

# AI ASSISTED CODING

## LAB-4.1

**B.VIGHNESH**  
**230351795**  
**Batch-12**

### **Problem Statement 0:**

A news aggregation platform wants to automatically categorize headlines into Politics, Sports, Technology, and Entertainment without training a machine learning model.

#### **1. Prepare Sample Data**

**Collect 10 news headlines, each belonging to one of the four categories.**

The 83rd Golden Globe Awards took place, with films like One Battle After Another, Hamnet, and Adolescence winning top honors.

Timothée Chalamet won his first Golden Globe for Best Actor in Marty Supreme and publicly thanked his partner, Kylie Jenner, who was present at the ceremony.

The song "Golden" from the animated film KPop Demon Hunters won Best Original Song, marking a first for a K-pop track at the Golden Globes.

Donald Trump has declared himself the "Acting President of Venezuela" following a U.S. military strike that captured Nicolas Maduro.

New U.S. Ambassador Sergio Gor has officially taken charge in New Delhi, inviting India to the "Pax Silica" initiative and signaling a potential visit from Trump in 2026 or 2027.

India Wins 1st ODI vs. NZ: India defeated New Zealand by 4 wickets in the series opener at Vadodara.

Virat Kohli Records: Kohli became the fastest player to reach 28,000 international runs, surpassing Sachin Tendulkar's record by 20 innings.

T20 World Cup Controversy: Tensions between the BCB and ICC continue; Pakistan has reportedly offered to host Bangladesh's games for the upcoming tournament due to diplomatic friction with India.

OpenAI and SoftBank are jointly investing \$1 billion in SB Energy to expand data center and power infrastructure for their "Stargate" AI initiative.

Google is partnering with major retailers like Walmart and Shopify to integrate shopping features within its Gemini AI chatbot, aiming to transform the retail experience.

#### **2. Zero-shot Prompting**

##### **Prompt:**

""""Categorize these headlines

The 83rd Golden Globe Awards took place, with films like One Battle After Another, Hamnet, and Adolescence winning top honors.

Timothée Chalamet won his first Golden Globe for Best Actor in Marty Supreme and publicly thanked his partner, Kylie Jenner, who was present at the ceremony.

The song "Golden" from the animated film KPop Demon Hunters won Best Original Song, marking a first for a K-pop track at the Golden Globes.

Donald Trump has declared himself the "Acting President of Venezuela" following a U.S. military strike that captured Nicolas Maduro.

New U.S. Ambassador Sergio Gor has officially taken charge in New Delhi, inviting India to the "Pax Silica" initiative and signaling a potential visit from Trump in 2026 or 2027.

India Wins 1st ODI vs. NZ: India defeated New Zealand by 4 wickets in the series opener at Vadodara.

Virat Kohli Records: Kohli became the fastest player to reach 28,000 international runs, surpassing Sachin Tendulkar's record by 20 innings.

T20 World Cup Controversy: Tensions between the BCB and ICC continue; Pakistan has reportedly offered to host Bangladesh's games for the upcoming tournament due to diplomatic friction with India.

OpenAI and SoftBank are jointly investing \$1 billion in SB Energy to expand data center and power infrastructure for their "Stargate" AI initiative.

Google is partnering with major retailers like Walmart and Shopify to integrate shopping features within its Gemini AI chatbot, aiming to transform the retail experience into 'Entertainment', 'Politics', 'Sports', and 'Technology' categories."'"

### Given Code:

```
**Categorize these headlines.
The 83rd Golden Globe Awards took place, with films like One Night After Another, Heret, and Adolescence winning top honors.
Matthew McConaughey won his first Golden Globe for Best Actor in Many Summers and publicly thanked his partner, Kylie Jenner, who was present at the ceremony.
The song "Golden" from the animated film KPop Demon Hunters won Best Original Song, marking a first for a K-pop track at the Golden Globes.
Donald Trump has declared himself the "Acting President of Venezuela" following a U.S. military strike that captured Nicolas Maduro.
New U.S. Ambassador Sergio Gor has officially taken charge in New Delhi, inviting India to the "Pax Silica" initiative and signaling a potential visit from Trump in 2026 or 2027.
India Wins 1st ODI vs. NZ: India defeated New Zealand by 4 wickets in the series opener at Vadodara.
Virat Kohli Records: Kohli became the fastest player to reach 28,000 international runs, surpassing Sachin Tendulkar's record by 20 innings.
T20 World Cup Controversy: Tensions between the BCB and ICC continue; Pakistan has reportedly offered to host Bangladesh's games for the upcoming tournament due to diplomatic friction with India.
OpenAI and SoftBank are jointly investing $1 billion in SB Energy to expand data center and power infrastructure for their "Stargate" AI initiative.
Google is partnering with major retailers like Walmart and Shopify to integrate shopping features within its Gemini AI chatbot, aiming to transform the retail experience into 'Entertainment', 'Politics', 'Sports', and 'Technology' categories."'"
```

```
headlines = [
    "The 83rd Golden Globe Awards took place, with films like One Night After Another, Heret, and Adolescence winning top honors.",
    "Matthew McConaughey won his first Golden Globe for Best Actor in Many Summers and publicly thanked his partner, Kylie Jenner, who was present at the ceremony.",
    "The song \"Golden\" from the animated film KPop Demon Hunters won Best Original Song, marking a first for a K-pop track at the Golden Globes.",
    "Donald Trump has declared himself the \"Acting President of Venezuela\" following a U.S. military strike that captured Nicolas Maduro.",
    "New U.S. Ambassador Sergio Gor has officially taken charge in New Delhi, inviting India to the \"Pax Silica\" initiative and signaling a potential visit from Trump in 2026 or 2027.",
    "India Wins 1st ODI vs. NZ: India defeated New Zealand by 4 wickets in the series opener at Vadodara.",
    "Virat Kohli Records: Kohli became the fastest player to reach 28,000 international runs, surpassing Sachin Tendulkar's record by 20 innings.",
    "T20 World Cup Controversy: Tensions between the BCB and ICC continue; Pakistan has reportedly offered to host Bangladesh's games for the upcoming tournament due to diplomatic friction with India.",
    "OpenAI and SoftBank are jointly investing $1 billion in SB Energy to expand data center and power infrastructure for their \"Stargate\" AI initiative.",
    "Google is partnering with major retailers like Walmart and Shopify to integrate shopping features within its Gemini AI chatbot, aiming to transform the retail experience into 'Entertainment', 'Politics', 'Sports', and 'Technology' categories."'"
```

```
categories = [
    "Entertainment": [],
    "Politics": [],
    "Sports": [],
    "Technology": []
]

for headline in headlines:
    if any(keyword in headline for keyword in ["Golden Globes", "film", "actor", "song", "K-pop"]):
        categories["Entertainment"].append(headline)
    elif any(keyword in headline for keyword in ["Donald Trump", "ambassador", "Venezuela", "India"]):
        categories["Politics"].append(headline)
    elif any(keyword in headline for keyword in ["ODI", "Virat Kohli", "T20 World Cup", "controversy"]):
        categories["Sports"].append(headline)
    elif any(keyword in headline for keyword in ["OpenAI", "SoftBank", "Google", "AI", "data center"]):
        categories["Technology"].append(headline)

for category, items in categories.items():
    print(f"\n{category} Headlines:")
    for item in items:
        print(f"- {item}\n")
```

