Indian Statistical Institute

End Semester Examination

M. Tech. (CS), Machine Learning II

Maximum Marks: 100 Duration: 3 Hrs. 15 Mins.

Note:

- Attempt all questions.
- Assume suitable values wherever necessary and state assumptions clearly.
- All notations follow standard definitions used in deep learning and transformer architectures.
- Use of calculator is permitted.

---

1. Consider a linear autoencoder defined as follows:

$$x \in \mathcal{R}^n, \quad h = W_1 x, \quad \hat{x} = W_2 h,$$

where $W_1 \in \mathcal{R}^{m \times n}$ is the encoder weight matrix and $W_2 \in \mathcal{R}^{n \times m}$ is the decoder weight matrix.

(a) If the hidden layer dimension $m = n$, determine what the product $W_2 W_1$ must be for perfect reconstruction of all inputs x.

(b) If $m < n$, explain mathematically why perfect reconstruction of all inputs is not possible, and describe what $W_2 W_1$ represents in this case.

(c) Briefly interpret the geometric meaning of part (b): what kind of subspace does the autoencoder learn when $m < n$?

[3 + 3 + 2 = 8]

2. Let $p(k)$ be a one-dimensional discrete probability distribution defined on nonnegative integers $k = 0, 1, 2, \dots$, which we wish to approximate by a Poisson distribution $q(k; \lambda)$ having parameter $\lambda > 0$. The Poisson distribution is given by

$$q(k; \lambda) = \frac{e^{-\lambda} \lambda^k}{k!}, \quad k = 0, 1, 2, \dots$$

(a) Show that minimizing the KL divergence $KL(p || q_\lambda)$ with respect to λ leads to

$$\lambda^* = \sum_{k=0}^{\infty} k\, p(k),$$

that is, the optimal Poisson parameter equals the mean of $p(k)$. Provide all intermediate steps clearly.

(b) Explain intuitively why this result implies that the "best" Poisson approximation to any discrete nonnegative distribution matches its expected value.

(c) Show that $KL(p; q_\lambda)$ is a convex function of λ, and hence verify that the stationary point $\lambda^* = E_p[k]$ corresponds to a unique global minimum.

[6 + 3 + 3 = 12]

3. (a) A simple Recurrent Neural Network (RNN) operates on an input sequence

$$x = [x_1, x_2] = [1, 2],$$

with parameters:

$$W_x = 1.0, \quad W_h = 0.5, \quad b = 0, \quad \text{and activation } f(z) = \tanh(z).$$

Assume the initial hidden state $h_0 = 0$.

i. Compute h_1 and h_2 step-by-step, showing all intermediate values before and after applying tanh.

ii. If the output is computed as $y_t = h_t$, write the final output sequence $[y_1, y_2]$.

(b) Consider an LSTM cell with the same input sequence $x = [1, 2]$ and the following parameters:

$$W_i = W_f = W_o = W_c = 1, \quad U_i = U_f = U_o = U_c = 0, \quad b_i = b_f = b_o = b_c = 0,$$

and the initial states $h_0 = 0$, $c_0 = 0$. Use the activation functions $\sigma(z) = \frac{1}{1+e^{-z}}$ and $\tanh(z)$.

i. Compute the gate activations (i_t, f_t, o_t) and candidate cell state $(\tilde{c}_t)$ for $t = 1, 2$.

ii. Compute the cell states c_1, c_2 and hidden states h_1, h_2.

iii. Compare how the LSTM's memory cell c_t differs from the RNN's hidden state in preserving information across time steps.

[(4+4)+(4+5+3) = 20]

4. A Transformer's Scaled Dot-Product Attention mechanism operates using three matrices: Query (Q), Key (K), and Value (V). You are given the following input matrices for a single attention head:

$$Q = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix}, \quad K = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}, \quad V = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

(a) Compute the unnormalized attention scores.

(b) Apply the appropriate scaling factor to the computed scores to obtain the scaled attention scores.

(c) Calculate the normalized attention weights using the Softmax function applied to the scaled scores.

(d) Using the obtained attention weights, compute the final attention output vector as a weighted sum of the value vectors.

[4+6=10]

7. (a) Consider a graph G with node indices $\{0, 1, 2, 3, 4\}$ and edge indices given by:

$$\begin{bmatrix} 0 & 0 & 0 & 1 & 2 & 2 & 3 & 3 & 3 & 4 \\ 1 & 2 & 3 & 0 & 0 & 3 & 0 & 2 & 4 & 3 \end{bmatrix}$$

Draw the computation graph for node ID 2 for a 2-hop message passing operation. Assume the node feature matrix of G is:

$$X = \begin{bmatrix} 0 & -1 \\ 1 & -2 \\ 0 & 2 \\ 1 & 0 \\ -1 & 3 \end{bmatrix}$$

Estimate the aggregated features for node 2 using a bottom-to-top approach with the mean aggregator (no self-loop addition required). Further, compute A^2X, where A is the normalized adjacency matrix, to aggregate two-hop node features. Show that the updated node features for node 2 are equal under both aggregation approaches.

