → classification → integrate into a personal finance dashboard.

Learning outcomes: text classification for practical apps.

59. **Handwriting Style Transfer** — *Advanced*

Description: Convert typed text into a particular person's handwriting style.

Tools / Dataset: image generation / GANs, handwriting datasets.

Key steps: model style and content separation → generate synthetic handwriting.

Learning outcomes: generative modeling, one-shot style transfer.

60. **Predictive Maintenance for Manufacturing** — *Intermediate*

Description: Predict machine failures using sensor logs.

Tools / Dataset: time-series anomaly detection, PHM datasets.

Key steps: extract features from sensor streams → build classifier/regressor for remaining useful life.

Learning outcomes: practical industry applications, survival analysis.

61. **Real Estate Recommendation Engine** — *Intermediate*

Description: Recommend properties to users based on preferences.

Tools / Dataset: collaborative + content-based recommender techniques.

Key steps: combine user profiles and property features → ranking algorithm → UI.

Learning outcomes: hybrid recommenders, personalized ranking.

62. **Smart Attendance System using Face Recognition** — *Intermediate*

Description: Mark attendance of students automatically via face ID.

Tools / Dataset: OpenCV, face recognition models.

Key steps: capture images → enroll faces → real-time recognition and logging.

Learning outcomes: building practical educational tools, privacy considerations.

63. **Document Similarity Search Engine** — *Intermediate*

Description: Retrieve similar documents to a query using embeddings.

Tools / Dataset: sentence-transformers, FAISS, document corpora.

Key steps: embed documents → build vector index → nearest neighbor retrieval.

Learning outcomes: semantic search, vector databases.

64. **Style-based Text Generation (Creative Writing Aid)** — *Advanced*

Description: Generate text in the style of a chosen author for creative

prompts.

Tools / Dataset: GPT-style models, fine-tuning datasets.

Key steps: collect style data → fine-tune → evaluate human-likeness.

Learning outcomes: language modeling, creative use-cases and ethics.

65. **Multi-modal Sentiment Analysis (Text + Images)** — *Advanced*

Description: Predict sentiment using post text and attached images.

Tools / Dataset: transformers for text, CNN for images, social media datasets.

Key steps: extract features from both modalities → fuse and classify.

Learning outcomes: multimodal fusion strategies.

66. **Anomaly Detection in Network Traffic** — *Advanced*

Description: Detect intrusions or unusual patterns in network logs.

Tools / Dataset: UNSW-NB15/KDDCup datasets, autoencoders.

Key steps: model normal behavior → detect deviations → alerting thresholds.

Learning outcomes: cybersecurity applications, unsupervised detection.

67. **Visual Question Answering (VQA)** — *Advanced*

Description: Answer questions about images using combined vision and language models.

Tools / Dataset: VQA datasets, multimodal transformers.

Key steps: encode image and question → multimodal fusion → answer generation.

Learning outcomes: complex multimodal reasoning.

68. **Predicting Crop Yield** — *Intermediate*

Description: Forecast crop yields using weather and satellite data.

Tools / Dataset: remote sensing data, regression models.

Key steps: pre-process geospatial data → combine with weather → model training.

Learning outcomes: working with satellite data, agronomy-focused modeling.

69. **Automated Bug Triage from Issue Descriptions** — *Intermediate*

Description: Assign software bugs to the correct team or priority.

Tools / Dataset: NLP, issue trackers (GitHub) data.

Key steps: label mapping → classification → evaluation.

Learning outcomes: practical software engineering tooling.

70. **Emotion-aware Music Player** — *Advanced*

Description: Suggest music based on user's detected mood (face or voice).

Tools / Dataset: emotion detection + recommender systems.

Key steps: detect mood → map to playlist → real-time updates.

Learning outcomes: combining perception with personalization.

71. **Text-to-Image Generation (Diffusion/GAN)** — *Advanced*

Description: Generate images from textual prompts.

