AI ASSISTED CODING

ASSIGNMENT-4.5

M.AKHIL REDDY

2303A52315

BATCH-45

Objective: To explore and compare Zero-shot, One-shot, and Few-shot prompting techniques for classifying emails into predefined categories using a large language model (LLM).

1. Suppose that you work for a company that receives hundreds of customer emails daily. Management wants to automatically classify emails into categories like "Billing", "Technical Support", "Feedback", and "Others" before assigning them to appropriate departments. Instead of training a new model, your task is to use prompt engineering techniques with an existing LLM to handle the classification.

Tasks to be completed are as below

a. Prepare Sample Data:

• Create or collect 10 short email samples, each belonging to one of the 4 categories.

b. Zero-shot Prompting:

• Design a prompt that asks the LLM to classify a single email without providing any examples.

• Example prompt:

“Classify the following email into one of the following categories: Billing, Technical Support, Feedback, Others. Email: ‘I have not received my invoice for last month.’”

c. One-shot Prompting:

• Add one labeled example before asking the model to classify a new email.

d. Few-shot Prompting:

• Use 3–5 labeled examples in your prompt before asking the

model to classify a new email.

e. Evaluation:

• Run all three techniques on the same set of 5 test emails.

• Compare and document the accuracy and clarity of responses.
PROMPTS:

Classify the following email into one of the following categories: Billing, Technical Support, Feedback, Others. Provide only the category name as your response.

Email: [EMAIL_TEXT]

Classify the following email into one of the following categories: Billing, Technical Support, Feedback, Others. Provide only the category name as your response.

Example:

Email: "I was charged twice for my subscription this month. Please refund the extra amount."

Category: Billing

Email: [EMAIL_TEXT]

Classify the following email into one of the following categories: Billing, Technical Support, Feedback, Others. Provide only the category name as your response.

Example:

Email: "I was charged twice for my subscription this month. Please refund the extra amount."

Category: Billing

Example:

Email: "My app keeps crashing when I try to login. Can you help fix this?"

Category: Technical Support

Example:

Email: "I love the new interface update! It's much easier to navigate."

Category: Feedback

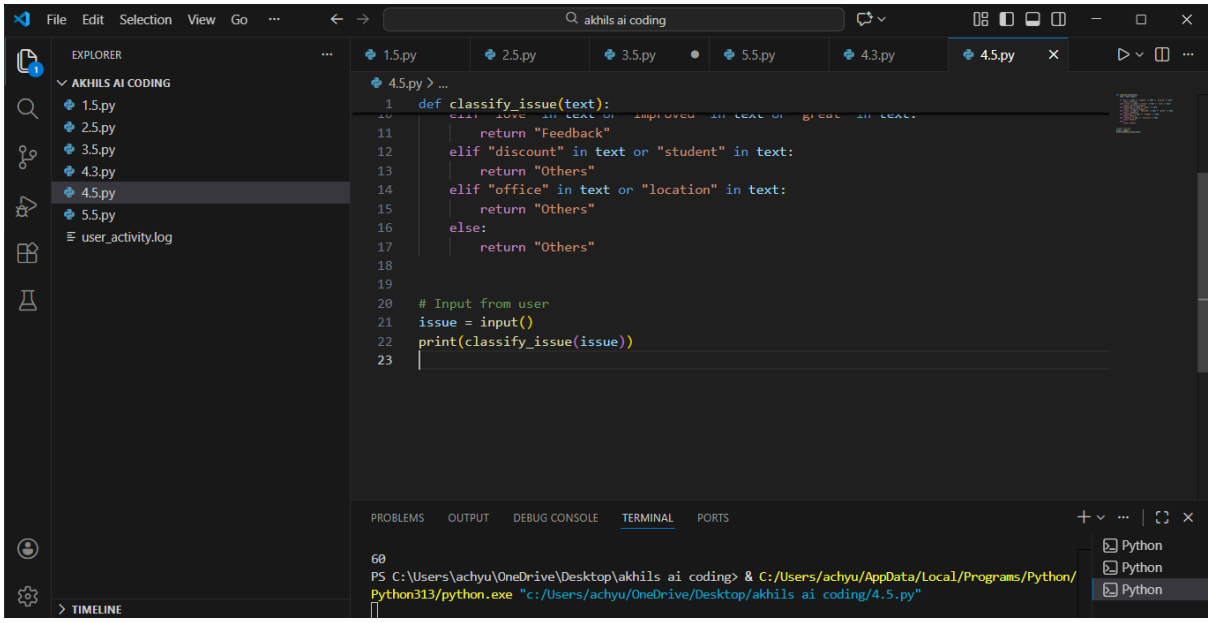
Example:

Email: "What are your office hours? I'd like to visit for a meeting."

Category: Others

Email: [EMAIL_TEXT]

EXPECTED OUTPUT:



2. Travel Query

Classification Scenario:

A travel assistant must classify queries into Flight Booking, Hotel Booking, Cancellation, or General Travel Info.

