CSE556: NLP (Quiz-3)

Duration: 15mins **Max Marks: 10**

1. Define [3]
   - **Zero-shot learning:** The model is given zero example to refer/learn from before making predictions on unseen samples. No training happens.
   - **One-shot learning:** The model is given one example to refer/learn from before making predictions on unseen samples.
   - **Few-shot learning:** The model is given more than one example to refer/learn from before making predictions on unseen samples.

2. Why do we use an Add layer in Transformer? [1]

   The add layer in Transformer acts as a residual/skip connection, which allows gradients to propagate back to the lower layer through a bypass path.

3. Write steps (incl equations, if possible) of Bahadanu's and Vaswani's Attention mechanism. Do NOT draw the encoder-decoder architecture. [6]

<table>
<tr>
<td>
Bahadanu's (Standard attention)

1. Compute hidden representations h_j
2. Compute attention model
$$e_{ij} = a(s_{i-1}, h_j)$$
3. Compute softmax (attention weights)
$$a_{ij} = \frac{\exp(e_{ij})}{\sum \exp(e_{ik})}$$
4. Compute context vector
$$c_i = \sum a_{ij} h_j$$
</td>
<td>
Vaswani's (Self-attention)

1. Compute query (Q), key (K) and value (V) through projection.
$$Q = W_Q . X$$
$$K = W_K . X$$
$$V = W_V . X$$
2. Compute dot product between a query vector and all other key vectors.
$$\forall_j\, s_{ij} = q_i\, k_j$$
3. Scale by sqrt(d_k)
$$s'_{ij} = s_{ij} / \text{sqrt}(d_k)$$
4. Compute softmax (attention weights)
$$a_{ij} = Softmax(s'_{ij})$$
5. Compute attended value vectors
$$v'_{ij} = a_{ij} . v_{ij}$$
6. Sum all value vectors
$$\sum_j v'_{ij}$$
</td>
</tr>
</table>