Tools / Dataset: Stable Diffusion/StyleGAN, captioned image datasets.

Key steps: train or fine-tune generator → evaluate image quality.

Learning outcomes: generative modeling, prompt engineering.

72. **Automated Sports Highlights Generator** — *Advanced*

Description: Detect exciting moments in sports video to create highlight reels.

Tools / Dataset: video analysis, multi-modal features.

Key steps: event detection → summarization → minimal editing pipeline.

Learning outcomes: video analytics, event detection.

73. **Multilingual Chatbot** — *Intermediate*

Description: Build a chatbot that supports multiple languages.

Tools / Dataset: Transformers multilingual models, translation APIs.

Key steps: intent detection across languages → response generation or retrieval.

Learning outcomes: cross-lingual modeling.

74. **Predicting Hospital Readmission** — *Advanced*

Description: Predict likelihood of patient readmission using EMR data.

Tools / Dataset: MIMIC dataset (requires access), clinical models.

Key steps: privacy-aware preprocessing → model with explainability → validate clinically.

Learning outcomes: healthcare ML with ethics and privacy.

75. **Graph-based Recommender (Social Networks)** — *Advanced*

Description: Use graph embeddings to recommend friends or content.

Tools / Dataset: NetworkX, PyTorch Geometric.

Key steps: build graph → apply node embeddings → recommend via similarity.

Learning outcomes: graph ML, link prediction.

76. **Bias Detection in ML Models** — *Advanced*

Description: Analyze trained models for demographic bias and propose

fixes.

Tools / Dataset: fairness libraries (AIF360), domain datasets.

Key steps: measure fairness metrics → mitigate bias → re-evaluate.

Learning outcomes: responsible AI, mitigation strategies.

77. **Image Super-Resolution** — *Advanced*

Description: Increase image resolution using deep learning (SRCNN, EDSR).

Tools / Dataset: DIV2K dataset, PyTorch/TensorFlow.

Key steps: upsampling models → train on patches → evaluate PSNR/SSIM.

Learning outcomes: image restoration techniques.

78. **Ad Click-Through Rate (CTR) Prediction** — *Advanced*

Description: Predict probability a user clicks an ad using large-scale features.

Tools / Dataset: xDeepFM, large click datasets.

Key steps: feature engineering for categorical high-cardinality → build deep models → evaluate AUC.

Learning outcomes: large-scale sparse feature modeling.

79. **Automated Market News Summarizer for Traders** — *Advanced*

Description: Summarize financial news and predict short-term market reaction.

Tools / Dataset: transformers, financial news datasets.

Key steps: extract summaries → sentiment/regression for impact → backtest strategies.

Learning outcomes: domain-specific NLP, model evaluation in finance.

80. **Personalized Learning Path Recommendation** — *Intermediate*

Description: Recommend courses/resources based on student progress and gaps.

Tools / Dataset: educational datasets, collaborative filtering.

Key steps: model user proficiency → recommend next steps → feedback loop.

Learning outcomes: adaptive learning systems.

81. **Image Colorization** — *Intermediate / Advanced*

Description: Automatically colorize grayscale images using CNNs.

Tools / Dataset: CIFAR/ImageNet or custom photos.

Key steps: design encoder-decoder → train using L2/perceptual losses → evaluate visually.

Learning outcomes: generative image tasks and evaluation.

82. **Emotion-aware Text Reply Suggestion** — *Intermediate*

Description: Suggest empathetic responses in chat applications.

Tools / Dataset: transformers, dialogue datasets.

Key steps: detect emotion → generate response templates or neural replies.

Learning outcomes: conversational AI with emotional intelligence.

83. **Predicting Road Accidents Hotspots** — *Intermediate*

Description: Identify locations with high accident risk using historical data.

Tools / Dataset: GIS data, traffic datasets.

Key steps: spatial feature engineering → classification/regression → mapping results.

Learning outcomes: geospatial ML, public safety analytics.

