

SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE		DEPARTMENT OF COMPUTER SCIENCE ENGINEERING	
Program Name: B. Tech		Assignment Type: Lab	AcademicYear:2025-2026
Course Coordinator Name		Venkataramana Veeramsetty	
Instructor(s)Name		1. Dr. Mohammed Ali Shaik 2. Dr. T Sampath Kumar 3. Mr. S Naresh Kumar 4. Dr. V. Rajesh 5. Dr. Brij Kishore 6. Dr Pramoda Patro 7. Dr. Venkataramana 8. Dr. Ravi Chander 9. Dr. Jagjeeth Singh	
Course Code	24CS002PC215	Course Title	AI Assisted Coding
Year/Sem	II/I	Regulation	R24
Date and Day of Assignment	06-08-2025	Time(s)	
Duration	2 Hours	Applicable to Batches	
AssignmentNumber:4.5(Present assignment number)/24(Total number of assignments)			
Q. No.	Question	ExpectedTime to complete	
1	<p>Lab 4: Advanced Prompt Engineering: Zero-shot, one-shot, and few-shot techniques</p> <p>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).</p> <p>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.</p> <p>Tasks to be completed are as below</p> <p>1. Prepare Sample Data:</p> <ul style="list-style-type: none"> Create or collect 10 short email samples, each belonging to one of the 4 categories. <p>2. Zero-shot Prompting:</p> <ul style="list-style-type: none"> Design a prompt that asks the LLM to classify a single email without providing any examples. Example prompt: <i>"Classify the following email into one of the following categories: Billing, Technical Support, Feedback, Others. Email: 'I have not received my invoice"</i> 	08.08.2025 EOD	

for last month."

Prompt:

Write a py code for collection of 10 short email sample

Code:

```
1 emails = [  
2     "I have not received my invoice for last month.", # Billing  
3     "My internet is not working properly.", # Technical Support  
4     "Great service, I am very happy with your support.", # Feedback  
5     "Can you share your office address?", # Others  
6     "The payment I made yesterday is not reflecting.", # Billing  
7     "My laptop crashes after installing your software.", # Technical Support  
8     "The delivery of my order was quick and smooth.", # Feedback  
9     "Please update me on the job openings.", # Others  
10    "I was charged twice for the same subscription.", # Billing  
11    "Your app crashes whenever I try to upload a photo." # Technical Support  
12 ]  
13  
14 # Zero-shot prompt for each email  
15 for i, email in enumerate(emails, 1):  
16     print(f"\nEmail (i): {email}")  
17     print(f"Prompt: Classify this email into Billing, Technical Support, Feedback, or Others.\nEmail: \"{email}\"")
```

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  
  
Email 6: My laptop crashes after installing your software.  
Prompt: Classify this email into Billing, Technical Support, Feedback, or Others.  
Email: "My laptop crashes after installing your software."  
  
Email 7: The delivery of my order was quick and smooth.  
Prompt: Classify this email into Billing, Technical Support, Feedback, or Others.  
Email: "The delivery of my order was quick and smooth."  
  
Email 8: Please update me on the job openings.  
Prompt: Classify this email into Billing, Technical Support, Feedback, or Others.  
Email: "The payment I made yesterday is not reflecting."  
  
Email 6: My laptop crashes after installing your software.  
Prompt: Classify this email into Billing, Technical Support, Feedback, or Others.  
Email: "My laptop crashes after installing your software."  
  
Email 7: The delivery of my order was quick and smooth.  
Prompt: Classify this email into Billing, Technical Support, Feedback, or Others.  
Email: "The delivery of my order was quick and smooth."  
  
Email 8: Please update me on the job openings.  
Email: "The payment I made yesterday is not reflecting."  
  
Email 6: My laptop crashes after installing your software.  
Prompt: Classify this email into Billing, Technical Support, Feedback, or Others.  
Email: "My laptop crashes after installing your software."  
  
Email 7: The delivery of my order was quick and smooth.  
Prompt: Classify this email into Billing, Technical Support, Feedback, or Others.  
Email: "The delivery of my order was quick and smooth."  
  
Email 8: Please update me on the job openings.  
Prompt: Classify this email into Billing, Technical Support, Feedback, or Others.  
Email: "My laptop crashes after installing your software."  
  
Email 7: The delivery of my order was quick and smooth.  
Prompt: Classify this email into Billing, Technical Support, Feedback, or Others.  
Email: "The delivery of my order was quick and smooth."  
  
Email 8: Please update me on the job openings.  
Email: "The payment I made yesterday is not reflecting."  
  
Email 7: The delivery of my order was quick and smooth.  
Prompt: Classify this email into Billing, Technical Support, Feedback, or Others.  
Email: "The delivery of my order was quick and smooth."
```

3. One-shot Prompting:

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

Prompt:

Write a py code in one short one example for email samples

Code:

```
# Labeled example
example_email = "I was charged twice for my bill."
example_label = "Billing"

# New email to classify
new_email = "My internet is not working since morning."

# One-shot prompt
prompt = f"""
Example:
Email: "{example_email}"
Category: {example_label}

Now classify this email into Billing, Technical Support, Feedback, or Others:
Email: "{new_email}"
"""

print(prompt)
```

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\PROGRAMMES VSCODE\AI coding> & C:\Users\venkatesh\AppData\Local\Programs\Python\Python313\python.exe

Example:
Email: "I was charged twice for my bill."
Category: Billing

Now classify this email into Billing, Technical Support, Feedback, or Others:
Email: "My internet is not working since morning."

PS C:\PROGRAMMES VSCODE\AI coding> |
```

4. Few-shot Prompting:

- Use 3–5 labeled examples in your prompt before asking the model to classify a new email.
- Prompt: create py code of email sample with few short example.

Code:

```
# Labeled examples
examples = [
    ("I was charged twice for my bill.", "Billing"),
    ("My internet is not working since morning.", "Technical Support"),
    ("Great service, I am very happy with your support.", "Feedback"),
]

# New email to classify
new_email = "Can you send me your company profile?"

# Build few-shot prompt
prompt = "Examples:\n"
for text, label in examples:
    prompt += f'Email: "{text}"\nCategory: {label}\n\n'

prompt += f'Now classify this email into Billing, Technical Support, Feedback, or Others:\nEmail: "{new_email}"\n'

print(prompt)
```

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\PROGRAMMES VSCODE\AI coding> & C:\Users\venkatesh\AppData\Local\Programs\Python\Python313\python.exe "c:/PROG
Examples:
Email: "I was charged twice for my bill."
Category: Billing

Email: "My internet is not working since morning."
Category: Technical Support

Email: "Great service, I am very happy with your support."
Category: Feedback

Now classify this email into Billing, Technical Support, Feedback, or Others:
Email: "Can you send me your company profile?"
PS C:\PROGRAMMES VSCODE\AI coding> |
```

	<p>Comment:</p> <p>Few-shot Prompting: Shows 3 labeled email examples to guide the model, then asks it to classify a new email into Billing, Technical Support, Feedback, or Others.</p> <p>5. Evaluation:</p> <ul style="list-style-type: none"> • Run all three techniques on the same set of 5 test emails. • Compare and document the accuracy and clarity of responses. <p>Requirements:</p> <ul style="list-style-type: none"> • VS Code with Github Copilot or Cursor IDE and/or Google Colab with Gemini <p>Deliverables:</p> <ul style="list-style-type: none"> • A .txt or .md file showing prompts and model responses. • A comparison table showing classification accuracy for each technique. • A short reflection on which method was most effective and why 	
--	--	--