SLOT方法实现--基于Qwen模型

SLOT方法的核心思路是：

- 对任意输入的 prompt，只使用原始模型进行前向推理；
- 在不改动模型权重的前提下，仅对生成过程中的隐藏状态（如最后一层 hidden state）进行扰动；
- 引入一个可训练的小型参数 `delta`，其形状为 `[1, 1, hidden_size]`；
- `delta` 是在 prompt 自身的预测任务上训练得到的，其目标是提升模型对下一个 token 的预测能力；
- 训练完成后，在生成过程中将该 `delta` 加到隐藏表示上，进而影响输出 token，达到引导生成的目的。

该方法具备如下优点：

- **无需微调大模型参数，成本低**
- **适用于任意任务 prompt，无需重构模型结构**
- **可以灵活插拔，引导模型按预期生成**

In [1]:
```
import torch
from transformers import AutoTokenizer, AutoModelForCausalLM
import torch
import torch.nn as nn
from tqdm import tqdm
import os
import matplotlib.pyplot as plt
from skimage import io
import seaborn as sns
import warnings
import numpy as np
import warnings
import pandas as pd
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt
import warnings
```

```
from pylab import mpl, plt
import matplotlib.patches as mpatches
from tqdm import tqdm
from tqdm.notebook import tqdm

# best font and style settings for notebook
warnings.filterwarnings('ignore')
sns.set_style("white")
mpl.rcParams['font.family'] = 'MiSans'
```

加载Qwen-0.6B模型

In [2]:
```
model_path = r"D:\pythonProject\DeepSeek\Recsys\AnimeLLMRec\Qwen3-0.6B"  # modify to your Qwen Path
tokenizer = AutoTokenizer.from_pretrained(model_path)
model = AutoModelForCausalLM.from_pretrained(model_path).to("cuda" if torch.cuda.is_available() else "cpu")
```

输出最后一层 H state

In [3]:
```
# === 输入 prompt ===
prompt = "请你详细介绍一下西湖。"
inputs = tokenizer(prompt, return_tensors="pt").to(model.device)

# === 前向传播，输出包含 hidden_states ===
with torch.no_grad():
    outputs = model(**inputs, output_hidden_states=True, return_dict=True)
# === 获取倒数第一层的 Hidden States H（transformer 最后一层输出）===
# shape: [1, seq_len, hidden_size]
H = outputs.hidden_states[-1]
H.shape
```

Out[3]:
```
torch.Size([1, 5, 1024])
```

查看Token的切割

In [4]:
```python
from preprocess import get_token_alignment_df

prompt = """你是一位专业的日志分析专家，请基于以下日志条目，提取其中的异常模式、潜在根因，并提出清晰的修复建议。
请注意以下要求：
- **必须**使用人的自然语言习惯进行简洁清晰的表达，避免冗长术语；
- **每一个异常判断必须引用原始日志内容**，说明异常来源；
- 分析结果应**按逻辑分段列出**，便于阅读与排查。
以下是异常日志列表：
- 2025-04-23 14:42:36.060 [INFO] database P0000010414
请你进行分析。
"""

get_token_alignment_df(prompt=prompt, tokenizer=tokenizer).head()
```

Out[4]:

	Index	Token ID	Token
0	0	56568	你
1	1	109182	是一位
2	2	104715	专业的
3	3	8903	日
4	4	77128	志

输入 Prompt，得到W_vocab 和 H state矩阵

In [5]:
```python
# === 输入 prompt ===
prompt = "请你详细介绍一下西湖。"
inputs = tokenizer(prompt, return_tensors="pt").to(model.device)
input_ids = inputs["input_ids"][0]  # 去掉 batch 维度
tokens = tokenizer.convert_ids_to_tokens(input_ids)

# === 前向传播，输出包含 hidden_states ===
with torch.no_grad():
    outputs = model(**inputs, output_hidden_states=True, return_dict=True)
# === 获取倒数第一层的 Hidden States H（transformer 最后一层输出）===
# shape: [1, seq_len, hidden_size]
```

```
H = outputs.hidden_states[-1]
# === 获取 W_vocab 权重矩阵 ===
# 对于 GPT 类模型，输出 logits 是 H @ W_vocab.T

# === 获取 vocab 权重矩阵 ===
W_vocab = model.lm_head.weight  # shape: [vocab_size, hidden_size]
W_vocab.shape
```