### Output:

```
Entertainment Headlines:
- The 83rd Golden Globe Awards took place, with films like One Night After Another, Heret, and Adolescence winning top honors.
- Matthew McConaughey won his first Golden Globe for Best Actor in Many Summers and publicly thanked his partner, Kylie Jenner, who was present at the ceremony.
- The song "Golden" from the animated film KPop Demon Hunters won Best Original Song, marking a first for a K-pop track at the Golden Globes.

Politics Headlines:
- Donald Trump has declared himself the "Acting President of Venezuela" following a U.S. military strike that captured Nicolas Maduro.
- New U.S. Ambassador Sergio Gor has officially taken charge in New Delhi, inviting India to the "Pax Silica" initiative and signaling a potential visit from Trump in 2026 or 2027.
- India Wins 1st ODI vs. NZ: India defeated New Zealand by 4 wickets in the series opener at Vadodara.
- T20 World Cup Controversy: Tensions between the BCB and ICC continue; Pakistan has reportedly offered to host Bangladesh's games for the upcoming tournament due to diplomatic friction with India.

Sports Headlines:
- Virat Kohli Records: Kohli became the fastest player to reach 28,000 international runs, surpassing Sachin Tendulkar's record by 20 innings.

Technology Headlines:
- OpenAI and SoftBank are jointly investing $1 billion in SB Energy to expand data center and power infrastructure for their "Stargate" AI initiative.
- Google is partnering with major retailers like Walmart and Shopify to integrate shopping features within its Gemini AI chatbot, aiming to transform the retail experience into 'Entertainment', 'Politics', 'Sports', and 'Technology' categories."'"
```

### 3. One-shot Prompting

#### Prompt:

""""Categorize these headlines

The 83rd Golden Globe Awards took place, with films like One Battle After Another, Hamnet, and Adolescence winning top honors.

Timothée Chalamet won his first Golden Globe for Best Actor in Marty Supreme and publicly thanked his partner, Kylie Jenner, who was present at the ceremony.

The song "Golden" from the animated film KPop Demon Hunters won Best Original Song, marking a first for a K-pop track at the Golden Globes.

Donald Trump has declared himself the "Acting President of Venezuela" following a U.S. military strike that captured Nicolas Maduro.

New U.S. Ambassador Sergio Gor has officially taken charge in New Delhi, inviting India to the "Pax Silica" initiative and signaling a potential visit from Trump in 2026 or 2027.

India Wins 1st ODI vs. NZ: India defeated New Zealand by 4 wickets in the series opener at Vadodara.

Virat Kohli Records: Kohli became the fastest player to reach 28,000 international runs, surpassing Sachin Tendulkar's record by 20 innings.

T20 World Cup Controversy: Tensions between the BCB and ICC continue; Pakistan has reportedly offered to host Bangladesh's games for the upcoming tournament due to diplomatic friction with India.

OpenAI and SoftBank are jointly investing \$1 billion in SB Energy to expand data center and power infrastructure for their "Stargate" AI initiative.-Technology

Google is partnering with major retailers like Walmart and Shopify to integrate shopping features within its Gemini AI chatbot, aiming to transform the retail experience. into 'Entertainment', 'Politics', 'Sports', and 'Technology' categories.""""

#### Given Code:

```
headlines = [
    "The 83rd Golden Globe Awards took place, with films like One Battle After Another, Hamnet, and Adolescence winning top honors.",
    "Timothée Chalamet won his first Golden Globe for Best Actor in Marty Supreme and publicly thanked his partner, Kylie Jenner, who was present at the ceremony.",
    "The song \"Golden\" from the animated film KPop Demon Hunters won Best Original Song, marking a first for a K-pop track at the Golden Globes.",
    "Donald Trump has declared himself the \"Acting President of Venezuela\" following a U.S. military strike that captured Nicolas Maduro.",
    "New U.S. Ambassador Sergio Gor has officially taken charge in New Delhi, inviting India to the \"Pax Silica\" initiative and signaling a potential visit from Trump in 2026 or 2027."
]

categories = [
    "Entertainment": [],
    "Politics": [],
    "Sports": [],
    "Technology": []
]

for headline in headlines:
    if any(keyword in headline for keyword in ["Golden Globe", "Film", "Actor", "Song", "K-pop"]):
        categories["Entertainment"].append(headline)
    elif any(keyword in headline for keyword in ["Donald Trump", "Ambassador", "Venezuela", "India"]):
        categories["Politics"].append(headline)
    elif any(keyword in headline for keyword in ["Kohli", "Virat Kohli", "T20 World Cup", "Cricket"]):
        categories["Sports"].append(headline)
    elif any(keyword in headline for keyword in ["OpenAI", "SoftBank", "Google", "AI", "Datacenter"]):
        categories["Technology"].append(headline)

for category, items in categories.items():
    print(f"\n{category}:")
    for item in items:
        print(f"- {item}")
```

#### Output:

**Entertainment Headlines:**

- The 83rd Golden Globe Awards took place, with films like One Battle After Another, Hamnet, and Adolescence winning top honors.
- Timothée Chalamet won his first Golden Globe for Best Actor in Marty Supreme and publicly thanked his partner, Kylie Jenner, who was present at the ceremony.
- The song "Golden" from the animated film KPop Demon Hunters won Best Original Song, marking a first for a K-pop track at the Golden Globes.

