Deep Learning

End-Term Project Instructions

Project Weight: 100 Points

Project Overview

This end-term project requires students to synthesize multiple deep learning concepts into a cohesive, innovative application. Students will explore advanced architectures and techniques while addressing real-world problems through rigorous experimentation and analysis.

Core Requirements

Students must integrate at least 3 of the 5 topic areas covered in this course, demonstrate deep understanding through implementation and experimentation, present clear analysis of results with comprehensive ablation studies, and submit reproducible code with proper documentation.

1 Project Tracks

Students must select one track from the following options.

1.1 Track 1: Multimodal Creative AI System

Topic Integration: Vision + Transformers + Generative Models

Build an AI system that understands and generates creative content across modalities.

Project Options

Story-to-Video Generator: Transform text narratives into video sequences using text-to-image generation combined with temporal modeling.

Style-Conditioned Image Captioning: Generate artistic descriptions of images in different literary styles (poetry, technical, dramatic).

Music-Driven Animation: Create visual animations synchronized with music using audio transformers and GANs/VAEs.

Required Components

The project must include a vision encoder (CNN/Vision Transformer) for image understanding, attention mechanisms for cross-modal alignment, a generative model (VAE/GAN) for content creation, and regularization techniques to prevent mode collapse and overfitting.

Dataset Flexibility

Suggested datasets include COCO, Flickr30k, ImageNet, WikiArt, YouTube-8M, AudioSet, or custom collected data. Minimum recommended samples: 5,000.

1.2 Track 2: Predictive Intelligence for Time-Series

Topic Integration: Sequences + Regularization + Transformers

Develop a sophisticated forecasting system with uncertainty quantification capabilities.

Project Options

Multi-Horizon Financial Predictor: Predict stock prices, cryptocurrency values, or economic indicators with confidence intervals.

Healthcare Trajectory Modeling: Forecast patient vital signs or disease progression from medical time-series data.

Climate & Weather Forecasting: Predict temperature, precipitation, or extreme weather events using historical climate data.

Smart Grid Energy Prediction: Forecast electricity demand incorporating renewable energy sources.

Required Components

The project must implement a sequence model (RNN/LSTM/GRU baseline with Transformer comparison), attention mechanisms for identifying critical temporal dependencies, regularization techniques (dropout, batch normalization, weight decay), and uncertainty estimation through ensemble methods or Bayesian approaches.

Dataset Flexibility

For financial applications, students may use Yahoo Finance, Cryptocurrency APIs, or FRED Economic Data. Healthcare options include MIMIC-III, PhysioNet, or COVID-19 datasets. Climate data can be obtained from NOAA or ERA5 reanalysis data. Energy forecasting may use smart grid datasets or utility company data. Minimum recommended timesteps: 1,000+.

1.3 Track 3: Autonomous Vision-Based Agent

Topic Integration: Vision + Robotics + Sequences + Regularization

Create an intelligent agent that perceives and acts in visual environments through learned representations.

Project Options

Autonomous Navigation System: Train an agent to navigate complex environments using visual input.

Robotic Manipulation Planner: Learn to grasp and manipulate objects from RGB/depth camera images.

Gesture-Controlled Interface: Build a system that understands human gestures for intuitive device control.

Visual SLAM & Exploration: Implement simultaneous localization and mapping with learned visual representations.

Required Components

The project must include a vision processing pipeline (CNN/ResNet/EfficientNet), sequence model for temporal reasoning and action planning, reinforcement learning or imitation learning framework, and regularization strategies to ensure robust generalization to novel environments.

Dataset Flexibility

Simulation environments include Unity ML-Agents, PyBullet, or OpenAI Gym. Real-world datasets include RoboNet, KITTI, TUM RGB-D, or custom collected data. Gesture datasets include 20BN-Jester, WLASL, or ChaLearn. Minimum recommended episodes/trajectories: 2,000.

1.4 Track 4: Advanced Generative Content System

Topic Integration: GANs + VAEs + Transformers + Regularization

Push the boundaries of generative modeling with controlled, high-quality synthesis.

Project Options

Controllable Face/Object Generation: Generate images with specific attributes (age, style, pose, expression).

Neural Style Transfer Pipeline: Implement multi-style transfer with semantic preservation using attention mechanisms.

Data Augmentation Engine: Generate synthetic training data to improve downstream task performance.

Text-to-3D Generation: Create three-dimensional models from textual descriptions.

Required Components

The project must implement at least two generative models (comparative study of VAE vs GAN approaches), an attention mechanism for fine-grained control, comprehensive regularization strategy (spectral normalization, gradient penalty, KL annealing), and quantitative evaluation metrics (FID, Inception Score, reconstruction loss).

