MIDDLE EAST TECHNICAL UNIVERSITY

SEMESTER I EXAMINATION 2024-2025

CENG 403 – Deep Learning - CNN Architectures & RNN Introduction (University Sources)

January 2025 TIME ALLOWED: 3 HOURS

INSTRUCTIONS TO CANDIDATES

1. This examination paper contains **SEVEN (7)** questions and comprises **TEN (10)** printed pages.
2. Answer all questions. The marks for each question are indicated at the beginning of each question.
3. Answer each question beginning on a **FRESH** page of the answer book.
4. This **IS NOT an OPEN BOOK** exam.
5. Show all mathematical derivations clearly with proper notation.
6. For architectural diagrams, draw clear and labeled components.
7. Calculate all requested parameters and show intermediate steps.
8. Explain computational complexity where requested.

Question 1. CNN Architectural Fundamentals and Calculations (25 marks)
Based on D2L.ai and university CNN course materials covering computational aspects.

(a) For a convolutional layer with the following specifications, calculate the output dimensions and number of parameters: (12 marks)

- Input: 224×224×3 RGB image
- 64 filters of size 7×7
- Stride: 2
- Padding: 3
- Bias terms included

Show all calculations including:

- Output height and width
- Total number of parameters
- Memory requirements for storing activations

(b) Explain the difference between ”Valid Padding” and ”Same Padding” in CNNs. For a 12×12 input with a 3×3 filter and stride 1: (8 marks)

- Calculate output size with valid padding
- Calculate padding needed for same padding
- Discuss trade-offs between the two approaches

(c) Compare parameter sharing in CNNs versus fully connected networks. For an image of size 256×256×3, calculate the number of parameters needed for: (5 marks)

- First layer as fully connected (to 512 units)
- First layer as convolutional (64 filters, 5×5)
- Explain the computational advantage

Question 2. ResNet Architecture and Skip Connections (30 marks)
Based on university deep learning courses and D2L.ai educational content.

(a) Explain the mathematical foundation of residual learning. Given a target function $H(x)$, derive why learning the residual mapping $F(x) = H(x) - x$ is easier than learning $H(x)$ directly. (10 marks)

Include discussion of:

- Identity function learning difficulty
- Gradient flow advantages
- Why zero functions are easier to learn

(b) Design and draw a complete ResNet basic block showing: (12 marks)

- Two 3×3 convolutional layers
- Skip connection implementation
- Activation function placement
- Dimension matching considerations

Compare this with a bottleneck block design (1×1, 3×3, 1×1 structure).

Draw ResNet Basic Block

Include: Conv layers, skip connections, activations

(c) Analyze gradient flow in ResNet vs. vanilla deep networks. For a 50-layer network, explain mathematically why ResNet can avoid vanishing gradients. (8 marks)

Include:

- Gradient computation through skip connections
- Comparison with traditional deep networks
- Why identity mappings preserve gradient magnitude

Question 3. DenseNet and Advanced CNN Architectures (22 marks)
Based on modern CNN architecture research and educational materials.

(a) Compare DenseNet with ResNet architectures. Explain the key difference: (8 marks)

$$\text{ResNet: } x_l = H_l(x_{l-1}) + x_{l-1}$$

$$\text{DenseNet: } x_l = H_l([x_0, x_1, \ldots, x_{l-1}])$$

Discuss advantages and disadvantages of each approach.

(b) For a DenseNet block with 4 layers, each producing 12 feature maps (growth rate k=12), and input of 64 channels: (10 marks)

- Calculate the number of input channels for each layer
- Compute total memory requirements for concatenations
- Explain how transition layers reduce dimensionality
- Calculate parameters for 1×1 conv in transition layer

(c) Design a Highway Network gate mechanism. Write the mathematical equations for: (4 marks)

$$y = H(x, W_H) \cdot T(x, W_T) + x \cdot C(x, W_C)$$

Explain how this differs from standard residual connections.