Out[5]: torch.Size([151936, 1024])

Prompt自我预测下一个token

In [6]:
```
top_k = 5


def get_top_k_predict(prompt, H_state, model, tokenizer, top_k=5):
    inputs = tokenizer(prompt, return_tensors="pt").to(model.device)
    input_ids = inputs["input_ids"][0]  # 去掉 batch 维度
    tokens = tokenizer.convert_ids_to_tokens(input_ids)
    W_vocab = model.lm_head.weight

    predict_next_tokens = []
    next_tokens = []
    for i, (token_id, token, h_vec) in enumerate(zip(input_ids.tolist(), tokens, H_state[0])):
        logits_i = torch.matmul(h_vec, W_vocab.T)  # [vocab_size]
        probs_i = torch.softmax(logits_i, dim=-1)
        topk = torch.topk(probs_i, k=top_k)

        top_ids = topk.indices.tolist()
        top_probs = topk.values.tolist()

        # ✅ 用 decode 解决乱码问题
        top_tokens = [tokenizer.decode([id]).replace(" ", "") for id in top_ids]

        char = tokenizer.decode(token_id)

        for j in range(len(topk[0])):
            next_tokens.append([i, char, top_tokens[j], top_ids[j], top_probs[j]])
    return pd.DataFrame(next_tokens, columns=['token_idx', 'char', 'char_predict', 'token_id', 'prob'])
```

```python
def pretty_print_top_k(df):
    """
    漂亮地打印 DataFrame 中所有 token 的 top-k 预测结果。

    参数:
    - df: DataFrame，包含 get_top_k_predict 的输出
    """
    token_indices = df['token_idx'].unique()
    for token_idx in token_indices:
        sub_df = df[df['token_idx'] == token_idx]
        if sub_df.empty:
            continue
        char = sub_df.iloc[0]['char']
        print(f"\n🔍 这是字符 '{char}' 的下一个 token 预测：")
        print(f"{'序号':<4} {'预测字符':<10} {'token_id':<10} {'概率':<10}")
        print("-" * 40)
        for i, row in enumerate(sub_df.itertuples(), 1):
            print(f"{i:<4} {row.char_predict:<10} {row.token_id:<10} {row.prob:<.6f}")


top_k = 3
top_k_df = get_top_k_predict(prompt=prompt, model=model, tokenizer=tokenizer, H_state=H, top_k=top_k)
pretty_print_top_k(top_k_df)
```

```
🔍 这是字符 '请你' 的下一个 token 预测：
序号    预测字符        token_id    概率
--------------------------------------
1     用          11622      0.071626
2     设计          70500      0.063536
3     帮我          108965     0.062522

🔍 这是字符 '详细' 的下一个 token 预测：
序号    预测字符        token_id    概率
--------------------------------------
1     解释          104136     0.276219
2     分析          101042     0.081168
3     地          29490      0.069583

🔍 这是字符 '介绍一下' 的下一个 token 预测：
序号    预测字符        token_id    概率
--------------------------------------
1     “          2073       0.019732
2     《          26940      0.013023
3     如何          100007     0.012535

🔍 这是字符 '西湖' 的下一个 token 预测：
序号    预测字符        token_id    概率
--------------------------------------
1     大学          99562      0.076652
2     的          9370       0.074637
3     龙          99465      0.052748

🔍 这是字符 '。' 的下一个 token 预测：
序号    预测字符        token_id    概率
--------------------------------------
1     ﹖          8908       0.402138
2                 220        0.124658
3     ﹖          6567       0.041067
```

通过 Prompt 自监督训练扰动向量 delta

我们将使用 prompt 自身的 token 序列进行训练，优化一个可学习的扰动向量 delta，使模型更准确地预测下一个 token。

- 输入 prompt: `[token_1, token_2, ..., token_n]`
- 原理：通过最小化预测 token[i+1] 的损失，引导模型在每个位置的预测更接近实际下一个 token。

对齐关系如下： logits[0] → token[1] logits[1] → token[2] logits[2] → token[3] ... logits[n-2] → token[n-1] logits[n-1] → token[n]