**Politics Headlines:**

- Donald Trump has declared himself the "Acting President of Venezuela" following a U.S. military strike that captured Nicolas Maduro.
- New U.S. Ambassador Sergio Gor has officially taken charge in New Delhi, inviting India to the "Pax Silica" initiative and signaling a potential visit from Trump in 2026 or 2027.
- India Wins 1st ODI vs. NZ: India defeated New Zealand by 4 wickets in the series opener at Vadodara.
- T20 World Cup Controversy: Tensions between the BCB and ICC continue; Pakistan has reportedly offered to host Bangladesh's games for the upcoming tournament due to diplomatic friction with India.

**Sports Headlines:**

- Virat Kohli Records: Kohli became the fastest player to reach 28,000 international runs, surpassing Sachin Tendulkar's record by 20 innings.

**Technology Headlines:**

- OpenAI and SoftBank are jointly investing \$1 billion in SB Energy to expand data center and power infrastructure for their "Stargate" AI initiative.-Technology
- Google is partnering with major retailers like Walmart and Shopify to integrate shopping features within its Gemini AI chatbot, aiming to transform the retail experience.

## 4. Few-shot Prompting

**Prompt:**

""""Categorize these headlines

The 83rd Golden Globe Awards took place, with films like One Battle After Another, Hamnet, and Adolescence winning top honors.-Entertainment

Timothée Chalamet won his first Golden Globe for Best Actor in Marty Supreme and publicly thanked his partner, Kylie Jenner, who was present at the ceremony.

The song "Golden" from the animated film KPop Demon Hunters won Best Original Song, marking a first for a K-pop track at the Golden Globes.

Donald Trump has declared himself the "Acting President of Venezuela" following a U.S. military strike that captured Nicolas Maduro.-Politics

New U.S. Ambassador Sergio Gor has officially taken charge in New Delhi, inviting India to the "Pax Silica" initiative and signaling a potential visit from Trump in 2026 or 2027.

India Wins 1st ODI vs. NZ: India defeated New Zealand by 4 wickets in the series opener at Vadodara.Sports

Virat Kohli Records: Kohli became the fastest player to reach 28,000 international runs, surpassing Sachin Tendulkar's record by 20 innings.

T20 World Cup Controversy: Tensions between the BCB and ICC continue; Pakistan has reportedly offered to host Bangladesh's games for the upcoming tournament due to diplomatic friction with India.

OpenAI and SoftBank are jointly investing \$1 billion in SB Energy to expand data center and power infrastructure for their "Stargate" AI initiative.-Technology

Google is partnering with major retailers like Walmart and Shopify to integrate shopping features within its Gemini AI chatbot, aiming to transform the retail experience. into 'Entertainment', 'Politics', 'Sports', and 'Technology' categories.""""

**Given Code:**

```

headlines = []
    "The 83rd Golden Globe Awards took place, with films like One Battle After Another, Hamnet, and Adolescence winning top honors.-Entertainment",
    "Timothée Chalamet won his first Golden Globe for Best Actor in 'Marty Supreme' and publicly thanked his partner, Kylie Jenner, who was present at the ceremony.",
    "The song 'Golden' from the animated film KPop Demon Hunters won Best Original Song, marking a first for a K-pop track at the Golden Globes.",
    "Donald Trump has declared himself the 'acting President of Venezuela' following a U.S. military strike that captured Nicolas Maduro.-Politics",
    "New U.S. Ambassador Sergio Carrasquillo has officially taken charge in New Delhi, inviting India to the 'Pax Silica' initiative and signaling a potential visit in 2026 or 2027.",
    "India wins 1st ODI vs. NZ; India defeated New Zealand by 4 wickets in the series opener at Vadodara.-Sports",
    "Virat Kohli Records: Kohli became the fastest player to reach 28,000 international runs, surpassing Sachin Tendulkar's record by 20 innings.-Sports",
    "T20 World Cup Controversy: Tensions between the ICC and PCB continue; Pakistan has reportedly offered to host Bangladesh's games for the upcoming tournament and Softbank are jointly investing $1 billion in Sri Energy to expand data center and power infrastructure for their 'Stargate' AI initiative.-Technology",
    "Google is partnering with major retailers like Walmart and Shopify to integrate shopping features within its Gemini AI chatbot, aiming to transform the retail experience."
]

categories = {
    "Entertainment": [],
    "Politics": [],
    "Sports": [],
    "Technology": []
}

for headline in headlines:
    if "entertainment" in headline or "Golden Globe" in headline or "KPop" in headline:
        categories["Entertainment"].append(headline)
    elif "politics" in headline or "assassination" in headline or "Venezuela" in headline:
        categories["Politics"].append(headline)
    elif "sports" in headline or "ODI" in headline or "Kohli" in headline or "T20 World Cup" in headline:
        categories["Sports"].append(headline)
    elif "technology" in headline or "Oscar" in headline or "Google" in headline:
        categories["Technology"].append(headline)

for category, heads in categories.items():
    print(f"\n{category} Headlines:")
    for head in heads:
        print(f"- {head}")

```

## Output:

```

entertainment headlines:
- The 83rd Golden Globe Awards took place, with films like One Battle After Another, Hamnet, and Adolescence winning top honors.-Entertainment
- Timothée Chalamet won his first Golden Globe for Best Actor in 'Marty Supreme' and publicly thanked his partner, Kylie Jenner, who was present at the ceremony.
- The song 'Golden' from the animated film KPop Demon Hunters won Best Original Song, marking a first for a K-pop track at the Golden Globes.

politics headlines:
- Donald Trump has declared himself the 'Acting President of Venezuela' following a U.S. military strike that captured Nicolas Maduro.-Politics
- New U.S. Ambassador Sergio Carrasquillo has officially taken charge in New Delhi, inviting India to the 'Pax Silica' initiative and signaling a potential visit in 2026 or 2027.

Sports Headlines:
- India wins 1st ODI vs. NZ; India defeated New Zealand by 4 wickets in the series opener at Vadodara.-Sports
- Virat Kohli Records: Kohli became the fastest player to reach 28,000 international runs, surpassing Sachin Tendulkar's record by 20 innings.-Sports
- T20 World Cup Controversy: Tensions between the ICC and PCB continue; Pakistan has reportedly offered to host Bangladesh's games for the upcoming tournament due to diplomatic friction with India.

Technology Headlines:
- Softbank and Google are jointly investing $1 billion in Sri Energy to expand data center and power infrastructure for their 'Stargate' AI initiative.-Technology
- Google is partnering with major retailers like Walmart and Shopify to integrate shopping features within its Gemini AI chatbot, aiming to transform the retail experience.