84. **Automated Grading for Coding Assignments** — *Advanced*

Description: Evaluate code submissions for correctness and style.

Tools / Dataset: static analysis, test harnesses, code similarity detectors.

Key steps: run unit tests → measure performance → detect plagiarism.

Learning outcomes: building evaluation pipelines, code analysis.

85. **Pose-guided Animation Transfer** — *Advanced*

Description: Transfer poses from a source video to animate a character.

Tools / Dataset: deep generative models, keypoint datasets.

Key steps: detect keypoints → condition generator → synthesize frames.

Learning outcomes: motion transfer, conditional generation.

86. **Predicting Floods Using Satellite Imagery** — *Advanced*

Description: Use remote sensing to detect flood risks and extent.

Tools / Dataset: satellite imagery, segmentation models.

Key steps: preprocess multispectral data → segmentation/regression → validate against ground truth.

Learning outcomes: geospatial analysis, disaster response ML.

87. **Speech Emotion Conversion** — *Advanced*

Description: Convert speech from one emotion to another while preserving content.

Tools / Dataset: voice datasets, sequence-to-sequence with style transfer.

Key steps: disentangle content and style → map style vectors → synthesize speech.

Learning outcomes: advanced speech synthesis and style transfer.

88. **Customer Lifetime Value (CLV) Prediction** — *Intermediate*

Description: Predict future value a customer will bring to business.

Tools / Dataset: transaction logs, survival models or regression.

Key steps: compute historical features → model CLV → segment customers.

Learning outcomes: business metrics, long-term forecasting.

89. **Automatic Diagram Understanding** — *Advanced*

Description: Interpret flowcharts or diagrams into structured representations.

Tools / Dataset: computer vision + graph extraction techniques.

Key steps: detect nodes and edges → build graph → map semantics.

Learning outcomes: visual-structure parsing, domain-specific extraction.

90. **Personalized News Feed Ranking** — *Advanced*

Description: Rank news articles personalized to user preferences in real-time.

Tools / Dataset: ranking models, user logs.

Key steps: build candidate retrieval → ranking model → evaluation (NDCG).

Learning outcomes: production ranking systems, online evaluation metrics.

91. **Text De-identification for Privacy** — *Advanced*

Description: Detect and mask personal identifiers in text (names, emails).

Tools / Dataset: NER models, privacy toolkits.

Key steps: NER detection → masking → evaluate recall on sensitive entities.

Learning outcomes: privacy-preserving NLP, compliance.

92. **Retail Inventory Demand Forecasting** — *Intermediate*

Description: Predict SKU demand to optimize stocking and reduce stockouts.

Tools / Dataset: time-series models, historical sales data.

Key steps: hierarchical forecasting → incorporate promotions → evaluate forecast accuracy.

Learning outcomes: operational ML for retail.

93. **AI-based Language Tutor** — *Advanced*

Description: Provide corrective feedback on language learners' utterances.

Tools / Dataset: ASR + grammar checking models, learner corpora.

Key steps: transcribe speech → parse grammar mistakes → suggest

corrections with examples.

Learning outcomes: tutoring systems, error detection and feedback design.

94. **Autonomous Wheelchair Navigation (Simulation)** — *Advanced*

Description: Build navigation policy for an autonomous wheelchair in a simulated environment.

Tools / Dataset: ROS, Gazebo, RL algorithms.

Key steps: environment setup → reward design → train navigation policy.

Learning outcomes: robotics, safe RL practices.

95. **Wine Recommendation by Taste Profile** — *Intermediate*

Description: Recommend wines based on user's taste preferences and past ratings.

Tools / Dataset: collaborative filtering, content-based features.

Key steps: build user profile → match wines by attributes → evaluate user satisfaction.

Learning outcomes: personalized recommendations, user modeling.

96. **Automated Meeting Minute Generator** — *Advanced*

Description: Transcribe meetings and summarize key points and action items.

