Table 1: Module Design – Complete Sequence Modeling

	Module	Learning Goals	Learning Objectives	Learning Activities	Instructional Materials	Delegate
Chapter 1 Deep Learning for Tabular Data	**Module 1: Linear Transformations to Neural Networks**	**1:** Matrix operations & geometric intuition **2:** Nonlinearity necessity **3:** Layer composition power **4:** Universal approximation intuition **5:** SGD fundamentals	• Matrix-vector multiplication geometry • XOR impossibility demonstration • Biological neuron threshold analogy • ReLU activation introduction • Multilayer perceptron construction • Grasp SGD as iterative parameter updates	• Linear transformation visualization • XOR failure analysis • "Breaking linearity" experiment • MLP implementation from scratch • Conceptual SGD walkthrough on a loss surface • Function approximation playground	**Framework:** PyTorch **D2L:** 2.3, 3.1.4, 4.1, 5.1, 12.4 **Materials:** NumPy tutorials, geometric visualizers	
	Module 2: Backprop & Automatic Differentiation	**5:** Gradient computation mastery **6:** Computational graphs **7:** Modern autograd frameworks	• Chain rule gradient calculations • Computational graph construction • Forward/backward pass mechanics • Autograd for automatic differentiation • Gradient checking verification • PyTorch autograd mastery	• Hand-calculated gradient exercise • Computational graph drawing • Gradient verification implementation • Build mini-autograd system • "Gradients without tears" coding	**D2L:** 2.4, 2.5, 5.3 **Materials:** PyTorch autograd guide, computational graph tools	
	Module 3: Training Deep MLP's & Controlling data and gradient flows	**8:** Core optimizers **9:** Ablation studies **10:** Paper analysis skills **11:** Experiment tracking & HPO	• SGD, momentum, Adam basics • Learning rate scheduling • Ablation study methodology • Reading ML papers effectively • Introduce experiment tracking tools • Systematic experimentation	• Optimizer comparison experiment • Design ablation study for MLPs • Log experiments with Weights & Biases • Systematic hyperparameter search • "Deconstructing a paper" workshop • Research notebook practices	**D2L:** 3.6–3.7, Ch. 12, Ch. 19 **Materials:** Classic papers, ablation templates, W&B tutorials	
Chapter 2 Deep Learning for Image Data	**Module 4: CNN Revolution & Spatial Intelligence**	**11:** MLP limitations & spatial invariance **12:** Parameter sharing power **13:** CNN optimization	• Analyze MLP failure (random features, poor generalization) • Motivate convolution for spatial invariance • Data augmentation for vision • Dropout in conv layers	• MLP vs CNN shifted-image demo • CNN optimization ablation • Data augmentation experiments • Regularization placement study • "What CNNs see" exploration • Paper analysis: AlexNet	**D2L:** 5.6, Ch. 7, 8.1, 14.1 **Materials:** Conv visualizers, AlexNet paper, augmentation tools	