```

## 5. Evaluation

**Zero-Shot:** In zero shot the code classified using the keywords that are common in the headlines since there are no examples. So that's why it confused 2 sports headline and classified it as politics.

**One-Shot:** In one shot also the ai got confused and did the same thing as above because I only gave one example

**Few-Shot:** In few shot I gave examples for 4 headlines and it understood the keywords and classified correctly.

Therefore, giving proper examples gives correct output. So, I think Few-shot is more efficient.

## 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

## **1. Zero-Shot Prompting:**

### **Prompt:**

"""Write code to categorize the following emails into Billing, Technical Support, Feedback, Others

Subject: Question Regarding Recent Invoice

Hello,

I hope you are doing well. I am writing to ask for clarification regarding my recent invoice. There are a few charges listed that I do not fully understand, and I would appreciate it if you could provide more details.

Please let me know if you need any additional information from my side. I look forward to your response.

Thank you for your assistance.

Best regards

Subject: Discrepancy Found on Billing Statement

Hello,

I am contacting you regarding a discrepancy I noticed on my latest billing statement. The amount charged does not seem to match my expected usage or plan.

Could you please review this and advise on how it can be corrected? I would appreciate your help in resolving this matter as soon as possible.

Thank you for your support.

Best regards

Subject: Request for Technical Assistance

Hello,

I am currently experiencing a technical issue with the service. The problem started recently and has been affecting my ability to use the platform as expected.

I would appreciate your guidance on how to resolve this issue or any troubleshooting steps I should follow. Please let me know if further details are required.

Thank you for your time and support.

Best regards

Subject: Feedback on Recent Experience Hello,

I wanted to take a moment to share my feedback regarding my recent experience with your service. Overall, I found it to be very helpful and easy to use.

I appreciate the effort your team puts into maintaining quality service, and I hope this feedback is useful for future improvements.

Thank you for your time.

Kind regards

Subject: General Inquiry

Hello,

I am writing to inquire about some general information related to your services. I was unable to find clear details on your website and would appreciate your guidance.

Please let me know the best way to proceed or if there is someone specific I should contact. Thank you in advance for your help.

Best regards """

#### **Given Code:**

```
category_email(category, email)
    subject = email['subject']
    body = email['body']

    if "Invoice" in subject or "Billing" in subject or "Charges" in body:
        return "Billing"
    elif "Technical" in subject or "Issue" in body or "Troubleshooting" in body:
        return "Technical Support"
    elif "Feedback" in subject or "Experience" in body:
        return "Feedback"
    else:
        return "Others"
category_email_usage
emails = [
    {
        "subject": "Question Regarding Recent Invoice",
        "body": """Hello,
I hope you are doing well. I am writing to ask for clarification regarding my recent invoice. There are a few charges listed that I do not fully understand, and I would appreciate your guidance on how to proceed. Please let me know if you need any additional information from my side. I look forward to your responses.
Thank you for your assistance,
Best regards""",
        "category": "Others"
    },
    {
        "subject": "Discrepancy Found on Billing Statement",
        "body": """Hello,
I am contacting you regarding a discrepancy I noticed on my latest billing statement. The amount charged does not seem to match my expected usage or plan. Could you please review this and advise on how it can be corrected? I would appreciate your help in resolving this matter as soon as possible.
Thank you for your support,
Best regards""",
        "category": "Billing"
    },
    {
        "subject": "Request for Technical Assistance",
        "body": """Hello,
I am currently experiencing a technical issue with the service. The problem started recently and has been affecting my ability to use the platform as expected. I would appreciate your guidance on how to resolve this issue or any troubleshooting steps I should follow.
Please let me know if further details are required.
Thank you for your time and support,
Best regards""",
        "category": "Technical Support"
    },
    {
        "subject": "Feedback on Recent Experience",
        "body": """Hello,
I wanted to take a moment to share my feedback regarding my recent experience with your service. Overall, I found it to be very helpful and easy to use. I appreciate the effort you put into maintaining quality service, and I hope this feedback is useful for future improvements.
Thank you for your time,
Best regards""",
        "category": "Feedback"
    },
    {
        "subject": "General Inquiry",
        "body": """Hello,
I am writing to inquire about some general information related to your services. I was unable to find clear details on your website and would appreciate your guidance. Please let me know the best way to proceed or if there is someone specific I should contact. Thank you in advance for your help.
Best regards""",
        "category": "Others"
    }
]
for email in emails:
    category = category_email(email['subject'], email['body'])
    print(f"Subject: {email['subject']}\nCategory: {category}\n\n")
```

#### **Output:**

```
Subject: Question Regarding Recent Invoice
Category: Billing

Subject: Discrepancy Found on Billing Statement
Category: Billing

Subject: Request for Technical Assistance
Category: Technical Support

Subject: Feedback on Recent Experience
Category: Feedback

Subject: General Inquiry
Category: Others
```

#### **2. One-Shot Prompting:**

**Prompt:**

""""Write code to categorize the following emails into Billing, Technical Support, Feedback, Others  
Subject: Question Regarding Recent Invoice

Hello,

I hope you are doing well. I am writing to ask for clarification regarding my recent invoice. There are a few charges listed that I do not fully understand, and I would appreciate it if you could provide more details.

Please let me know if you need any additional information from my side. I look forward to your response.

Thank you for your assistance.

