

Adaptation Prompt Tuning

Instructor

Sourab Mangulkar

Machine Learning Engineer at 
Creator of  PEFT



Adaptation Prompt Tuning (Llama Adapter Paper)

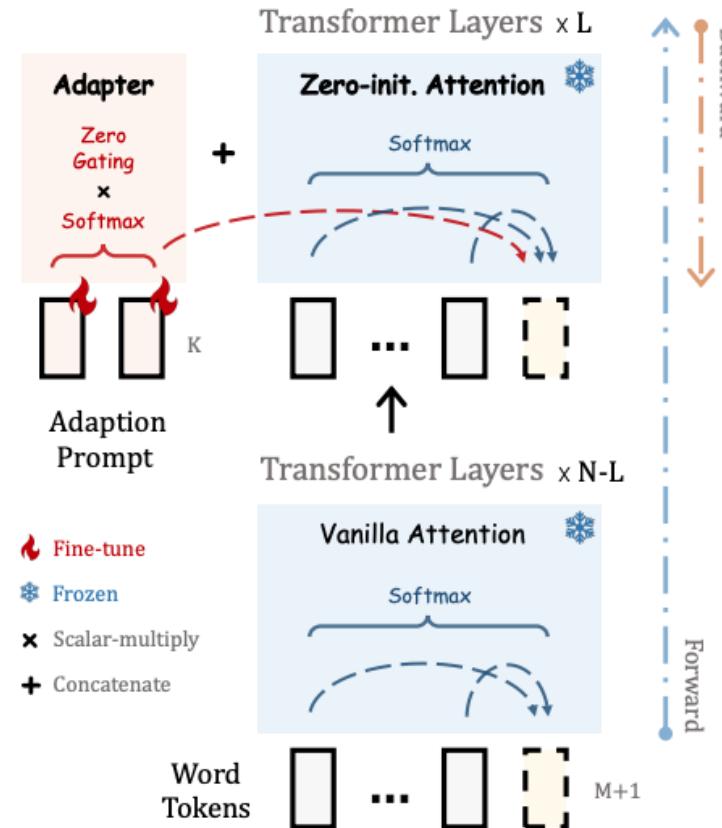


FIGURE EXPLAINING HOW ADAPTION PROMPT TUNING WORKS FROM [ORIGINAL PAPER](#), FIGURE 2

Adaptation Prompt Tuning (Llama Adapter Paper)