(b) Two nodes p and q in a connected graph respectively have degrees m and n. Assume all nodes possess identical features. If neighborhood aggregation is performed, design two different aggregators that yield distinct neighborhood representations for nodes p and q under the following cases:

(a) $m = n$

(b) $m \neq n$

(c) Let L be a graph Laplacian with two eigenvectors

$$u = \begin{bmatrix} -1 & 3a & b \end{bmatrix}, \quad v = \begin{bmatrix} 2b & c & -a \end{bmatrix}$$

corresponding to eigenvalues greater than zero, where $a, b, c \in R$. Find the values of a, b, and c that satisfy orthogonality and normalization conditions of Laplacian eigenvectors.

(d) Discuss three advantages of employing **Knowledge Distillation** from a larger (teacher) network to a smaller (student) network, in comparison to designing an end-to-end trainable compact model.

[8 + 4 + 5 + 3 = 20]

8. (a) A binary classifier predicts whether a candidate is *selected* ($\hat{Y} = 1$) or *not selected* ($\hat{Y} = 0$) in a job interview for the position of Data Scientist. The following table shows predictions for 12 candidates, equally divided between males and females.

Candidate ID	Gender	Predicted Outcome ($\hat{Y}$)
1	Female	0
2	Male	1
3	Male	1
4	Female	1
5	Male	1
6	Female	0
7	Female	0
8	Male	0
9	Female	1
10	Male	1
11	Female	0
12	Male	1

i. Compute the *Demographic Parity Difference (DPD)* between male and female candidates.

ii. Interpret your result: which gender group receives more favorable predictions?

(b) A probabilistic language model assigns conditional probabilities to each token in a sequence $(w_1, w_2, \ldots, w_N)$. The perplexity (PP) of the model is defined as:

$$\mathrm{PP} = \exp\left(-\frac{1}{N}\sum_{i=1}^{N}\log P(w_i \mid w_1, \ldots, w_{i-1})\right).$$

i. Derive this expression starting from the cross-entropy formulation

$$H(p, q) = -\frac{1}{N}\sum_{i=1}^{N}\log q(w_i).$$

ii. Compute the perplexity for a 4-token sequence where the model assigns probabilities $P(w_1) = 0.4$, $P(w_2|w_1) = 0.3$, $P(w_3|w_1, w_2) = 0.2$, and $P(w_4|w_1, w_2, w_3) = 0.1$.

iii. Interpret what this perplexity value implies about the model's confidence and prediction quality.

[(4+2)+(2+2+2) = 12]

[4 + 5 + 4 + 7 = 20]

5. Explainable AI techniques such as LIME and SHAP help interpret the contribution of individual input features to a model's prediction. Consider a simple linear regression model used for predicting a student's exam score based on two features:

$$f(x_1, x_2) = 5 + 3x_1 + 2x_2$$

where x_1 = hours studied and x_2 = number of practice tests taken. We wish to explain the prediction for a specific student with input $(x_1, x_2) = (2, 1)$.

   (a) Using the SHAP framework, assume the following coalition function values (model outputs when only a subset of features is present):

$$\begin{aligned} f(\emptyset) &= 5, \\ f(\{x_1\}) &= 11, \\ f(\{x_2\}) &= 9, \\ f(\{x_1, x_2\}) &= 15. \end{aligned}$$

   Compute the Shapley values ϕ_{x_1} and ϕ_{x_2} for this prediction and verify that:

$$f(x_1, x_2) = f(\emptyset) + \phi_{x_1} + \phi_{x_2}.$$

   (b) Interpret the computed Shapley values: Which feature contributes more to the model's output for this particular prediction, and by how much relative to the baseline?

[5+ 3 = 8]

6. Let $s \in R^n$ be a vector of attention scores with components s_j, and define the *softmax* function

$$a = \text{softmax}(s), \qquad a_i = \frac{e^{s_i}}{\sum_{t=1}^{n} e^{s_t}}, \qquad i = 1, \ldots, n.$$

The vector a represents the attention weights corresponding to s.

   (a) Prove that the softmax function is invariant to adding the same constant to all entries of s; that is, for any scalar $c \in R$,

$$\text{softmax}(s + c\mathbf{1}) = \text{softmax}(s),$$

   where $\mathbf{1}$ denotes the all-ones vector.

   (b) Derive the *Jacobian matrix* of the softmax function, defined as

$$J_s = \frac{\partial a}{\partial s} \in R^{n \times n}, \quad \text{where} \quad (J_s)_{ij} = \frac{\partial a_i}{\partial s_j}.$$

   Show that $J_s = \text{diag}(a) - aa^\top$ and hence that for all i, j,

$$\frac{\partial a_i}{\partial s_j} = a_i(\delta_{ij} - a_j),$$

   where δ_{ij} is the Kronecker delta.

   (Hint: Use the quotient rule starting from $a_i = e^{s_i} / \sum_t e^{s_t}$.)