3. Latent Mediator Transformer

3.1 Problem Definition

Let

$$F_q \in \mathbb{R}^{B \times N_q \times C}, \quad F_s \in \mathbb{R}^{B \times N_s \times C}$$

denote query and support feature embeddings extracted from a shared backbone, where:

- B: batch size
- N_q, N_s: number of tokens
- C: feature dimension

Unlike conventional transformers that directly compute cross-attention between F_q and F_s, we introduce a learnable latent mediator space.

Define mediator tokens:

$$M \in \mathbb{R}^{B \times N_m \times C}$$

where $N_m \ll N_q, N_s$.

The mediator serves as a latent knowledge bottleneck that governs bidirectional information exchange.

3.2 Mediator Knowledge Aggregation

The mediator first aggregates information jointly from query and support.

Concatenate context features:

$$F_c = [F_q; F_s] \in \mathbb{R}^{B \times (N_q + N_s) \times C}$$

Project to query, key, value spaces:

$$Q_m = MW_Q^m$$

$$K_c = F_c W_K^c$$

$$V_c = F_c W_V^c$$

where:

$$W_Q^m, W_K^c, W_V^c \in \mathbb{R}^{C \times C}$$

Compute mediator attention:

$$A_{mc} = \mathrm{softmax}\left(\frac{Q_m K_c^\top}{\sqrt{C}}\right)$$

Mediator update:

$$\tilde{M} = A_{mc} V_c$$

Residual update:

$$M^{(1)} = M + \tilde{M}$$

Feed-forward refinement:

$$M^{(1)} = M^{(1)} + \mathrm{FFN}(M^{(1)})$$

3.3 Mediator-to-Query Knowledge Projection

The refined mediator distributes knowledge to query features.

Projection:

$$Q_q = F_q W_Q^q$$

$$K_m = M^{(1)} W_K^m$$

$$V_m = M^{(1)} W_V^m$$

Attention:

$$A_{qm} = \mathrm{softmax}\left(\frac{Q_q K_m^\top}{\sqrt{C}} \right)$$

Query update:

$$\tilde{F}_q = A_{qm} V_m$$

$$F_q^* = F_q + \tilde{F}_q$$

Feed-forward refinement:

$$F_q^* = F_q^* + \mathrm{FFN}(F_q^*)$$

3.4 Mediator-to-Support Knowledge Projection

Similarly, support features are updated.

Projection:

$$Q_s = F_s W_Q^s$$

$$K_m = M^{(1)} W_K^m$$

$$V_m = M^{(1)} W_V^m$$

Attention:

$$A_{sm} = \mathrm{softmax}\left(\frac{Q_s K_m^\top}{\sqrt{C}}\right)$$

Support update:

$$\tilde{F}_s = A_{sm} V_m$$

$$F_s^* = F_s + \tilde{F}_s$$

Feed-forward refinement:

$$F_s^* = F_s^* + \mathrm{FFN}(F_s^*)$$

3.5 Iterative Mediator Evolution

The mediator evolves over transformer layers:

$$M^{(l+1)} = M^{(l)} + \mathrm{Attn}\left(M^{(l)}, [F_q^{(l)}; F_s^{(l)}]\right) + \mathrm{FFN}(M^{(l)})$$

Query update:

$$F_q^{(l+1)} = F_q^{(l)} + \mathrm{Attn}(F_q^{(l)}, M^{(l+1)}) + \mathrm{FFN}(F_q^{(l)})$$

Support update:

$$F_s^{(l+1)} = F_s^{(l)} + \mathrm{Attn}(F_s^{(l)}, M^{(l+1)}) + \mathrm{FFN}(F_s^{(l)})$$