AI-Assisted Coding

Week-4.1

2303A51018

Batch-28

Problem statement 1

Customer Email Classification

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 five short sample emails, each belonging to one of

the above categories.

2. Write a zero-shot prompt to classify a given email into one of

the categories without providing any examples.

3. Write a one-shot prompt by including one labeled email

example and ask the model to classify a new email.

4. Write a few-shot prompt by including two or three labeled

email examples and ask the model to classify a new email.

5. Compare the outputs obtained using zero-shot, one-shot, and

few-shot prompting techniques and briefly comment on their

Effectiveness

Customer Email Classification Using Prompt Engineering

1. Sample Customer Emails

No.	Email Content	Category
E1	"I was charged twice for my monthly subscription."	Billing
E2	"My app crashes every time I try to log in."	Technical Support
E3	"I really like the new features in your latest update."	Feedback
E4	"How can I update my contact details?"	Others
E5	"My internet connection stops frequently."	Technical Support

ChatGPT ⌄

Share ···

New chat
Search chats
Images
Apps
Codex
Projects

Your chats
Statistical Summary in Python
Syntax Error Fix
Segment Tree Temperature ...
JSX Syntax Overview
Convert Temperature 2469
Brute Force LC 832
MITM Subset Sum
Context for 3658
LeetCode 263 Ugly Number
Black Image Analysis

Janga Sanjana
Free   Upgrade

2. Zero-Shot Prompting

Prompt Used

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

Email: "I was charged twice for my monthly subscription."
```

Output

Billing

Observation

The model correctly classified the email without being provided with any examples.

3. One-Shot Prompting

Prompt Used

Ask anything

ChatGPT ⌄

Share ···

New chat
Search chats
Images
Apps
Codex
Projects

Your chats
Statistical Summary in Python
Syntax Error Fix
Segment Tree Temperature ...
JSX Syntax Overview
Convert Temperature 2469
Brute Force LC 832
MITM Subset Sum
Context for 3658
LeetCode 263 Ugly Number
Black Image Analysis

Janga Sanjana
Free   Upgrade

3. One-Shot Prompting

Prompt Used

```vbnet
Example:
Email: "My app crashes when I open it."
Category: Technical Support

Now classify the following email into Billing, Technical Support, Feedback, or Others.

Email: "I was charged twice for my monthly subscription."
```

Output

Billing

Observation

Providing one labeled example improved clarity and reduced ambiguity.

Ask anything

ChatGPT ⌄

Share

4. Few-Shot Prompting

Prompt Used

```vbnet
Example 1:
Email: "I was charged an extra amount on my bill."
Category: Billing

Example 2:
Email: "The website is not loading properly."
Category: Technical Support

Example 3:
Email: "The new design looks great."
Category: Feedback

Now classify the following email into Billing, Technical Support, Feedback, or Others.

Email: "I was charged twice for my monthly subscription."
```

Output

Billing

Ask anything

ChatGPT can make mistakes. Check important info. See Cookie Preferences.

ChatGPT ⌄

Share

Output

Billing

Observation

Few-shot prompting produced the most confident and accurate classification.

5. Comparison and Effectiveness

Prompting Technique	Effectiveness
Zero-shot	Works well for clear and simple emails
One-shot	Improves understanding and consistency
Few-shot	Provides highest accuracy and reliability

Conclusion

Zero-shot prompting is useful for straightforward classification tasks. One-shot prompting improves model understanding by providing context, while few-shot prompting offers the best performance by guiding the model with multiple examples. Prompt engineering enables effective email classification without training a new machine learning model.

Ask anything

ChatGPT can make mistakes. Check important info. See Cookie Preferences.

Problem Statement 2

Intent Classification for Chatbot Queries

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 techniques.

Tasks to be Completed

1. Prepare Sample Data

Create 6 short chatbot user queries, each mapped to one of

the four intents.

2. Zero-shot Prompting

Design a prompt that asks the LLM to classify a user query

into the given intent categories without examples.

3. One-shot Prompting

Provide one labeled query in the prompt before classifying a

new query.

4. Few-shot Prompting

Include 3–5 labeled intent examples to guide the LLM before

classifying a new query.

5. Evaluation

Apply all three techniques to the same set of test queries and

document differences in performance.

ChatGPT

Share

Intent Classification for Chatbot Queries Using Prompt Engineering

1. Sample Chatbot User Queries

No.	User Query	Intent
Q1	"I am unable to log into my account."	Account Issue
Q2	"Where is my order?"	Order Status
Q3	"Does this laptop support fast charging?"	Product Inquiry
Q4	"What are your customer service working hours?"	General Question
Q5	"My account got locked after multiple login attempts."	Account Issue
Q6	"When will my package be delivered?"	Order Status

2. Zero-Shot Prompting

Prompt Used

```mathematica
```

Ask anything

ChatGPT can make mistakes. Check important info. See Cookie Preferences.

ChatGPT

Share

Q6	"When will my package be delivered?"	Order Status

2. Zero-Shot Prompting

Prompt Used

```mathematica
Classify the following user query into one of these intents:
Account Issue, Order Status, Product Inquiry, or General Question.