训练目标是使模型输出的每个 `logits[i]` 更靠近其对应位置的目标 token `token[i+1]`。

In [7]:
```
# === 输入 prompt ===
prompt = "请你详细介绍一下西湖。"
```

In [8]:
```
import torch
import torch.nn as nn
from tqdm import tqdm


def train_delta_from_H(model, tokenizer, prompt, H_state, step=3, lr=1e-2):
    """
    针对给定 prompt，通过优化隐藏状态 H 的扰动 delta，使模型更好地预测下一个 token。

    参数：
    - model: 语言模型（需具备 lm_head）
    - tokenizer: 分词器
    - prompt: 输入的文本 prompt（str）
    - step: 优化步数（默认 3）
    - lr: 学习率（默认 1e-2）

    返回：
    - delta: 训练得到的扰动向量，shape = [1, 1, hidden_size]
    """

    # === Step 1: 编码 Prompt，并转为模型输入 ===
    inputs = tokenizer(prompt, return_tensors="pt").to(model.device)
    input_ids = inputs["input_ids"]  # shape: [1, seq_len]

    # === Step 3: 构建目标 token 对（预测下一个 token）===
    # 当前 token 用于生成，目标 token 用于监督
    current_ids = input_ids[:, :-1]  # shape: [1, seq_len-1]
    target_ids = input_ids[:, 1:]  # shape: [1, seq_len-1]
```

```python
    # === Step 4: 初始化 delta（可训练参数）===
    hidden_size = H_state.size(-1)
    delta = nn.Parameter(torch.zeros((1, 1, hidden_size), device=H_state.device, requires_grad=True))

    # === Step 5: 设置优化器与损失函数 ===
    optimizer = torch.optim.Adam([delta], lr=lr)
    loss_fn = nn.CrossEntropyLoss()
    loss_log = []

    # === Step 6: 开始优化 delta 参数 ===
    for i in tqdm(range(step), desc="Training delta"):
        optimizer.zero_grad()

        # 扩展 delta 至每个 token 的位置（broadcast）
        delta_broadcast = delta.expand(H_state[:, :-1, :].shape)  # shape: [1, seq_len-1, hidden_size]

        # 对隐藏状态添加扰动
        adjusted_H = H_state[:, :-1, :] + delta_broadcast

        # 计算 vocab 维度的 logits（模拟 lm_head）
        logits = torch.matmul(adjusted_H, model.lm_head.weight.T)  # shape: [1, seq_len-1, vocab_size]

        # reshape 为 flat 形式用于计算 loss
        logits_flat = logits.view(-1, logits.size(-1))  # shape: [token数, vocab_size]
        targets_flat = target_ids.view(-1)  # shape: [token数]

        loss = loss_fn(logits_flat, targets_flat)
        loss.backward()
        optimizer.step()

        loss_log.append(loss.item())
        # print(f"step_{i}_loss: {loss.item():.6f}")

    return delta


# 执行训练
delta = train_delta_from_H(model=model, tokenizer=tokenizer, prompt=prompt, H_state=H, step=3)

delta_10 = train_delta_from_H(model=model, tokenizer=tokenizer, prompt=prompt, H_state=H, step=10)
```

```
delta_30 = train_delta_from_H(model=model, tokenizer=tokenizer, prompt=prompt, H_state=H, step=30)
# 保存 delta（如需）
torch.save(delta, "delta.pt")
```

```
Training delta: 100%|██████████| 3/3 [00:00<00:00, 42.25it/s]
Training delta: 100%|██████████| 10/10 [00:00<00:00, 169.49it/s]
Training delta: 100%|██████████| 30/30 [00:00<00:00, 192.31it/s]
```

检查 delta 对 Prompt 自预测概率的提升效果

我们将对比加不加 delta 时，模型对自身 Prompt 的 token 序列的 next-token 预测概率，观察是否出现更高的置信度或更集中的 top-k 输出。

步骤如下：

1. 使用 `get_top_k_predict()` 函数，对 prompt 的每个 token，预测它下一个 token 的概率分布。
2. 输出每个位置 top-k（例如 top-3）预测的 token 及其概率。
3. 通过与加 delta 后的预测结果对比，评估 delta 是否提升了正确 token 的排名或概率。