Best regards-Billing

Subject: Discrepancy Found on Billing Statement

Hello,

I am contacting you regarding a discrepancy I noticed on my latest billing statement. The amount charged does not seem to match my expected usage or plan.

Could you please review this and advise on how it can be corrected? I would appreciate your help in resolving this matter as soon as possible.

Thank you for your support.

Best regards

Subject: Request for Technical Assistance

Hello,

I am currently experiencing a technical issue with the service. The problem started recently and has been affecting my ability to use the platform as expected.

I would appreciate your guidance on how to resolve this issue or any troubleshooting steps I should follow. Please let me know if further details are required.

Thank you for your time and support.

Best regards

Subject: Feedback on Recent Experience Hello,

I wanted to take a moment to share my feedback regarding my recent experience with your service. Overall, I found it to be very helpful and easy to use.

I appreciate the effort your team puts into maintaining quality service, and I hope this feedback is useful for future improvements.

Thank you for your time.

Kind regards

Subject: General Inquiry

Hello,

I am writing to inquire about some general information related to your services. I was unable to find clear details on your website and would appreciate your guidance.

Please let me know the best way to proceed or if there is someone specific I should contact. Thank you in advance for your help.

Best regards """"

### **Given Code:**

```
def categorize_email(subject, body):
    subject = subject.lower()
    body = body.lower()

    if "invoice" in subject or "billing" in subject or "charge" in body:
        return "Billing"
    elif "technical issue" in body or "technical assistance" in subject or "troubleshooting" in body:
        return "Technical Support"
    elif "feedback" in subject or "experience" in body:
        return "Feedback"
    else:
        return "Others"

# Example usage
emails = [
    {
        "Subject": "Question Regarding Recent Invoice",
        "Body": "Hello, I hope you are doing well. I am writing to ask for clarification regarding my recent invoice. There are a few charges listed that I do not fully understand, and I would appreciate it if you could provide more details. Thank you."
    },
    {
        "Subject": "Discrepancy Found on Billing Statement",
        "Body": "Hello, I am contacting you regarding a discrepancy I noticed in my latest billing statement. The amount charged does not seem to match my expected usage or plan. Could you please investigate this for me?"
    },
    {
        "Subject": "Request for Technical Assistance",
        "Body": "Hello, I am currently experiencing a technical issue with the service. The problem started recently and has been affecting my ability to use the platform as expected. I would appreciate your help in resolving this issue quickly."
    },
    {
        "Subject": "Feedback on Recent Experience",
        "Body": "Hello, I wanted to take a moment to share my feedback regarding my recent experience with your service. Overall, I found it to be very helpful and easy-to-use. I appreciate your efforts in providing such great support!"
    },
    {
        "Subject": "General Inquiry",
        "Body": "Hello, I am writing to inquire about some general information related to your service. I was unable to find clear details on your website and would appreciate your guidance on this matter."
    }
]

for email in emails:
    category = categorize_email(email["Subject"], email["Body"])
    print(f"Subject: {email['Subject']}\nCategory: {category}\n\n")
```

### **Output:**

```
Subject: Question Regarding Recent Invoice
Category: Billing

Subject: Discrepancy Found on Billing Statement
Category: Billing

Subject: Request for Technical Assistance
Category: Technical Support

Subject: Feedback on Recent Experience
Category: Feedback

Subject: General Inquiry
Category: Others
```

### **3. Few-Shot Prompting:**

#### **Prompt:**

""""Write code to categorize the following emails into Billing, Technical Support, Feedback, Others

Subject: Question Regarding Recent Invoice

Hello,

I hope you are doing well. I am writing to ask for clarification regarding my recent invoice. There are a few charges listed that I do not fully understand, and I would appreciate it if you could provide more details.

Please let me know if you need any additional information from my side. I look forward to your response.

Thank you for your assistance.

Best regards-Billing

Subject: Discrepancy Found on Billing Statement

Hello,

I am contacting you regarding a discrepancy I noticed on my latest billing statement. The amount charged does not seem to match my expected usage or plan.

Could you please review this and advise on how it can be corrected? I would appreciate your help in resolving this matter as soon as possible.

Thank you for your support.

Best regards

Subject: Request for Technical Assistance

Hello,

I am currently experiencing a technical issue with the service. The problem started recently and has been affecting my ability to use the platform as expected.

I would appreciate your guidance on how to resolve this issue or any troubleshooting steps I should follow. Please let me know if further details are required.

Thank you for your time and support.

Best regards-Technical Support

Subject: Feedback on Recent Experience Hello,

I wanted to take a moment to share my feedback regarding my recent experience with your service. Overall, I found it to be very helpful and easy to use.

I appreciate the effort your team puts into maintaining quality service, and I hope this feedback is useful for future improvements.

Thank you for your time.

Kind regards-Feedback

Subject: General Inquiry

Hello,

I am writing to inquire about some general information related to your services. I was unable to find clear details on your website and would appreciate your guidance.

Please let me know the best way to proceed or if there is someone specific I should contact. Thank you in advance for your help.

Best regards """"

**Given Code:**

```

def categorize_email(subject, body):
    subject = subject.lower()
    body = body.lower()

    if "subject" in subject or "Billing" in body or "charges" in body:
        return "Billing"
    elif "technical issue" in body or "troubleshooting" in body or "technical assistance" in subject:
        return "Technical Support"
    elif "feedback" in subject or "experience" in body:
        return "Feedback"
    else:
        return "Others"

# Sample usage
emails = [
    {"subject": "Question regarding Recent Invoice", "body": "Hello, I hope you are doing well. I am writing to ask for clarification regarding my recent invoice. There are a few charges I'm not understanding. Could you please review my billing statement?", "category": "General Inquiry", "id": 1},
    {"subject": "Request for Technical Assistance", "body": "Hello, I am currently experiencing a technical issue with the service. The problem started recently and has been affecting my productivity.", "category": "Technical Support", "id": 2},
    {"subject": "Feedback on Recent Experience", "body": "Hello, I wanted to take a moment to share my feedback regarding my recent experience with your service. Overall, I found it to be excellent.", "category": "Feedback", "id": 3},
    {"subject": "General Inquiry", "body": "Hello, I am writing to inquire about some general information related to your services. I was unable to find these details on your website.", "category": "General Inquiry", "id": 4}
]

for email in emails:
    category = categorize_email(email['subject'], email['body'])
    print(f"Email {email['id']} - Category: {category}")

```

## Output:

```

Query: I can't log into my account even after resetting my password.
Category: Account Issue

Query: Why was my account locked after yesterday's login attempt?
Category: Account Issue

Query: Can you tell me where my order #45821 is right now?
Category: Order Status

Query: It's been three days—has my package shipped yet?
Category: Order Status

Query: Does this laptop support 32GB RAM and Windows 11?
Category: Product Inquiry

Query: What are your customer support hours on weekends?
Category: Product Inquiry

```

## 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.

### 1. Zero-Shot Prompting:

#### Prompt:

""""Write a code to classify a user query into categories-Account Issue, Order Status, Product Inquiry, or General Question using prompt engineering techniques.

