

ASSIGNMENT-4.1

HT No:2303A51585

Batch No:28

Course:AI Assisted Coding

Problem Statement 1: Customer Email Classification

Problem Description :

A company receives a large number of customer emails every day

and wants to automatically classify them into the following

categories:

- Billing
- Technical Support
- Feedback
- Others

Instead of training a new machine learning model, the company

decides to use prompt engineering techniques with an existing large

language model.

Tasks:

1. Prepare Sample Data:

No	Email Content	Category
E1	I was charged extra on my last bill	Billing
E2	Unable to log into my account	Technical Support

E3	The service quality is excellent	Feedback
E4	How can I update my profile details?	Others
E5	My internet connection is not working	Technical Support

2. Zero-shot Prompting:

Prompt Used: Classify the following customer email into one of these categories:
Billing, Technical Support, Feedback, Others.

Email: "I was charged twice for my payment."

Output:

Billing

Observation:

The model correctly classified the email without any examples.

3. One-shot Prompting:

Prompt Used:

Example:

Email: "My bill amount is incorrect."

Category: Billing

Now classify the following email:

Email: "The app crashes whenever I open it."

Output:

Technical Support

Observation:

Providing one example improved clarity and consistency.

4. Few-shot Prompting

Prompt Used:

Example 1:

Email: "Extra charges appear on my bill."

Category: Billing

Example 2:

Email: "Cannot reset my password."

Category: Technical Support

Example 3:

Email: "Great customer service."

Category: Feedback

Now classify the following email:

Email: "How do I change my registered phone number?"

Output:

Others

Observation:

Few-shot prompting produced the most accurate classification.

5.Evaluation:

Few-shot prompting performed best due to richer context and examples.

Problem Statement 2-Intent Classification for Chatbot Queries

Problem Description:

A company wants to deploy a chatbot to handle customer queries. Each query must be classified into one of the following intents: Account Issue, Order Status, Product Inquiry, or General Question using prompt engineering technique

Tasks:

1.Prepare Sample Data:

No	Query	Intent
Q1	I forgot my password	Account Issue
Q2	Where is my order?	Order Status
Q3	Does this phone support 5G?	Product Inquiry
Q4	What are your working hours?	General Question
Q5	My account is locked	Account Issue
Q6	Is this product available in blue?	Product Inquiry

2. Zero-shot Prompting:

Prompt Used:

Classify the following query into

Account Issue, Order Status, Product Inquiry, General Question.

Query: "When will my order arrive?"

Output:

Order Status

Observation:

Correct classification without examples.

3. One-shot Prompting:

Prompt Used:

Example:

Query: "I cannot access my account."

Intent: Account Issue

Now classify: Query: "Does this laptop have a warranty?"

Output:

Product Inquiry

Observation:

Improved understanding of emotional tone through an example.

4. Few-shot Prompting:

Prompt Used:

Example 1 :

Query: "My order is delayed."

Intent: Order Status

Example 2:

Query: "Is this product available?"

Intent: Product Inquiry

Example 3:

Query: "I forgot my password."

Intent: Account Issue

Now classify:

Query: "What time does customer support open?"

Output:

General Question

Observation:

The most accurate and consistent results by clearly differentiating between all intent categories.

5. Evaluation:

Few-shot prompting reduced ambiguity and improved intent detection.

Problem Statement 3: Student Feedback Analysis

Problem description: A university collects student feedback and wants to categorize comments as Positive, Negative, or Neutral.

Tasks:

1. Prepare Sample Data

No	Student Feedback	Sentiment
F1	The lectures were very interesting and engaging	Positive
F2	The syllabus coverage was average	Neutral
F3	Too much workload throughout the semester	Negative
F4	Faculty support was very helpful	Positive
F5	Exam pattern was confusing	Negative

2.Zero-shot Prompt :

Prompt Used:

Classify the sentiment of the following feedback as:

Positive, Negative, or Neutral.

Feedback: "The course content is very helpful."

