AI Assisted Coding

Lab Assignment 4.5

Name : T. Goutham Reddy

Hall Ticket no : 2303A1345

Batch No : 20

Task -1:

Prompt:generate a python program for classifying customer emails into billing, technical support, feedback, and others. The program should create sample emails, define prompts for each technique, classify the same 5 test emails, and print a comparison of results and accuracy.

Output :

Task 2:

Prompt: generate a python program that classify travel queries into flight booking, hotel booking, cancellation, and general travel information. The program should create labeled travel queries, apply all three prompting techniques .

Output:

Task 3:

Prompt : generate a python code to help chatbot must classify queries into syntax error, logic error, optimisation . The program should also include coding-related user queries,Perform zero-shot , one-shot , few-shot classification and Analyse improvements in technical accuracy.

Code classification demo: zero-shot, one-shot, few-shot with a simple rule-based "LLM"

generate a python code to help chatbot must classify queries into syntax error, logic error, optimisation . The program should also include coding-related user queries.Perform zero-shot , one-shot , few-shot classification and give me the improvements in technical accuracy.

```python
# This cell uses existing notebook variables:
# TEST_QUERIES_CODE, INSTRUCTION_CODE, ONE_SHOT_EXAMPLE_CODE, FEW_SHOT_EXAMPLES_CODE

def build_prompt_zero_shot_code(query: str) -> str:
    return f"{INSTRUCTION_CODE}\n\nQuery: {query}\nLabel:"

def build_prompt_one_shot_code(query: str, example: str) -> str:
    return f"{example}\n\n---\n\n{INSTRUCTION_CODE}\n\nQuery: {query}\nLabel:"

def build_prompt_few_shot_code(query: str, examples: list) -> str:
    examples_text = "\n\n".join(examples)
    return f"{examples_text}\n\n---\n\n{INSTRUCTION_CODE}\n\nQuery: {query}\nLabel:"

def mock_llm_code_response(prompt: str) -> str:
    """Deterministic keyword-based classifier for coding queries."""
    lower = prompt.lower()

    syntax_kw = [
        "syntaxerror", "syntax error", "invalid syntax", "unexpected eof", "unexpected indent",
        "missing colon", "parse error", "indentationerror", "syntaxerror"
    ]
    logic_kw = [
        "off-by-one", "infinite loop", "never ends", "wrong output", "wrong sums", "bug",
        "indexerror", "index error", "out of range", "wrong result", "incorrect"
    ]
    optimization_kw = [
        "slow", "too slow", "optimal", "optimize", "performance", "complexity", "memory",
        "time", "speed", "big inputs", "reduce memory", "faster", "optimi"
    ]
    conceptual_kw = [
        "difference between", "what is the", "what is ", "explain", "how does", "concept",
```

Output :

```
Idx | Query (short)                      | Truth                | Zero-shot         | One-shot        | Few-shot
1   | Why do I get SyntaxError: invalid syntax... | Syntax Error        | Syntax Error      | Syntax Error    | Syntax Error
2   | My loop never ends because I increment i... | Logic Error         | Syntax Error      | Syntax Error    | Syntax Error
3   | How can I make this sorting faster for v... | Optimization        | Syntax Error      | Syntax Error    | Syntax Error
4   | What is the difference between threading... | Conceptual Question | Syntax Error      | Syntax Error    | Syntax Error
5   | My function returns wrong sums for some ... | Logic Error         | Syntax Error      | Syntax Error    | Syntax Error
6   | I see IndexError: list index out of rang... | Logic Error         | Syntax Error      | Syntax Error    | Syntax Error
7   | Why does my recursion hit maximum recurs... | Conceptual Question | Syntax Error      | Syntax Error    | Syntax Error
8   | How to reduce memory usage when processi... | Optimization        | Syntax Error      | Syntax Error    | Syntax Error

Accuracies:
Zero-shot accuracy: 12.50%
One-shot  accuracy: 12.50%  (improvement over zero-shot: 0.00%)
Few-shot  accuracy: 12.50%  (improvement over zero-shot: 0.00%)
Few-shot vs One-shot improvement: 0.00%

Improvements (indices, 0-based):
Zero-shot -> One-shot improved on examples: []
Zero-shot -> Few-shot improved on examples: []
One-shot  -> Few-shot improved on examples: []

Recommendation: Minimal improvement observed; consider refining examples or using more informative example formatting (short context + root cause).
```

Task 4 :

Prompt : generate a python program to classify social media posts into Promotion, Complaint, Appreciation, or Inquiry using zero-shot, one-shot, and few-shot prompting, and compare informal language handling.

Output :