Query: "Where is my order?"
```

Output

Order Status

Observation

The model correctly identified the intent without any prior examples.

3. One-Shot Prompting

Ask anything

ChatGPT can make mistakes. Check important info. See Cookie Preferences.

3. One-Shot Prompting

Prompt Used

```vbnet
Example:
Query: "I am unable to log into my account."
Intent: Account Issue

Now classify the following user query into one of the given intents.

Query: "Where is my order?"
```

Output

Order Status

Observation

Providing one labeled example improved clarity and consistency in classification.

4. Few-Shot Prompting

Prompt Used

```vbnet
Example 1:
Query: "I am unable to log into my account."
Intent: Account Issue

Example 2:
Query: "When will my package arrive?"
Intent: Order Status

Example 3:
Query: "Does this phone support wireless charging?"
Intent: Product Inquiry

Example 4:
Query: "What are your working hours?"
Intent: General Question

Now classify the following user query into one of the given intents.
```
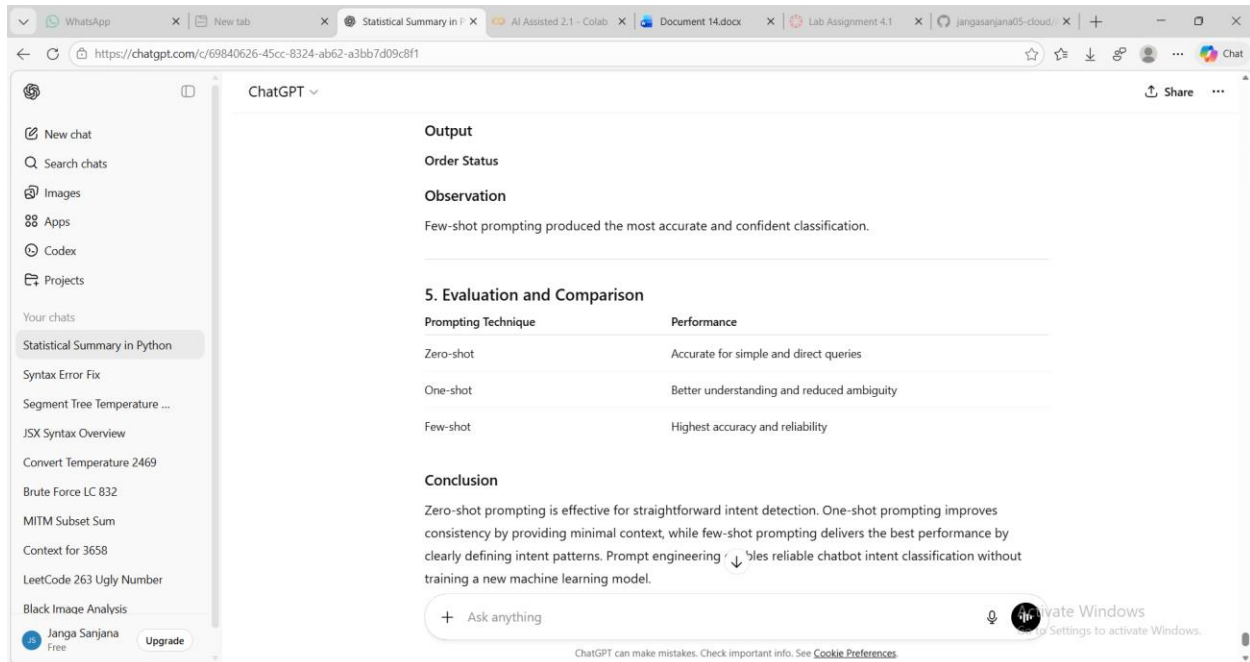
ChatGPT

Output

Order Status

Observation

Few-shot prompting produced the most accurate and confident classification.

5. Evaluation and Comparison

Prompting Technique	Performance
Zero-shot	Accurate for simple and direct queries
One-shot	Better understanding and reduced ambiguity
Few-shot	Highest accuracy and reliability

Conclusion

Zero-shot prompting is effective for straightforward intent detection. One-shot prompting improves consistency by providing minimal context, while few-shot prompting delivers the best performance by clearly defining intent patterns. Prompt engineering enables reliable chatbot intent classification without training a new machine learning model.

Ask anything

Problem Statement 3

Student Feedback Analysis

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

Questions:

a) Write a Zero-shot prompt to classify feedback sentiment.

b) Provide a One-shot prompt with one feedback example.

c) Create a Few-shot prompt using multiple labeled feedback samples.

d) Explain how examples improve sentiment classification Accuracy.

Student Feedback Sentiment Analysis Using Prompt Engineering

a) Zero-Shot Prompt

Prompt:

```
Classify the following student feedback as Positive, Negative, or
Neutral.