Tasks:

a. Prepare labeled travel queries.

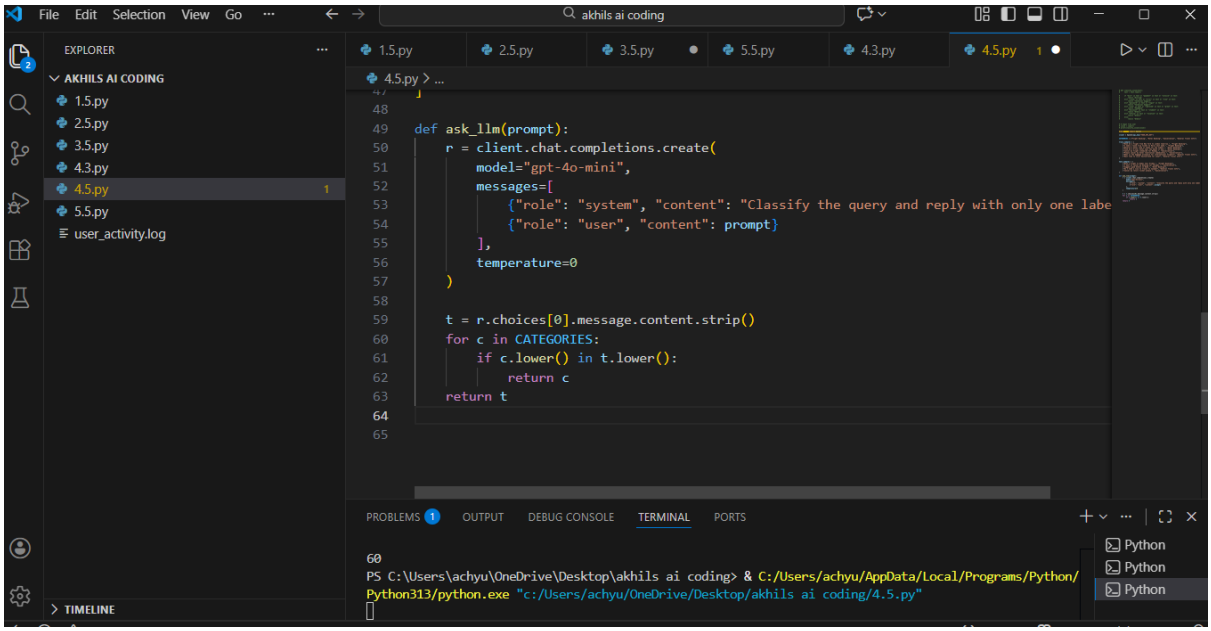
b. Apply Zero-shot prompting.

c. Apply One-shot prompting.

d. Apply Few-shot prompting.

e. Compare response consistency.

Expected output:



3.Programming Question Type Identification

Scenario:

A coding help chatbot must classify queries into Syntax Error, Logic Error, Optimization, or Conceptual Question.

Tasks:

a. Prepare coding-related user queries.

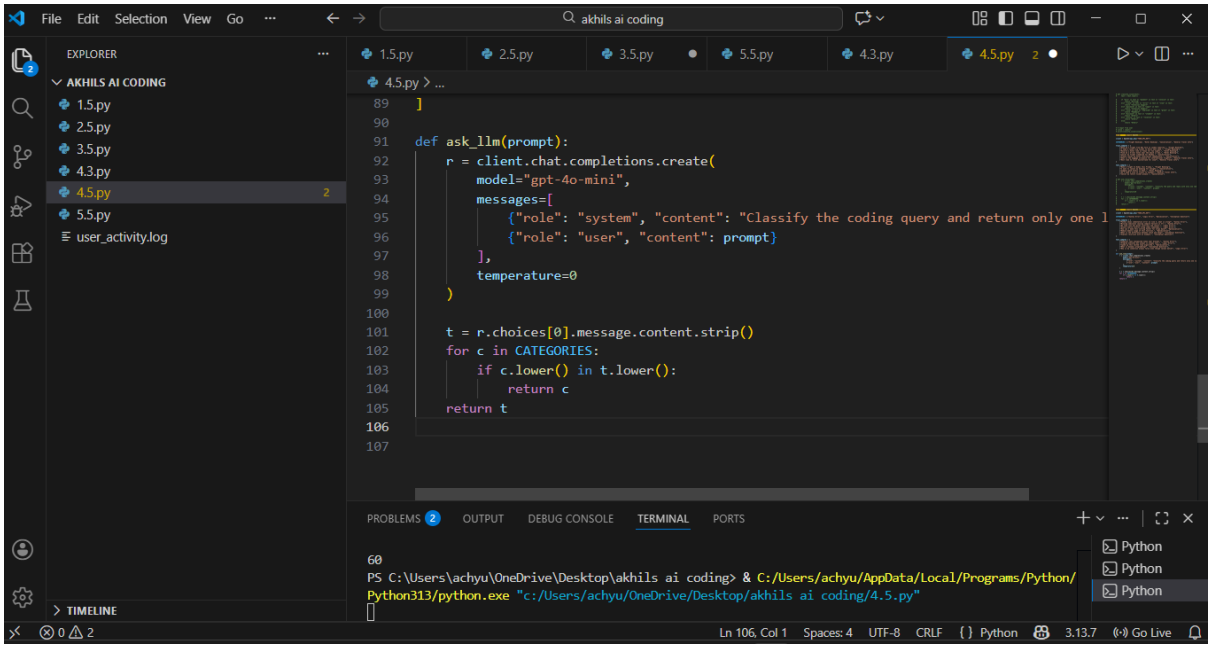
b. Perform Zero-shot classification.