Question 4. CNN Optimization and Efficiency (20 marks)
Based on practical CNN implementation and optimization techniques.

(a) Analyze binary neural networks for edge deployment. Given a standard CNN with: (10 marks)

- 10M parameters (32-bit floats)
- 50 GFLOPS for inference

Calculate:

- Memory reduction with binary weights
- Speed improvement estimates
- Accuracy trade-offs to consider
- When binary networks are appropriate

(b) Compare different normalization strategies in deep CNNs: (10 marks)

- Batch Normalization: benefits and limitations
- Why BatchNorm helps in very deep networks (10,000+ layers)
- Relationship between BatchNorm and gradient stability
- Alternative normalization methods

Question 5. RNN Fundamentals and Unfolding (28 marks)
Based on sequence modeling and RNN theory from university courses.

(a) Classify the following problems and suggest appropriate architectures: (8 marks)

- Image captioning
- Spam email detection
- Machine translation
- Real-time speech recognition

For each, specify: one-to-one, one-to-many, many-to-one, or many-to-many architecture.

(b) Explain RNN unfolding process. For the recurrent equation: (12 marks)

$$h_t = \tanh(W_{hx}x_t + W_{hh}h_{t-1} + b_h)$$

$$y_t = W_{hy}h_t + b_y$$

Draw the unfolded network for T=3 time steps showing:

- Weight sharing across time
- Hidden state connections
- How this becomes a feedforward network
- Why sequences of different lengths can be handled

(c) Discuss the Turing completeness of RNNs. Explain: (8 marks)

- What it means for RNNs to be Turing complete
- The role of recurrent connections in providing memory
- Difference between theoretical capacity and practical training
- Comparison with multilayer perceptrons as universal approximators

Question 6. RNN Training and Gradient Issues (25 marks)

Based on RNN training theory and backpropagation through time.

(a) Explain why hyperbolic tangent is preferred over ReLU in vanilla RNNs. Discuss: (8 marks)

- Need for bounded hidden state values
- Consistency of state representation across time
- Problems with unbounded activations in recurrent connections
- Trade-offs with gradient flow

(b) For an RNN unfolded for 100 time steps, analyze the vanishing gradient problem: (12 marks)

- Why this creates a 100-layer feedforward network
- Mathematical explanation of gradient diminishing
- Effect of squashing activation functions
- Impact on learning long-term dependencies

Include analysis of gradient computation:

$$\frac{\partial L}{\partial W_{hh}} = \sum_{t=1}^{T} \frac{\partial L_t}{\partial h_t} \frac{\partial h_t}{\partial W_{hh}}$$

(c) Compare solutions to RNN gradient problems: (5 marks)

- Gradient clipping for exploding gradients
- Penalty terms for vanishing gradients
- When each approach is appropriate

Question 7. LSTM Architecture and Memory Mechanisms (30 marks)
Based on LSTM theory and gating mechanisms for sequence modeling.

(a) Design the complete LSTM architecture with mathematical equations. For input x_t, previous hidden state h_{t-1}, and previous cell state C_{t-1}, derive: (15 marks)

- Forget gate: $f_t =?$
- Input gate: $i_t =?$
- Candidate values: $\tilde{C}_t =?$
- Cell state update: $C_t =?$
- Output gate: $o_t =?$
- Hidden state: $h_t =?$

(b) Explain why LSTM solves the vanishing gradient problem. Focus on: (10 marks)

- Gradient flow through the cell state path
- Why $\frac{\partial C_t}{\partial C_{t-1}}$ doesn't involve squashing functions
- How this enables learning of long-term dependencies
- Mathematical comparison with vanilla RNN gradient flow

(c) Compare LSTM variants: (5 marks)

- LSTM with peephole connections
- Coupled forget and input gates
- GRU vs LSTM trade-offs
- When to choose each variant

END OF PAPER