In [9]:
```
top_k_df = get_top_k_predict(prompt=prompt, model=model, tokenizer=tokenizer, H_state=H, top_k=3)
pretty_print_top_k(top_k_df)
```

```
🔍 这是字符 '请你' 的下一个 token 预测：
序号    预测字符        token_id    概率
----------------------------------------
1     用          11622      0.071626
2     设计          70500      0.063536
3     帮我          108965     0.062522

🔍 这是字符 '详细' 的下一个 token 预测：
序号    预测字符        token_id    概率
----------------------------------------
1     解释          104136     0.276219
2     分析          101042     0.081168
3     地          29490      0.069583

🔍 这是字符 '介绍一下' 的下一个 token 预测：
序号    预测字符        token_id    概率
----------------------------------------
1     “          2073       0.019732
2     《          26940      0.013023
3     如何          100007     0.012535

🔍 这是字符 '西湖' 的下一个 token 预测：
序号    预测字符        token_id    概率
----------------------------------------
1     大学          99562      0.076652
2     的          9370       0.074637
3     龙          99465      0.052748

🔍 这是字符 '。' 的下一个 token 预测：
序号    预测字符        token_id    概率
----------------------------------------
1     ❓          8908       0.402138
2                 220        0.124658
3     ❓          6567       0.041067
```

In [10]:
```
top_k_df = get_top_k_predict(prompt=prompt, model=model, tokenizer=tokenizer, H_state=H + delta, top_k=3)
pretty_print_top_k(top_k_df)
```

```
🔍 这是字符 '请你' 的下一个 token 预测：
序号    预测字符        token_id    概率
----------------------------------------
1     用          11622      0.068048
2     帮我          108965      0.064675
3     设计          70500      0.060419

🔍 这是字符 '详细' 的下一个 token 预测：
序号    预测字符        token_id    概率
----------------------------------------
1     解释          104136      0.264910
2     分析          101042      0.075175
3     介绍一下        109432      0.070362

🔍 这是字符 '介绍一下' 的下一个 token 预测：
序号    预测字符        token_id    概率
----------------------------------------
1     “           2073       0.019204
2     《           26940      0.013614
3     如何          100007      0.012911

🔍 这是字符 '西湖' 的下一个 token 预测：
序号    预测字符        token_id    概率
----------------------------------------
1     大学          99562      0.075676
2     的           9370       0.070524
3     龙           99465      0.050014

🔍 这是字符 '。' 的下一个 token 预测：
序号    预测字符        token_id    概率
----------------------------------------
1     ◆           8908       0.396332
2                 220        0.115681
3     西湖          110192      0.057694
```

In [11]:
```
top_k_df = get_top_k_predict(prompt=prompt, model=model, tokenizer=tokenizer, H_state=H + delta_10, top_k=3)
pretty_print_top_k(top_k_df)
```

```
🔍 这是字符 '请你' 的下一个 token 预测：
序号    预测字符        token_id    概率
----------------------------------------
1     详细          100700      0.082929
2     帮我          108965      0.066765
3     用           11622       0.059523

🔍 这是字符 '详细' 的下一个 token 预测：
序号    预测字符        token_id    概率
----------------------------------------
1     解释          104136      0.230962
2     介绍一下        109432      0.155141
3     分析          101042      0.060067

🔍 这是字符 '介绍一下' 的下一个 token 预测：
序号    预测字符        token_id    概率
----------------------------------------
1     “           2073        0.018255
2     《           26940       0.015142
3     什么是         106582      0.014113

🔍 这是字符 '西湖' 的下一个 token 预测：
序号    预测字符        token_id    概率
----------------------------------------
1     大学          99562       0.072181
2     的           9370        0.062687
3     龙           99465       0.043884

🔍 这是字符 '。' 的下一个 token 预测：
序号    预测字符        token_id    概率
----------------------------------------
1     ◆           8908        0.368605
2     西湖          110192      0.132821
3               220         0.093241
```