c. Perform One-shot classification.

d. Perform Few-shot classification.

e. Analyze improvements in technical accuracy.

EXPECTED OUTPUT:



4.Social Media Post Categorization

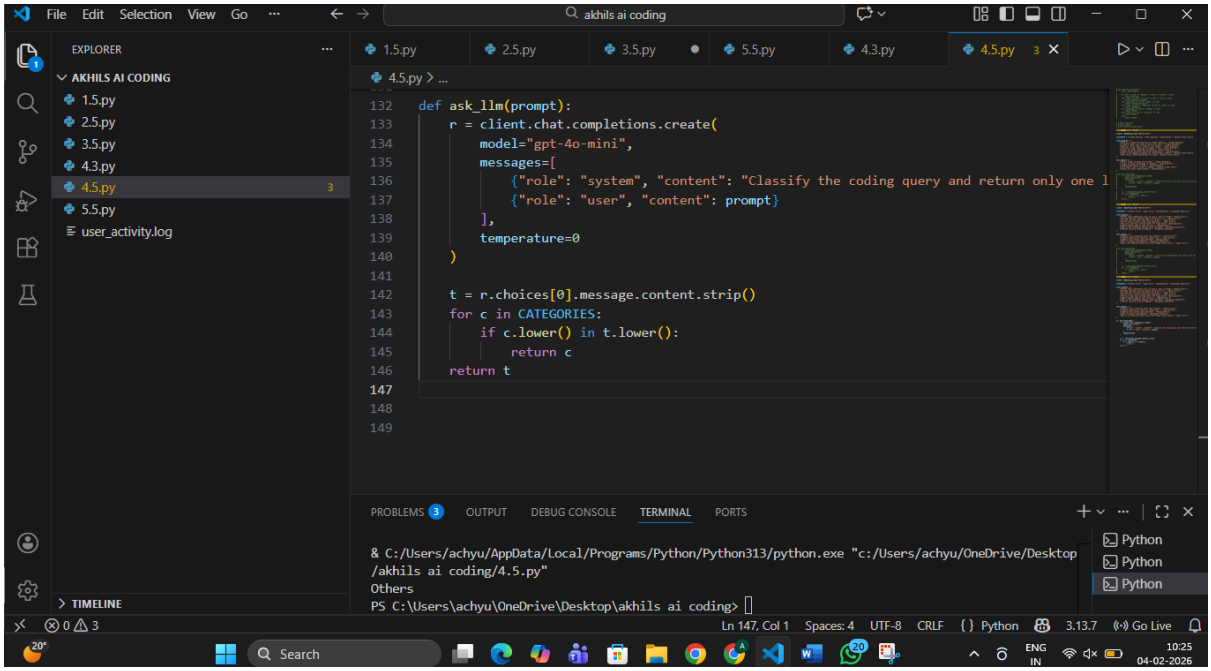
Scenario:

A social media analytics tool must classify posts into Promotion, Complaint, Appreciation, or Inquiry.

Tasks:

1. Prepare sample social media posts.
2. Use Zero-shot prompting.
3. Use One-shot prompting.
4. Use Few-shot prompting.
5. Analyze informal language handling.

EXPECTED OUTPUT:


File Edit Selection View Go
akhils ai coding
EXPLORER
AKHILS AI CODING
1.5.py
2.5.py
3.5.py
4.3.py
4.5.py
5.5.py
user_activity.log
TIMELINE
1.5.py
2.5.py
3.5.py
5.5.py
4.3.py
4.5.py
4.5.py > ...
def ask_llm(prompt):
r = client.chat.completions.create(
model="gpt-4o-mini",
messages=[
{"role": "system", "content": "Classify the coding query and return only one l
{"role": "user", "content": prompt}
],
temperature=0
)
t = r.choices[0].message.content.strip()
for c in CATEGORIES:
if c.lower() in t.lower():
return c
return t
PROBLEMS 3
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
& C:/Users/achyu/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/achyu/OneDrive/Desktop
/akhils ai coding/4.5.py"
Others
PS C:\Users\achyu\OneDrive\Desktop\akhils ai coding>
Python
Python
Python
Ln 147, Col 1
Spaces: 4
UTF-8
CRLF
Python
3.13.7
Go Live
Search
ENG
IN
10:25
04-02-2026