I can't log into my account even after resetting my password.

Why was my account locked after yesterday's login attempt?

Can you tell me where my order #45821 is right now? It's been three days—has my package shipped yet? Does this laptop support 32GB RAM and Windows 11? What are your customer support hours on weekends?  
"""

#### Given Code:

```

def classify_query(query):
    prompt = """Classify the following user query into one of the categories: Account Issue, Order Status, Product Inquiry, or General Question.
User Query: {query}
Category: """
    # Simulating a response from a language model
    if "account" in query.lower() or "log into" in query.lower() or "password" in query.lower() or "locked" in query.lower():
        return "Account Issue"
    elif "order" in query.lower() or "package" in query.lower() or "shipped" in query.lower():
        return "Order Status"
    elif "support" in query.lower() or "RAM" in query.lower() or "Windows" in query.lower():
        return "Product Inquiry"
    else:
        return "General Question"
# Test Cases
queries = [
    "I can't log into my account even after resetting my password.",
    "Why was my account locked after yesterday's login attempt?",
    "Can you tell me where my order #45821 is right now?",
    "It's been three days—has my package shipped yet?",
    "Does this laptop support 32GB RAM and Windows 11?",
    "What are your customer support hours on weekends?"
]
for query in queries:
    category = classify_query(query)
    print(f"Query: {query}\nCategory: {category}\n")

```

## Output:

```

Subject: Question Regarding Recent Invoice
Category: Billing

Subject: Discrepancy Found on Billing Statement
Category: Billing

Subject: Request for Technical Assistance
Category: Technical Support

Subject: Feedback on Recent Experience
Category: Feedback

Subject: General Inquiry
Category: Others

```

## 2. One-Shot Prompting:

### Prompt:

"""\nWrite a code to classify a user query into categories-Account Issue, Order Status, Product Inquiry, or General Question using prompt engineering techniques.

I can't log into my account even after resetting my password.-Account Issue

Why was my account locked after yesterday's login attempt?

Can you tell me where my order #45821 is right now? It's been three days—has my package shipped yet? Does this laptop support 32GB RAM and Windows 11? What are your customer support hours on weekends?  
"""

### Given Code:

```

def classify_query(user_query):
    """Classify the following user query into one of the categories: Account Issue, Order Status, Product Inquiry, or General Question.
    User Query: "{user_query}"
    Category:"""

    # Here we would normally call a language model API to get the classification
    # For demonstration purposes, we'll use a simple keyword-based approach
    user_query_lower = user_query.lower()
    if "account" in user_query_lower or "login" in user_query_lower or "password" in user_query_lower:
        return "Account Issue"
    elif "order" in user_query_lower or "package" in user_query_lower or "shipping" in user_query_lower:
        return "Order Status"
    elif "product" in user_query_lower or "laptop" in user_query_lower or "inquiry" in user_query_lower:
        return "Product Inquiry"
    else:
        return "General Question"
# Example usage
if __name__ == "__main__":
    queries = [
        "I can't log into my account even after resetting my password.",
        "Why was my account locked after yesterday's login attempt?",
        "Can you tell me where my order #45821 is right now?",
        "It's been three days--has my package shipped yet?",
        "Does this laptop support 32GB RAM and Windows 11?",
        "What are your customer support hours on weekends?"
    ]
    for query in queries:
        category = classify_query(query)
        print(f"User Query: '{query}' - Category: {category}")

```

## Output:

```

User Query: "I can't log into my account even after resetting my password." - Category: Account Issue
User Query: "Why was my account locked after yesterday's login attempt?" - Category: Account Issue
User Query: "Can you tell me where my order #45821 is right now?" - Category: Order Status
User Query: "It's been three days--has my package shipped yet?" - Category: Order Status
User Query: "Does this laptop support 32GB RAM and Windows 11?" - Category: Product Inquiry
User Query: "What are your customer support hours on weekends?" - Category: General Question

```

## Student Feedback Analysis

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

### 1. Zero-Shot Prompting:

#### Prompt:

""""Write a python code to categorize the feedback of university students into positive, negative, and neutral

The professor explained concepts clearly, and the course materials were very helpful.

I really enjoyed this class—assignments were engaging and relevant to real-world scenarios.

The teaching staff was supportive and always available to answer questions.

The lectures felt disorganized, and it was hard to understand what was expected.

Grading took too long, and feedback on assignments was minimal.

The course workload was overwhelming compared to the credit hours.

The course covered the topics listed in the syllabus.

Lectures were held twice a week, and attendance was mandatory.