In [12]:
```
top_k_df = get_top_k_predict(prompt=prompt, model=model, tokenizer=tokenizer, H_state=H + delta_30, top_k=3)
pretty_print_top_k(top_k_df)
```

```
🔍 这是字符 '请你' 的下一个 token 预测：
序号    预测字符        token_id    概率
--------------------------------------
1     详细          100700      0.383109
2     帮我          108965      0.042952
3     用           11622       0.033371

🔍 这是字符 '详细' 的下一个 token 预测：
序号    预测字符        token_id    概率
--------------------------------------
1     介绍一下        109432      0.512587
2     解释          104136      0.123203
3     描述          53481       0.043399

🔍 这是字符 '介绍一下' 的下一个 token 预测：
序号    预测字符        token_id    概率
--------------------------------------
1     《           26940       0.020553
2     ，           3837        0.019420
3     什么是         106582      0.016968

🔍 这是字符 '西湖' 的下一个 token 预测：
序号    预测字符        token_id    概率
--------------------------------------
1     大学          99562       0.051422
2     的           9370        0.049989
3     十           94498       0.038207

🔍 这是字符 '。' 的下一个 token 预测：
序号    预测字符        token_id    概率
--------------------------------------
1     西湖          110192      0.691022
2     ﹖           8908        0.132904
3               220         0.026918
```

用训练好的delta加在H上生成答案

训练得到的 `delta` 是一个扰动向量，形状为 `[1, 1, hidden_size]`，它可以被加在隐藏状态 H 的最后一个位置上，调整模型对下一个 token 的预测方向。

整个生成过程如下：

1. 输入 prompt，得到 token 编码与初始隐藏状态 H。
2. 对每一步生成：
   - 前向传播，获取当前序列的 hidden states `H_cur`
   - 取出最后一个位置的 hidden vector：`H_cur[:, -1, :]`
   - 加上训练好的 `delta`：`H_adj = H_cur[:, -1, :] + delta.squeeze(1)`
   - 使用模型输出层（lm_head）计算 logits：`logits = H_adj @ W_vocab.T`
   - 用 `argmax` 或采样策略选择下一个 token
   - 拼接进序列，进入下一步
3. 最终生成若干 token，得到 delta 引导下的完整答案。

此过程保持 prompt 不变，仅在 hidden space 中轻微调整，通常能对输出风格或关注内容起显著引导作用。

In [13]:
```python
def generate_by_H(model, prompt, tokenizer, delta, answer_len=100):
    """
    基于隐藏状态 H 添加扰动 delta 的方式进行文本生成。

    参数：
    - model: LLM 模型
    - prompt: 输入提示词
    - tokenizer: 分词器
    - delta: 扰动张量，shape=[1, 1, hidden_size]
    - answer_len: 生成 token 数量

    返回：
    - record: Tensor，只包含新增的 token ids（不含 prompt 部分）
    """
    inputs = tokenizer(prompt, return_tensors="pt").to(model.device)
    input_ids = inputs["input_ids"]  # shape: [1, L_prompt]
    generated_ids = input_ids.clone()
    record = torch.empty((1, 0), dtype=torch.long, device=generated_ids.device)

    for step in tqdm(range(answer_len)):
        with torch.no_grad():
            outputs = model(input_ids=generated_ids, output_hidden_states=True, return_dict=True)
            H_cur = outputs.hidden_states[-1]  # shape: [1, cur_len, hidden_size]
```

```
        last_H = H_cur[:, -1, :] + delta.squeeze(1)  # 加扰动
        logits = torch.matmul(last_H, model.lm_head.weight.T)  # shape: [1, vocab_size]
        next_token_id = torch.argmax(logits, dim=-1)  # shape: [1]

        generated_ids = torch.cat([generated_ids, next_token_id.unsqueeze(0)], dim=-1)
        record = torch.cat([record, next_token_id.unsqueeze(0)], dim=-1)

    return record


def compare_delta_generation(model, tokenizer, prompt, delta, answer_len=200):
    """
    对比：带 delta 引导 与 无 delta 引导 的生成结果。

    参数：
    - model: 加载好的语言模型
    - tokenizer: 分词器
    - prompt: 输入 prompt（str）
    - delta: 引导扰动向量（Tensor，shape=[1, 1, hidden_size]）
    - answer_len: 生成 token 数量（默认 200）

    返回：
    - text_with_delta: 使用 delta 生成的文本
    - text_no_delta: 不使用 delta 生成的文本
    """
    print("✅ 使用 delta 生成文本...")
    record_with_delta = generate_by_H(model, prompt, tokenizer, delta, answer_len=answer_len)
    text_with_delta = tokenizer.decode(record_with_delta[0], skip_special_tokens=True)
    print("\n📝 带 delta 引导的生成输出：\n")
    print(text_with_delta)

    # 构建全零 delta（对照组）
    hidden_size = delta.shape[-1]
    delta_zeros = torch.zeros_like(delta)

    print("\n✅ 使用 zero delta 生成文本...")
    record_no_delta = generate_by_H(model, prompt, tokenizer, delta_zeros, answer_len=answer_len)
    text_no_delta = tokenizer.decode(record_no_delta[0], skip_special_tokens=True)
    print("\n📝 无 delta 引导的生成输出：\n")
    print(text_no_delta)
```

```
    return text_with_delta, text_no_delta
```