Dataset Flexibility

For vision applications, students may use CelebA, FFHQ, LSUN, ImageNet subset, or custom image collections. For 3D generation, options include ShapeNet, ModelNet, or custom 3D scans. Minimum recommended samples: 10,000.

1.5 Track 5: Intelligent Document Understanding

Topic Integration: Transformers + Vision + Sequences

Build a system that deeply understands structured and unstructured documents.

Project Options

Visual Question Answering on Documents: Answer questions about charts, tables, and complex document layouts.

Scientific Paper Analyzer: Extract key findings, generate summaries, and identify relationships between concepts.

Historical Document Transcription: Implement OCR with context-aware correction for handwritten or degraded documents.

Legal/Medical Document Parser: Extract structured information from complex professional documents.

Required Components

The project must include a Transformer-based language model (BERT/GPT/T5 or custom architecture), vision encoder for document images (when applicable), sequence modeling for long-context understanding, and regularization techniques for effective domain adaptation.

Dataset Flexibility

Academic datasets include arXiv, PubMed, or Semantic Scholar. Document datasets include DocVQA, RVL-CDIP, FUNSD, or IIT-CDIP. Historical datasets include IAM Handwriting Database or NIST datasets. Legal/Medical options include MIMIC or i2b2 (with appropriate access permissions).

2 Evaluation Rubric

Evaluation Component	Points
Technical Implementation	40
Analysis & Insights	30
Innovation & Creativity	15
Report & Presentation	15
Total	**100**

2.1 Technical Implementation (40 points)

Integration of Topics (15 points)

Projects are evaluated on meaningful and sophisticated incorporation of at least three course topics, clear demonstration of how different techniques complement each other, and justified architectural choices based on problem requirements.

Architecture Design (10 points)

Evaluation considers well-motivated model design decisions, novel combinations or adaptations of existing approaches, and appropriate complexity for the problem scope.

Code Quality (10 points)

Code is assessed on cleanliness, modularity, organization, comprehensive documentation and comments, and reproducible results with clear setup instructions.

Experimentation (5 points)

Students must demonstrate systematic hyperparameter tuning, multiple design iterations with documented rationale, and evidence of iterative improvement process.

2.2 Analysis & Insights (30 points)

Ablation Studies (15 points)

Projects must include systematic analysis of individual component contributions, comparison of different architectural choices (e.g., RNN vs Transformer), evaluation of various regularization techniques, and analysis of learned representations, attention patterns, or generated samples.

Performance Evaluation (10 points)

Evaluation metrics must be selected and justified appropriately. Students should compare against meaningful baselines, conduct statistical significance testing where applicable, and perform error analysis and failure case investigation.

Visualization (5 points)

Reports should include informative plots of training dynamics and results, visualization of attention mechanisms or learned features, quality assessment of generated samples (if applicable), and clear presentation of quantitative and qualitative results.

2.3 Innovation & Creativity (15 points)

Novel Approach (8 points)

Points are awarded for creative problem formulation or methodology, original combinations of techniques, and innovative solutions to identified challenges.

Real-World Relevance (7 points)

Projects are evaluated on practical applicability of the proposed solution, consideration of deployment challenges and constraints, potential impact and societal benefits, and discussion of limitations and ethical considerations.

2.4 Report & Presentation (15 points)

Written Report (10 points)

The report should follow academic paper format and include a clear and compelling problem statement with motivation, comprehensive review of related work, detailed methodology description with appropriate technical depth, thorough results presentation and insightful discussion, honest assessment of limitations and well-reasoned future work proposals, and professional academic writing style. Length: 4–8 pages excluding references.

Presentation (5 points)

Presentations should be clear and engaging (10 minutes), demonstrate effective communication of key concepts and results, include high-quality slides with appropriate visualizations, provide a live demonstration or video of system functionality, and show professional handling of the Q&A session (5 minutes).

3 Important Notes

Project Guidelines

Team Size: Projects may be completed individually or in teams of up to 2 students.

Submission Deadline: Final project defense must be completed by the end of Week 10.

Alternative Projects: Students may propose alternative project ideas within the course scope, subject to instructor approval.

Pre-trained Models: Pre-trained models may be used as components but must be clearly documented and justified.

Experimental Evidence: All experimental claims must be supported by appropriate evidence and ablation studies.

Code Originality: Code must be original work with proper attribution for any external libraries or implementations.

Evaluation Criteria: Projects will be evaluated on both technical merit and presentation quality.