LLMs 激活函数篇

来自： AiGC面试宝典

宁静致远

2023年09月29日 12:41

1 介绍一下 FFN 块 计算公式?

$$FFN(x) = f(xW_1 + b_1)W_2 + b_2$$

2 介绍一下 GeLU 计算公式?

$$GeLU(x) \approx 0.5x(1 + tanh(\sqrt{\frac{2}{\pi}}(x + 0.044715x^3)))$$

3 介绍一下 Swish 计算公式?

$$Swish_\beta(x) = x \cdot \sigma(\beta x)$$

> 2个可训练权重矩阵，中间维度为 4h

4 介绍一下 使用 GLU 线性门控单元的 FFN 块 计算公式?

$$GLU(x) = \sigma(xW + b) \otimes xV$$
$$FFN_{GLU} = (f(xW_1) \otimes xV)W_2$$

5 介绍一下 使用 GeLU 的 GLU 块 计算公式?

$$GeGLU(x) = GeLU(xW) \otimes xV$$

6 介绍一下 使用 Swish 的 GLU 块 计算公式?

$$SwiGLU = Swish_\beta(xW) \otimes xV$$

> 3个可训练权重矩阵，中间维度为 4h*2/3

各LLMs 都使用哪种激活函数?

模型	激活函数
GPT3	GeLU
LLaMA	SwiGLU
LLaMA2	SwiGLU
baichuan	SwiGLU
ChatGLM-6B	GeLU
ChatGLM2-6B	SwiGLU
Bloom	GeLU
Falcon	GeLU

4h = 4*4096 = 16384

2/3 * 4h = 10022 -> 11008

11008/128 = 86

LLaMA -7B

Modules	params_shape	params_num
model.embed_tokens.weight	[32000, 4096]	131072000
model.layers.0.self_attn.q_proj.weight	[4096, 4096]	16777216
model.layers.0.self_attn.k_proj.weight	[4096, 4096]	16777216
model.layers.0.self_attn.v_proj.weight	[4096, 4096]	16777216
model.layers.0.self_attn.o_proj.weight	[4096, 4096]	16777216
model.layers.0.mlp.gate_proj.weight	[11008, 4096]	45088768
model.layers.0.mlp.down_proj.weight	[4096, 11008]	45088768
model.layers.0.mlp.up_proj.weight	[11008, 4096]	45088768
model.layers.0.input_layernorm.weight	[4096]	4096
model.layers.0.post_attention_layernorm.weight	[4096]	4096

BLOOM -7B

Modules	params_shape	params_num
transformer.word_embeddings.weight	[250880, 4096]	1027604480
transformer.word_embeddings_layernorm.weight	[4096]	4096
transformer.word_embeddings_layernorm.bias	[4096]	4096
transformer.h.0.input_layernorm.weight	[4096]	4096
transformer.h.0.input_layernorm.bias	[4096]	4096
transformer.h.0.self_attention.query_key_value.weight	[12288, 4096]	50331648
transformer.h.0.self_attention.query_key_value.bias	[12288]	12288
transformer.h.0.self_attention.dense.weight	[4096, 4096]	16777216
transformer.h.0.self_attention.dense.bias	[4096]	4096
transformer.h.0.post_attention_layernorm.weight	[4096]	4096
transformer.h.0.post_attention_layernorm.bias	[4096]	4096
transformer.h.0.mlp.dense_h_to_4h.weight	[16384, 4096]	67108864
transformer.h.0.mlp.dense_h_to_4h.bias	[16384]	16384
transformer.h.0.mlp.dense_4h_to_h.weight	[4096, 16384]	67108864
transformer.h.0.mlp.dense_4h_to_h.bias	[4096]	4096