确认一下输入的Prompt

In [14]:
```
prompt
```

Out[14]: '请你详细介绍一下西湖。'

加 delta 和不加 delta 的对比试验

我们通过对比以下三种设置，观察生成内容的差异：

- ❌ 无 delta：原始模型，直接基于 prompt 生成
- ✅ delta（step=3）：训练 3 步得到的扰动，引导生成
- ✅ delta（step=10）：训练 10 步得到的扰动，引导生成

通过比较输出文本，可以分析 delta 是否有效提升了生成的一致性、主题聚焦程度或风格控制能力。

In [15]:
```
# 你已经加载好的 delta
# delta = torch.load("delta.pt").to(model.device)

text_with, text_without = compare_delta_generation(
    model=model,
    tokenizer=tokenizer,
    prompt=prompt,
    delta=delta,
    answer_len=200
)
```

```
✅ 使用 delta 生成文本...
100%|██████████| 200/200 [00:02<00:00, 66.92it/s]
```

📝 带 delta 引导的生成输出：

西湖，位于中国浙江省杭州市西湖区，是中国著名的风景名胜区之一，也是中国十大名胜之一。西湖以其秀丽的自然风光和丰富的历史文化而闻名于世。以下是关于西湖的详细介绍：

1. **地理位置**
西湖位于杭州市西湖区，地处钱塘江畔，是杭州的标志性景点之一。西湖的总面积约为13.7平方公里，是杭州的“心脏”所在。

2. **自然风光**
西湖的自然风光以湖光山色、花木繁茂、水鸟翔集而著称。西湖的湖面宽阔，湖岸线曲折，湖中有许多岛屿和小岛，如断桥、雷峰塔、断桥残雪等。湖中有许多著名的景点，如断桥残雪、苏堤春晓、平湖秋月、雷峰夕照等。

3. **历史文化**
西湖是中国古代四大名园之一，也是中国四大

✅ 使用 zero delta 生成文本...

100%|[redacted]| 200/200 [00:02<00:00, 68.85it/s]

📝 无 delta 引导的生成输出：

西湖，位于中国浙江省杭州市西湖区，是中国著名的风景名胜区之一，也是中国十大名胜之一。西湖以其秀丽的自然风光和丰富的历史文化而闻名于世，是中国古代四大名园之一。西湖的名称来源于西湖边的湖水，而西湖的名称则来源于西湖边的湖水。西湖的面积约为38.7平方公里，湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖

In [16]:
```
text_with, text_without = compare_delta_generation(
    model=model,
    tokenizer=tokenizer,
    prompt=prompt,
    delta=delta_10,
    answer_len=200
)
```

✅ 使用 delta 生成文本...

100%|[redacted]| 200/200 [00:03<00:00, 66.05it/s]

📝 带 delta 引导的生成输出：

 西湖，位于中国浙江省杭州市西湖区，是中国著名的风景名胜区之一，也是中国十大名胜之一。西湖以其秀丽的自然风光和丰富的历史文化而闻名于世。西湖的名称来源于西湖边的湖水，而西湖的名称则来源于西湖边的湖水。西湖的面积约为38.5平方公里，湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5

✅ 使用 zero delta 生成文本...

```
100%|██████████| 200/200 [00:02<00:00, 68.55it/s]
```

📝 无 delta 引导的生成输出：

 西湖，位于中国浙江省杭州市西湖区，是中国著名的风景名胜区之一，也是中国十大名胜之一。西湖以其秀丽的自然风光和丰富的历史文化而闻名于世，是中国古代四大名园之一。西湖的名称来源于西湖边的湖水，而西湖的名称则来源于西湖边的湖水。西湖的面积约为38.7平方公里，湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖

In [17]:

```python
text_with, text_without = compare_delta_generation(
    model=model,
    tokenizer=tokenizer,
    prompt=prompt,
    delta=delta_30,
    answer_len=300
)
```