Continued on next page

Table 1 – continued from previous page

Chapter	Module	Learning Goals	Learning Objectives	Learning Activities	Instructional Materials	Delegate
	Module 5: Modern CNNs: Depth, Normalization & Transfer	**14:** Deep network challenges **15:** Normalization techniques **16:** Transfer learning	• Gradient vanishing in deep CNNs • Batch normalization mechanics • ResNet skip connections • Normalization comparisons • Transfer learning methodology • Architecture efficiency analysis	• Gradient flow study • BatchNorm ablations • ResNet reproduction • Normalization comparison • Transfer learning competition • FLOPs vs accuracy tradeoffs	**D2L:** 5.4.1, 8.2–8.6, 14.2 **Materials:** ResNet paper, normalization studies, model zoo	
Chapter 3 Deep Learning for Sequence Data	**Module 6: Encoder-Decoder Architectures**	**17:** Encoder-decoder framework **18:** CNN encoders **19:** U-Net architecture	• Compression as representation learning • Fixed-size bottlenecks • CNN-based encoders for images • Skip connections in U-Net • Encoder-decoder for sequences • Applications: segmentation, translation	• Autoencoder from scratch • CNN encoder-decoder • U-Net implementation • Skip connection ablation • Bottleneck size experiments • Seq2seq preview	**D2L:** 10.6, 14.11 **Materials:** U-Net paper, encoder-decoder papers	
	Module 7: RNNs: Vanilla, LSTM & GRU	**20:** Sequential processing **21:** LSTM/GRU gates **22:** Training challenges	• RNN hidden state evolution • Vanishing/exploding gradients • LSTM gate mechanisms • GRU simplification • Gradient clipping strategies • Truncated BPTT	• Char-RNN implementation • Gradient visualization • LSTM from scratch • Gate ablation studies • LSTM vs GRU comparison • BPTT length impact	**D2L:** 9.1-9.7, 10.1-10.4 **Materials:** LSTM/GRU papers, gradient tools	
Chapter 4 Transformers	**Module 8: Attention Revolution**	**23:** Attention mechanism **24:** Encoder–decoder **25:** Alignment learning	• Attention as soft dictionary lookup • Query–key–value framework • Bahdanau attention • Seq2seq with attention • Attention weight interpretation • Machine translation success	• Implement attention • Translation w/ attention • Weight visualization • "Where models look" analysis • Attention ablation • Translation demo	**D2L:** 10.5–10.7, 11.1–11.4 **Materials:** Translation datasets, attention visualizers	
	Module 9: Transformers: Architecture & Optimization	**26:** Self-attention mechanics **27:** Transformer optimization **28:** Scaling strategies	• Multi-head self-attention • Layer norm placement • LR warm-up necessity • Attention dropout • Positional encoding ablations • Efficiency tricks	• Transformer from scratch • Layer norm ablations • Warm-up experiments • Attention pattern analysis • Reproduce "Attention is All You Need" • Efficiency vs performance study	**D2L:** 11.5–11.7 **Materials:** Original paper, implementation guides	

Continued on next page

Table 1 – continued from previous page

Chapter	Module	Learning Goals	Learning Objectives	Learning Activities	Instructional Materials	Delegate
	Module 10: Vision Transformers & Cross-Modal Learning	**29:** Images as sequences **30:** Patch embeddings **31:** Unified architectures	• ViT design • Patch tokenization • CNN vs ViT comparison • Cross-modal architectures • CLIP-style training • Attention in vision tasks	• ViT from scratch • Patch size experiments • ViT vs ResNet comparison • Attention map visualization • "One architecture to rule all" • Multimodal project	**D2L:** 11.8 **Materials:** ViT papers, CLIP resources, vision tools	
	Module 11: Pretraining, Fine-tuning & Scale	**32:** Pretraining strategies **33:** BERT and GPT **34:** Scaling laws	• Self-supervised pretraining • Masked language modeling • GPT-style autoregressive training • Fine-tuning methods • Parameter-efficient FT • Scaling laws & emergence	• Fine-tune BERT • Prompt engineering lab • LoRA implementation • "Size matters" study • Efficient FT comparison • Build task-specific model	**D2L:** 11.9, 15.8–15.10, 16.6–16.7 **Materials:** Pretrained models, scaling papers	
Chapter 5 Generative Models	**Module 12: Generative Models & Future Frontiers**	**35:** Generation paradigms **36:** Diffusion models **37:** Ethical considerations	• VAE latent space intuition • GAN adversarial dynamics • Diffusion forward/reverse • Stable Diffusion architecture • Evaluation metrics • Bias, safety, ethics	• Train small GAN • VAE latent exploration • Diffusion demo • Text-to-image generation • "With great power…" ethics • Future directions brainstorm	**D2L:** 4.7.5, 20.1–20.2 **Materials:** Generative demos, diffusion tutorials, ethics readings	