The class used a mix of exams, quizzes, and assignments for evaluation. """"

### Given Code:

```

def categorize_feedback(feedback):
    positive_keywords = ['clearly', 'helpful', 'enjoyed', 'engaging', 'relevant', 'supportive', 'available']
    negative_keywords = ['disorganized', 'hard', 'took too long', 'minimal', 'overwhelming']

    feedback_lower = feedback.lower()

    if any(keyword in feedback_lower for keyword in positive_keywords):
        return 'Positive'
    elif any(keyword in feedback_lower for keyword in negative_keywords):
        return 'Negative'
    else:
        return 'Neutral'

# Sample Feedbacks
feedbacks = [
    "The professor explained concepts clearly, and the course materials were very helpful.",
    "I really enjoyed this class—assignments were engaging and relevant to real-world scenarios.",
    "The teaching staff was supportive and always available to answer questions.",
    "The lectures felt disorganized, and it was hard to understand what was expected.",
    "Grading took too long, and feedback on assignments was minimal.",
    "The course workload was overwhelming compared to the credit hours.",
    "The course covered the topics listed in the syllabus.",
    "Lectures were held twice a week, and attendance was mandatory.",
    "The class used a mix of exams, quizzes, and assignments for evaluation."
]
# Categorize and print feedbacks
for feedback in feedbacks:
    category = categorize_feedback(feedback)
    print(f'Feedback: "{feedback}"\nCategory: {category}\n')

```

## Output:

```

Feedback: "The professor explained concepts clearly, and the course materials were very helpful."
Category: Positive

Feedback: "I really enjoyed this class—assignments were engaging and relevant to real-world scenarios."
Category: Positive

Feedback: "The teaching staff was supportive and always available to answer questions."
Category: Positive

Feedback: "The lectures felt disorganized, and it was hard to understand what was expected."
Category: Negative

Feedback: "Grading took too long, and feedback on assignments was minimal."
Category: Negative

Feedback: "The course workload was overwhelming compared to the credit hours."
Category: Negative

Feedback: "The course covered the topics listed in the syllabus."
Category: Neutral

Feedback: "Lectures were held twice a week, and attendance was mandatory."
Category: Neutral

Feedback: "The class used a mix of exams, quizzes, and assignments for evaluation."
Category: Neutral

```

## 2. One-Shot Prompting:

### Prompt:

"""\nWrite a python code to categorize the feedback of university students into positive, negative, and neutral

The professor explained concepts clearly, and the course materials were very helpful.-Positive I really enjoyed this class—assignments were engaging and relevant to real-world scenarios.

The teaching staff was supportive and always available to answer questions.

The lectures felt disorganized, and it was hard to understand what was expected.

Grading took too long, and feedback on assignments was minimal.

The course workload was overwhelming compared to the credit hours.

The course covered the topics listed in the syllabus.

Lectures were held twice a week, and attendance was mandatory. The class used a mix of exams, quizzes, and assignments for evaluation. """

### Given Code:

```
import re
def categorize_feedback(feedback):
    positive_keywords = [
        "clearly", "helpful", "enjoyed", "engaging", "relevant",
        "supportive", "available", "organized", "understand"
    ]
    negative_keywords = [
        "disorganized", "hard", "took too long", "minimal",
        "overwhelming", "difficult", "confusing", "boring"
    ]

    feedback_lower = feedback.lower()

    if any(re.search(r'\b' + re.escape(word) + r'\b', feedback_lower) for word in positive_keywords):
        return "Positive"
    elif any(re.search(r'\b' + re.escape(word) + r'\b', feedback_lower) for word in negative_keywords):
        return "Negative"
    else:
        return "Neutral"

# Example usage
feedback_list = [
    "The professor explained concepts clearly, and the course materials were very helpful.",
    "I really enjoyed this class-assignments were engaging and relevant to real-world scenarios.",
    "The teaching staff was supportive and always available to answer questions.",
    "The lectures felt disorganized, and it was hard to understand what was expected.",
    "Grading took too long, and feedback on assignments was minimal.",
    "The course workload was overwhelming compared to the credit hours.",
    "The course covered the topics listed in the syllabus.",
    "Lectures were held twice a week, and attendance was mandatory.",
    "The class used a mix of exams, quizzes, and assignments for evaluation."
]
for feedback in feedback_list:
    category = categorize_feedback(feedback)
    print(f"Feedback: {feedback}\nCategory: {category}\n")
```

### Output:

```
Query: I'm new to programming and want to know which course to start with.
Classified as: Beginner

Query: Can you recommend an introductory course in data science.
Classified as: Beginner

Query: I have no experience with Python and need a beginner-friendly course.
Classified as: Beginner

Query: I understand basic Python and want to improve my skills further.
Classified as: Intermediate

Query: Which course should I take after learning introductory statistics.
Classified as: Intermediate

Query: I have built simple web applications and want to learn backend development.
Classified as: Intermediate

Query: Are there advanced courses on deep learning model optimization.
Classified as: Advanced

Query: I want to specialize in distributed systems at an expert level.
Classified as: Advanced

Query: Which course focuses on research-level natural language processing techniques.
Classified as: Advanced
```

## Course Recommendation System

An online learning platform wants to recommend courses by classifying learner queries into Beginner, Intermediate, or Advanced

levels.

### 1. Zero-Shot Prompting:

#### **Prompt:**

"""Write a python code to classify the learner queries into beginner, intermediate, or advanced levels

I'm new to programming and want to know which course to start with.

Can you recommend an introductory course in data science.

I have no experience with Python and need a beginner-friendly course.

I understand basic Python and want to improve my skills further.

Which course should I take after learning introductory statistics.

I have built simple web applications and want to learn backend development.

Are there advanced courses on deep learning model optimization.

I want to specialize in distributed systems at an expert level.

Which course focuses on research-level natural language processing techniques. """

#### **Given Code:**

```
def classify_query(query):
    beginner_keywords = ["new to programming", "introductory course", "no experience", "beginner-friendly"]
    intermediate_keywords = ["understand basic", "improve my skills", "after learning", "built simple"]
    advanced_keywords = ["advanced courses", "expert level", "research-level", "specialize"]

    query_lower = query.lower()

    if any(keyword in query_lower for keyword in beginner_keywords):
        return "Beginner"
    elif any(keyword in query_lower for keyword in intermediate_keywords):
        return "Intermediate"
    elif any(keyword in query_lower for keyword in advanced_keywords):
        return "Advanced"
    else:
        return "Unclassified"

# Example queries
queries = [
    "I'm new to programming and want to know which course to start with.",
    "Can you recommend an introductory course in data science.",
    "I have no experience with Python and need a beginner-friendly course.",
    "I understand basic Python and want to improve my skills further.",
    "Which course should I take after learning introductory statistics.",
    "I have built simple web applications and want to learn backend development.",
    "Are there advanced courses on deep learning model optimization.",
    "I want to specialize in distributed systems at an expert level.",
    "Which course focuses on research-level natural language processing techniques."
]

for query in queries:
    level = classify_query(query)
    print(f"Query: {query}\nClassified as: {level}\n")
```