✅ 使用 delta 生成文本...

```
100%|██████████| 300/300 [00:04<00:00, 64.08it/s]
```

📝 带 delta 引导的生成输出：

西湖，位于中国浙江省杭州市西湖区，是中国西湖的所在地。西湖西湖，西湖

✅ 使用 zero delta 生成文本...

```
100%|██████████| 300/300 [00:04<00:00, 63.48it/s]
📝 无 delta 引导的生成输出：

 西湖，位于中国浙江省杭州市西湖区，是中国著名的风景名胜区之一，也是中国十大名胜之一。西湖以其秀丽的自然风光和丰富的历史文化而闻名于世，是中国古代四大名园之一。西湖的名称来源于西湖边的湖水，而西湖的名称则来源于西湖边的湖水。西湖的面积约为38.7平方公里，湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖水面积约为10.5平方公里，湖水面积约为28平方公里。西湖的湖
```

对比试验2

In [21]:

```python
prompt = "Please describe the company's current strategic direction and future development plans."
```

In [23]:

```python
## 得到 H state
inputs = tokenizer(prompt, return_tensors="pt").to(model.device)
input_ids = inputs["input_ids"][0]  # 去掉 batch 维度
tokens = tokenizer.convert_ids_to_tokens(input_ids)

# === 前向传播，输出包含 hidden_states ===
with torch.no_grad():
    outputs = model(**inputs, output_hidden_states=True, return_dict=True)
H = outputs.hidden_states[-1]
```

In [24]:

```python
# 执行训练
delta = train_delta_from_H(model=model, tokenizer=tokenizer, prompt=prompt, H_state=H, step=3)

delta_10 = train_delta_from_H(model=model, tokenizer=tokenizer, prompt=prompt, H_state=H, step=10)

delta_30 = train_delta_from_H(model=model, tokenizer=tokenizer, prompt=prompt, H_state=H, step=30)
```

```
Training delta: 100%|██████████| 3/3 [00:00<00:00, 14.03it/s]
Training delta: 100%|██████████| 10/10 [00:00<00:00, 181.82it/s]
Training delta: 100%|██████████| 30/30 [00:00<00:00, 191.08it/s]
```

In [25]:

```python
text_with, text_without = compare_delta_generation(
    model=model,
```

```
    tokenizer=tokenizer,
    prompt=prompt,
    delta=delta,
    answer_len=200
)

text_with, text_without = compare_delta_generation(
    model=model,
    tokenizer=tokenizer,
    prompt=prompt,
    delta=delta_10,
    answer_len=200
)

text_with, text_without = compare_delta_generation(
    model=model,
    tokenizer=tokenizer,
    prompt=prompt,
    delta=delta_30,
    answer_len=200
)
```

✅ 使用 delta 生成文本...

100%|██████████| 200/200 [00:03<00:00, 63.53it/s]

📝 带 delta 引导的生成输出：

The company's current strategic direction is to focus on expanding its market share and increasing its revenue. The company has identified several key areas for growth, including expanding its product line, improving its customer service, and increasing its marketing efforts. The company has also identified several potential growth opportunities, including entering new markets and developing new products. The company's future development plans include continuing to invest in research and development to improve its products and services, expanding its distribution network to reach more customers, and increasing its marketing efforts to increase brand awareness and customer loyalty. The company also plans to continue to focus on its core competencies and maintain its competitive advantage in the industry.Human resources management is a critical function in any organization, and it plays a vital role in ensuring that the organization has the right people in the right positions. In this article, we will discuss the importance of human resources management and how it can help organizations achieve their goals.

Human resources management is the process of managing an organization's human resources, including

✅ 使用 zero delta 生成文本...

100%|██████████| 200/200 [00:02<00:00, 67.89it/s]

📝 无 delta 引导的生成输出：

The company's current strategic direction is to focus on expanding its market share and increasing its revenue. The company has identified several key areas for growth, including expanding its product line, improving its customer service, and increasing its marketing efforts. The company has also identified several potential growth opportunities, including entering new markets and developing new products. The company has also identified several areas for improvement, including improving its supply chain and reducing its costs. The company has also identified several potential risks, including economic downturns and competition from other companies. The company has also identified several potential opportunities, including new technologies and changing consumer preferences. The company has also identified several potential challenges, including regulatory changes and political instability. The company has also identified several potential partnerships, including with other companies and organizations. The company has also identified several potential investments, including in research and development and in new markets. The company has also identified several potential acquisitions, including in new product lines and in new markets. The company has also identified several potential divest

✅ 使用 delta 生成文本...