Tools / Dataset: ASR, summarization models.

Key steps: transcribe audio → segment topics → summarize and extract tasks.

Learning outcomes: multi-step pipeline integrating speech and NLP.

97. **Semantic Image Search Engine** — *Advanced*

Description: Search images using natural language queries.

Tools / Dataset: CLIP, image-caption datasets.

Key steps: embed images and text into shared space → nearest neighbor search → evaluate relevance.

Learning outcomes: cross-modal retrieval, modern embedding tools.

98. **Real-time Language Translation for Video Calls** — *Advanced*

Description: Provide on-the-fly subtitles in another language during calls.

Tools / Dataset: ASR → translation → TTS, streaming pipeline.

Key steps: low-latency ASR → machine translation → display subtitles.

Learning outcomes: systems engineering for low-latency pipelines.

99. **Smart Farming — Weed Detection and Removal** — *Advanced*

Description: Detect weeds in crop images to guide robotic weed removal.

Tools / Dataset: segmentation/detection datasets, robotics integration.

Key steps: build detection model → integrate with actuator control → field testing.

Learning outcomes: agri-tech, model-to-actuator integration.

100. **End-to-End ML Project: From Data to Deployment** — *Advanced*

Description: Choose a domain problem, collect data, build models, then deploy as a web app or API.

Tools / Dataset: any of the above datasets; Flask/FastAPI, Docker, CI/CD.

Key steps: problem scoping → data pipeline → modeling → packaging and deploying → monitoring.

Learning outcomes: full ML lifecycle, MLOps basics, maintainable pipelines.

Tips for choosing and completing a project

- **Start small:** If you're new, pick 1–3 beginner projects and complete them end-to-end.
- **Focus on process:** Document EDA, preprocessing, model selection, evaluation, and conclusions.
- **Use version control:** Keep code in GitHub with clear READMEs and sample notebooks.
- **Create a portfolio:** Turn the best projects into blog posts or demo notebooks with visualizations and explanations targeted at non-technical reviewers (e.g., hiring managers).
- **Practice model explainability:** Use SHAP, LIME, or feature importance plots to explain why models make decisions.
- **Consider ethics:** For projects involving people (faces, health, or sensitive attributes), think about privacy, bias, and consent. Add a short ethics section in your project report.
- **Deploy small apps:** Use Streamlit, Flask, or simple web UIs to show interactive demos.
- **Iterate and improve:** Add more data, try different models, optimize hyperparameters, and measure improvements.

Must Read: [150 Facts Project Ideas — Simple, clear & ready-to-use ideas](#)

Final Notes

This list of **100 Machine Learning Projects Ideas** is meant to guide students through practical, hands-on learning. Each project can be scaled up or down depending on your skill level and time availability.

Start with projects that excite you — interest fuels persistence. For every project you complete, write a short summary: problem statement, dataset, models tried, results, and what you learned.

That practice will build both technical skill and communication ability, which are essential for a career in machine learning.

If you want, I can help you pick the best 3 projects to start with based on your current skills (Python familiarity, math background, and whether you've used TensorFlow/**PyTorch**).

I can also provide a starter notebook template for any chosen project so you can begin coding right away.

 **Blog**



JOHN DEAR

I am a creative professional with over 5 years of experience in coming up with project ideas. I'm great at brainstorming, doing market research, and analyzing what's possible to develop innovative and impactful projects. I also excel in collaborating with teams, managing project timelines, and ensuring that every idea turns into a successful outcome. Let's work together to make your next project a success!



**150 Ready-to-use Topics – Policy
Project Ideas**

Best Project Ideas

Are you ready to make your big ideas happen? Let's connect and discuss how we can bring your vision to life. Together, we can create amazing results and turn your dreams into reality.

Top Pages

[Terms And Conditions](#)

[Disclaimer](#)

[Privacy Policy](#)

Follow Us

© 2024 [Best Project Ideas](#)