---

#### **Output:**

```
Query: I'm new to programming and want to know which course to start with.  
Classified as: Beginner  
  
Query: Can you recommend an introductory course in data science.  
Classified as: Beginner  
  
Query: I have no experience with Python and need a beginner-friendly course.  
Classified as: Beginner  
  
Query: I understand basic Python and want to improve my skills further.  
Classified as: Intermediate  
  
Query: Which course should I take after learning introductory statistics.  
Classified as: Intermediate  
  
Query: I have built simple web applications and want to learn backend development.  
Classified as: Intermediate  
  
Query: Are there advanced courses on deep learning model optimization.  
Classified as: Advanced  
  
Query: I want to specialize in distributed systems at an expert level.  
Classified as: Advanced  
  
Query: Which course focuses on research-level natural language processing techniques.  
Classified as: Advanced
```

## **2. One-Shot Prompting:**

### **Prompt:**

"""Write a python code to classify the learner queries into beginner, intermediate, or advanced levels

I'm new to programming and want to know which course to start with.-> Beginner

Can you recommend an introductory course in data science.

I have no experience with Python and need a beginner-friendly course.

I understand basic Python and want to improve my skills further.

Which course should I take after learning introductory statistics.

I have built simple web applications and want to learn backend development.

Are there advanced courses on deep learning model optimization.

I want to specialize in distributed systems at an expert level.

Which course focuses on research-level natural language processing techniques.

"""

### **Given Code:**

```

def classify_query(query):
    beginner_keywords = ["new to programming", "introductory course", "no experience", "beginner-friendly"]
    intermediate_keywords = ["understand basic", "improve skills", "after learning", "built simple"]
    advanced_keywords = ["advanced courses", "specialize", "expert level", "research-level"]

    query_lower = query.lower()

    if any(keyword in query_lower for keyword in beginner_keywords):
        return "Beginner"
    elif any(keyword in query_lower for keyword in intermediate_keywords):
        return "Intermediate"
    elif any(keyword in query_lower for keyword in advanced_keywords):
        return "Advanced"
    else:
        return "Unclassified"

# Test cases
queries = [
    "I'm new to programming and want to know which course to start with.",
    "Can you recommend an introductory course in data science.",
    "I have no experience with Python and need a beginner-friendly course.",
    "I understand basic Python and want to improve my skills further.",
    "Which course should I take after learning introductory statistics.",
    "I have built simple web applications and want to learn backend development.",
    "Are there advanced courses on deep learning model optimization.",
    "I want to specialize in distributed systems at an expert level.",
    "Which course focuses on research-level natural language processing techniques."
]
for query in queries:
    level = classify_query(query)
    print(f"Query: {query}\nClassified as: {level}\n")

```

## Output:

```

Query: I'm new to programming and want to know which course to start with.
Classified as: Beginner

Query: Can you recommend an introductory course in data science.
Classified as: Beginner

Query: I have no experience with Python and need a beginner-friendly course.
Classified as: Beginner

Query: I understand basic Python and want to improve my skills further.
Classified as: Intermediate

Query: Which course should I take after learning introductory statistics.
Classified as: Intermediate

Query: I have built simple web applications and want to learn backend development.
Classified as: Intermediate

Query: Are there advanced courses on deep learning model optimization.
Classified as: Advanced

Query: I want to specialize in distributed systems at an expert level.
Classified as: Advanced

Query: Which course focuses on research-level natural language processing techniques.
Classified as: Advanced

```

## 3. Few-Shot Prompting:

### Prompt:

"""Write a python code to classify the learner queries into beginner, intermediate, or advanced levels

I'm new to programming and want to know which course to start with.-> Beginner

Can you recommend an introductory course in data science.

I have no experience with Python and need a beginner-friendly course.

I understand basic Python and want to improve my skills further.

Which course should I take after learning introductory statistics.-> Intermediate

I have built simple web applications and want to learn backend development.

Are there advanced courses on deep learning model optimization.

I want to specialize in distributed systems at an expert level.-> Advanced

Which course focuses on research-level natural language processing techniques. """ Given Code:

```
def classify_query(query):
    beginner_keywords = ["new to programming", "introductory course", "no experience", "beginner-friendly"]
    intermediate_keywords = ["understand basic", "improve skills", "after learning", "built simple"]
    advanced_keywords = ["advanced courses", "specialize", "expert level", "research-level"]

    query_lower = query.lower()

    if any(keyword in query_lower for keyword in beginner_keywords):
        return "Beginner"
    elif any(keyword in query_lower for keyword in intermediate_keywords):
        return "Intermediate"
    elif any(keyword in query_lower for keyword in advanced_keywords):
        return "Advanced"
    else:
        return "Unclassified"

# Example usage
queries = []
    "I'm new to programming and want to know which course to start with.",
    "Can you recommend an introductory course in data science.",
    "I have no experience with Python and need a beginner-friendly course.",
    "I understand basic Python and want to improve my skills further.",
    "Which course should I take after learning introductory statistics.",
    "I have built simple web applications and want to learn backend development.",
    "Are there advanced courses on deep learning model optimization.",
    "I want to specialize in distributed systems at an expert level",
    "Which course focuses on research-level natural language processing techniques."
]

for query in queries:
    level = classify_query(query)
    print(f"Query: {query}\nClassified Level: {level}\n")
```

### Output:

```
Query: I'm new to programming and want to know which course to start with.
Classified as: Beginner

Query: Can you recommend an introductory course in data science.
Classified as: Beginner

Query: I have no experience with Python and need a beginner-friendly course.
Classified as: Beginner

Query: I understand basic Python and want to improve my skills further.
Classified as: Intermediate

Query: Which course should I take after learning introductory statistics.
Classified as: Intermediate

Query: I have built simple web applications and want to learn backend development.
Classified as: Intermediate

Query: Are there advanced courses on deep learning model optimization.
Classified as: Advanced

Query: I want to specialize in distributed systems at an expert level.
Classified as: Advanced

Query: Which course focuses on research-level natural language processing techniques.
Classified as: Advanced
```