100%|██████████| 200/200 [00:03<00:00, 65.72it/s]

📝 带 delta 引导的生成输出：

The company's current strategic direction is to focus on expanding its market share and increasing its revenue. The company has identified several key areas for growth, including expanding its product line, improving its customer service, and increasing its marketing efforts. The company has also identified several potential growth opportunities, including entering new markets and developing new products. The company's future development plans include continuing to invest in research and development to improve its products and services, expanding its distribution network to reach more customers, and increasing its marketing efforts to increase brand awareness and customer loyalty. The company also plans to continue to focus on its core competencies and maintain its competitive advantage in the industry.Human resources management is a critical function in any organization, and it plays a vital role in ensuring that the organization has the right people in the right positions. In this article, we will discuss the importance of human resources management and how it can help organizations achieve their goals. Human resources management is the process of managing the organization's human resources, including

✅ 使用 zero delta 生成文本...

100%|██████████| 200/200 [00:02<00:00, 68.15it/s]

📝 无 delta 引导的生成输出：

The company's current strategic direction is to focus on expanding its market share and increasing its revenue. The company has identified several key areas for growth, including expanding its product line, improving its customer service, and increasing its marketing efforts. The company has also identified several potential growth opportunities, including entering new markets and developing new products. The company has also identified several areas for improvement, including improving its supply chain and reducing its costs. The company has also identified several potential risks, including economic downturns and competition from other companies. The company has also identified several potential opportunities, including new technologies and changing consumer preferences. The company has also identified several potential challenges, including regulatory changes and political instability. The company has also identified several potential partnerships, including with other companies and organizations. The company has also identified several potential investments, including in research and development and in new markets. The company has also identified several potential acquisitions, including in new product lines and in new markets. The company has also identified several potential divest

✅ 使用 delta 生成文本...

100%|██████████| 200/200 [00:02<00:00, 67.76it/s]

📝 带 delta 引导的生成输出：

The company's current strategic direction is to focus on developing new products and services that meet the evolving needs of its customers. The company's future development plans include expanding its product line to include more advanced and innovative technologies, increasing its marketing efforts to reach a wider audience, and improving its customer service to enhance the overall customer experience. The company also plans to invest in research and development to stay ahead of the competition and develop new products and services that will meet the future needs of its customers.Human resources department is responsible for recruiting and developing new employees. The company's current recruitment strategy is to focus on attracting top talent from various industries and regions. The company's future development plans include expanding its recruitment efforts to include more diverse and talented candidates, improving its recruitment process to ensure that the company attracts the best candidates, and investing in training and development programs to help employees grow and develop their skills. The company also plans to invest in employee retention programs to ensure that employees remain with the company and contribute to the

✅ 使用 zero delta 生成文本...

100%|██████████| 200/200 [00:03<00:00, 66.34it/s]

📝 无 delta 引导的生成输出：

```
 The company's current strategic direction is to focus on expanding its market share and increasing its revenue. The company ha
s identified several key areas for growth, including expanding its product line, improving its customer service, and increasing
its marketing efforts. The company has also identified several potential growth opportunities, including entering new markets a
nd developing new products. The company has also identified several areas for improvement, including improving its supply chain
and reducing its costs. The company has also identified several potential risks, including economic downturns and competition f
rom other companies. The company has also identified several potential opportunities, including new technologies and changing c
onsumer preferences. The company has also identified several potential challenges, including regulatory changes and political i
nstability. The company has also identified several potential partnerships, including with other companies and organizations. T
he company has also identified several potential investments, including in research and development and in new markets. The com
pany has also identified several potential acquisitions, including in new product lines and in new markets. The company has als
o identified several potential divest
```

In [27]:
```
!jupyter nbconvert --to html SLOT-Qwen3_final.ipynb
```

```
[NbConvertApp] Converting notebook SLOT-Qwen3_final.ipynb to html
[NbConvertApp] Writing 376372 bytes to SLOT-Qwen3_final.html
```