Feedback: "The course content was very informative and useful."
```

Output:
 Positive

b) One-Shot Prompt

Example:

Feedback: "The lectures were boring and confusing."

Sentiment: Negative

Now classify the following feedback as Positive, Negative, or Neutral.

Feedback: "The course content was very informative and useful."

Output:
 Positive

c) Few-Shot Prompt

Example 1:

Feedback: "The instructor explained concepts clearly."

Sentiment: Positive

Example 2:

Feedback: "The syllabus is outdated and needs improvement."

Sentiment: Negative

Example 3:

Feedback: "The exam pattern was acceptable."

Sentiment: Neutral

Now classify the following feedback as Positive, Negative, or Neutral.

Feedback: "The course content was very informative and useful."

Output:
 Positive

d) Explanation:

Examples help the language model understand how different sentiments are expressed in text. One-shot prompting provides basic context, while few-shot prompting shows multiple sentiment patterns, reducing ambiguity. This allows the model to better distinguish between positive, negative, and neutral tones, leading to more accurate and consistent sentiment classification.

Problem Statement 4

Course Recommendation System

An online learning platform wants to recommend courses by

classifying learner queries into Beginner, Intermediate, or Advanced

levels.

Questions:

a) Write a Zero-shot prompt to classify learner queries.

b) Create a One-shot prompt with one example query.

c) Develop a Few-shot prompt with multiple labeled queries.

d) Discuss how Few-shot prompting improves recommendation

Quality.

Course Recommendation System Using Prompt Engineering

a) Zero-Shot Prompt

Prompt:

```
Classify the following learner query into one of these levels:
Beginner, Intermediate, or Advanced.

Query: "I am new to Python and want to start learning programming."
```

Output:
 Beginner

b) One-Shot Prompt

Prompt:

```
Example:
Query: "I want to learn the basics of Python."
Level: Beginner

Now classify the following learner query into Beginner, Intermediate,
or Advanced.
```

Query: "I am new to Python and want to start learning programming."

Output:
 Beginner

c) Few-Shot Prompt

Prompt:

```
Example 1:
Query: "I want to learn the basics of Python."
Level: Beginner

Example 2:
Query: "I know Python and want to build real-world projects."
Level: Intermediate

Example 3:
Query: "I want to optimize deep learning models using Python."
Level: Advanced

Now classify the following learner query into Beginner, Intermediate,
or Advanced.

Query: "I am new to Python and want to start learning programming."
```

Output:
 Beginner

d) Discussion

Few-shot prompting improves course recommendations by clearly defining different learning levels through examples. By seeing multiple labeled queries, the model better understands the skill expectations for Beginner, Intermediate, and Advanced learners. This reduces misclassification and ensures that learners receive course recommendations that match their knowledge level, leading to a better learning experience.

Problem Statement 5

Social Media Post Moderation

A social media platform wants to classify posts into Acceptable,

Offensive, or Spam.

Questions:

a) Write a Zero-shot prompt for post moderation.

b) Convert it into a One-shot prompt.

c) Design a Few-shot prompt using multiple examples.

d) Explain the challenges of Zero-shot prompting in content

moderation.

Social Media Post Moderation Using Prompt Engineering

a) Zero-Shot Prompt for Post Moderation

Prompt:

```
Classify the following social media post as Acceptable, Offensive, or
Spam.

Post: "Click here to win exciting prizes now!"
```

Output:
 Spam

b) One-Shot Prompt

Prompt:

Example:
Post: "You are useless and stupid."
Category: Offensive

Now classify the following post as Acceptable, Offensive, or Spam.

Post: "Click here to win exciting prizes now!"

Output:
 Spam

c) Few-Shot Prompt Using Multiple Examples

Prompt:

Example 1:
Post: "Have a great day everyone!"
Category: Acceptable

Example 2:
Post: "You are an idiot."
Category: Offensive

Example 3:
Post: "Win money fast by clicking this link!"
Category: Spam

Now classify the following post as Acceptable, Offensive, or Spam.

Post: "Click here to win exciting prizes now!"

Output:
 Spam

d) Challenges of Zero-Shot Prompting in Content Moderation

Zero-shot prompting can struggle with content moderation because posts may contain sarcasm, slang, or hidden intent. Without examples, the model may misclassify borderline or ambiguous content. It may also fail to recognize subtle offensive language or spam disguised as normal messages. Providing examples through one-shot or few-shot prompting improves accuracy by giving the model clearer context and category boundaries.

Conclusion

Zero-shot prompting is useful for simple moderation tasks, but one-shot and few-shot prompting significantly improve reliability and accuracy in real-world social media content moderation.