```
import os

from swarms import Agent
from swarm_models import OpenAIChat
from swarms.prompts.finance_agent_sys_prompt import (
    FINANCIAL_AGENT_SYS_PROMPT,
)
from examples.structs.swarms.experimental.a_star_swarm import (
    AStarSwarm,
)

# Set up the model as provided
api_key = os.getenv("OPENAI_API_KEY")
model = OpenAIChat(
    api_key=api_key, model_name="gpt-4o-mini", temperature=0.1
)


# Heuristic example (can be customized)
def example_heuristic(agent: Agent) -> float:
    """
    Example heuristic that prioritizes agents based on some custom logic.

    Args:
        agent (Agent): The agent to evaluate.
    Returns:
        float: The priority score for the agent.
    """
    # Example heuristic: prioritize based on the length of the agent's name (as a proxy for complexity)
    return len(agent.agent_name)



# Initialize root agent
root_agent = Agent(
    agent_name="Financial-Analysis-Agent",
    system_prompt=FINANCIAL_AGENT_SYS_PROMPT,
    llm=model,
    max_loops=2,
    autosave=True,
    dashboard=False,
    verbose=True,
    streaming_on=True,
    dynamic_temperature_enabled=True,
    saved_state_path="finance_agent.json",
    user_name="swarms_corp",
    retry_attempts=3,
    context_length=200000,
)

# List of child agents
child_agents = [
    Agent(
        agent_name="Child-Agent-1",
        system_prompt=FINANCIAL_AGENT_SYS_PROMPT,
        llm=model,
        max_loops=2,
        autosave=True,
        dashboard=False,
        verbose=True,
        streaming_on=True,
        dynamic_temperature_enabled=True,
        saved_state_path="finance_agent_child_1.json",
        user_name="swarms_corp",
        retry_attempts=3,
        context_length=200000,
    ),
    Agent(
        agent_name="Child-Agent-2",
        system_prompt=FINANCIAL_AGENT_SYS_PROMPT,
        llm=model,
        max_loops=2,
        autosave=True,
        dashboard=False,
        verbose=True,
        streaming_on=True,
        dynamic_temperature_enabled=True,
        saved_state_path="finance_agent_child_2.json",
            user_name="swarms_corp",
            retry_attempts=3,
            context_length=200000,
        ),
    ]

    # Create the A* swarm
    swarm = AStarSwarm(
        root_agent=root_agent,
        child_agents=child_agents,
        heauristic=example_heuristic,
    )

    # Run the task with the heuristic
    result = swarm.run(
        "What are the components of a startups stock incentive equity plan",
    )
    print(result)

    # Visualize the communication flow
    swarm.visualize()
```