Output:

Positive

Observation:

Identified sentiment correctly for straightforward feedback.

3.One-shot Prompt:

Prompt Used:

Example:

Feedback: "The lectures were boring."

Sentiment: Negative

Now classify:

Feedback: "The instructor explained topics clearly."

Output:

Positive

Observation:

Improved understanding of emotional tone through an example.

4.Few-shot Prompt:

Prompt Used:

Example 1:

Feedback: "I enjoyed the course."

Sentiment: Positive

Example 2:

Feedback: "The syllabus is average."

Sentiment: Neutral

Example 3:

Feedback: "Too much workload."

Sentiment: Negative

Now classify:

Feedback: "Assignments were manageable."

Output:

Neutral

Observation:

Provided the most reliable sentiment classification, especially for neutral and mixed feedback.

5.Explaination:

Examples help the model understand emotional tone and improve sentiment accuracy.

Problem Statement 4: Course Recommendation System

Problem Description: An online learning platform wants to recommend courses by

classifying learner queries into Beginner, Intermediate, or Advanced levels.

Tasks:

1. Prepare Sample Data

No	Learner Query	Level
C1	I want to learn programming basics	Beginner
C2	Explain object-oriented programming concepts	Intermediate
C3	Advanced deep learning techniques	Advanced
C4	What is Python language?	Beginner
C5	Build a full-stack web application	Intermediate

2.Zero-shot Prompt:

Prompt Used:

Classify the learner query into: Beginner, Intermediate, or Advanced.

Query: "I want to learn Python basics."

Output:

Beginner

Observation:

It was worked for basic level identification but lacked precision for overlapping skill levels.

3.One-shot Prompt:

Prompt Used:

Example:

Query: "Explain variables in Python."

Level: Beginner

Now classify:

Query: "Teach me object-oriented programming in Java."

Output:

Intermediate

Observation:

It helped the model understand the expected difficulty range.

4.Few-shot Prompt:

Prompt Used:

Example 1:

Query: "What is HTML?"

Level: Beginner

Example 2:

Query: "Build a REST API using Flask."

Level: Intermediate

Example 3:

Query: "Optimize deep learning models."

Level: Advanced

Now classify:

Query: "Learn data structures and algorithms."

Output:

Intermediate

Observation:

It is significantly improved recommendation accuracy by clearly defining beginner, intermediate, and advanced levels.

5.Explaination:

Few-shot prompting clearly defines skill boundaries, improving recommendation quality.

Problem Statement 5: Social Media Post Moderation

Problem Description: A social media platform wants to classify posts into Acceptable, Offensive, or Spam.

Tasks:

1. Prepare Sample Data

No	Social Media Post	Category
S1	Have a great day everyone	Acceptable
S2	Buy now and get 90% discount	Spam
S3	You are useless	Offensive
S4	Click this link to win prizes	Spam
S5	Welcome to our online community	Acceptable

2. Zero-shot Prompt:

Prompt Used:

Classify the following post as:

Acceptable, Offensive, or Spam.

Post: "Click here to win free prizes!"

Output:

Spam

Observation:

It struggled with ambiguous or sarcastic posts due to lack of examples.

3. One-shot Prompt:

Prompt Used:

Example:

Post: "Buy now and get 90% off!"

Category: Spam

Now classify:

Post: "You are useless."

Output:

Offensive

Observation:

It improved classification by setting a clear moderation standard.

4.Few-shot Prompt:

Prompt Used:

Example 1:

Post: "Limited offer! Buy now!"

Category: Spam

Example 2:

Post: "I hate you."

Category: Offensive

Example 3:

Post: "Have a great day everyone!"

Category: Acceptable

Now classify:

Post: "Subscribe to my channel for free gifts."

Output:

Spam

Observation:

delivered the most accurate and confident moderation results by learning patterns from multiple examples.

5.Explanation:

Zero-shot prompting struggles due to sarcasm, ambiguity, and lack of context.