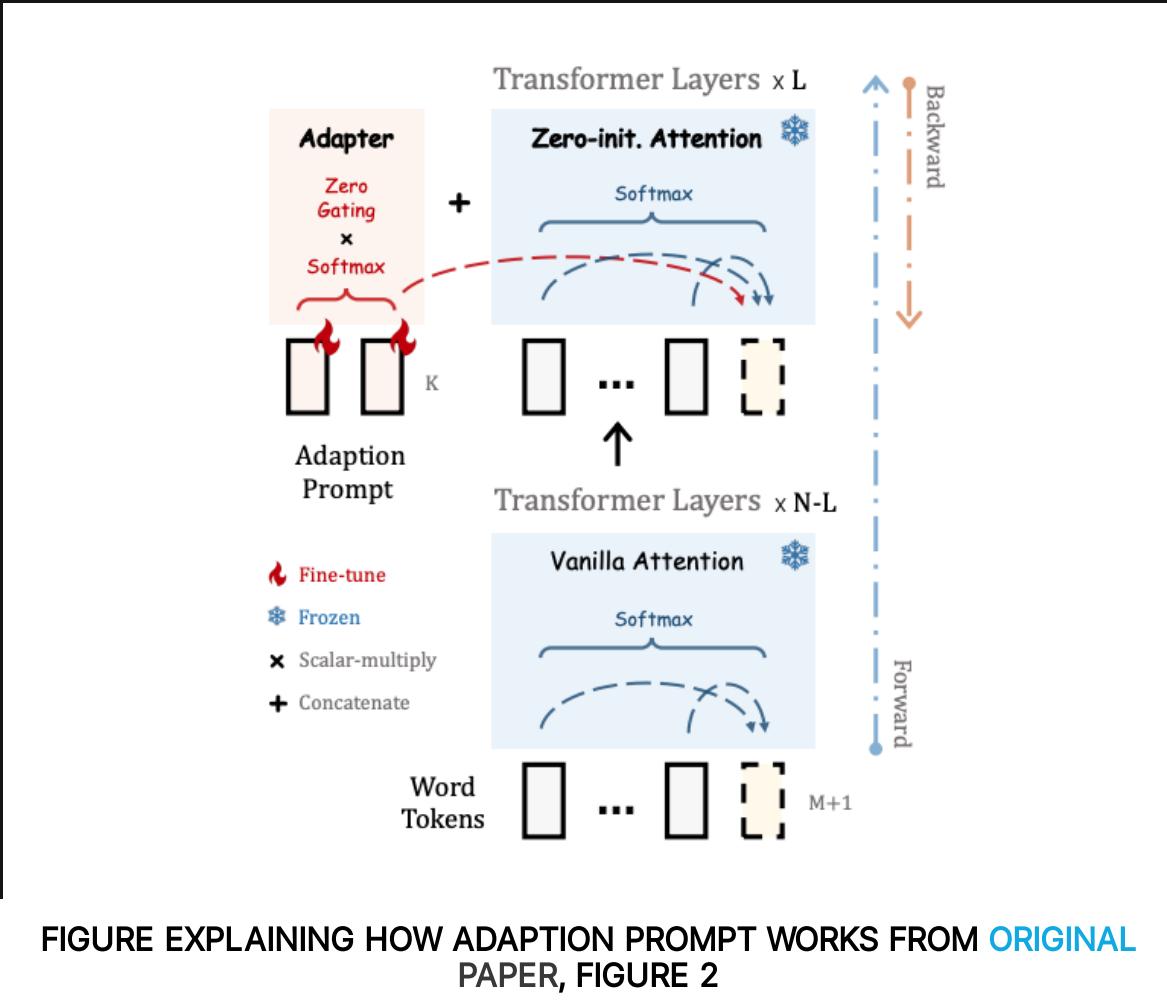


FIGURE EXPLAINING HOW ADAPTION PROMPT WORKS FROM ORIGINAL PAPER, FIGURE 2

```
class AdaptationPromptAttentionLayer(torch.nn.Module):
    def __init__(self, base_attention_layer, num_prompt_tokens, emb_dim, batch_size):
        super().__init__()
        self.base_attention_layer = base_attention_layer
        self.prompt_embedding = torch.nn.Embedding(num_prompt_tokens, emb_dim)
        self.adaption_gate = torch.nn.Parameter(torch.zeros(1))
        self.prompt_tokens = (torch.arange(num_prompt_tokens)
                             .unsqueeze(0)
                             .repeat(batch_size, 1)
                             .long())

    def forward(self, hidden_states, **kwargs):
        hidden_state, query_states =
            self.base_attention_layer(hidden_states=hidden_states, **kwargs)
        d = self.base_attention_layer.head_dim
        bsz, q_len = query_states.shape[0], query_states.shape[1]
        # get the soft prompt embeddings
        soft_prompts = self.prompt_embedding(self.prompt_tokens)
        # get the adaption k, v projections
        adaptation_k = self.base_attention_layer.key_layer(soft_prompts)
        adaptation_v = self.base_attention_layer.value_layer(soft_prompts)
        # get adaption attention scores
        scores = torch.matmul(query_states, adaptation_k.transpose(2, 3))/math.sqrt(d)
        scores = self.adaption_gate * torch.nn.functional.softmax(scores, dim=-1)
        # compute the adaption output values
        # (bsz, q_len, num_heads * head_dim)
        adaptation_output = torch.matmul(scores, adaptation_v).transpose(1,
                2).reshape(bsz, q_len, -1)

        hidden_state = hidden_state + adaptation_output
        return hidden_state
```