Lab 4: Advanced Prompt Engineering: Zero-shot, one-shot, and few shot techniques

COURSE: AI Assisted Coding

NAME: G. SHRAVANI

BATCH-34

HALLTICKET.NO : 2303A52361

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.

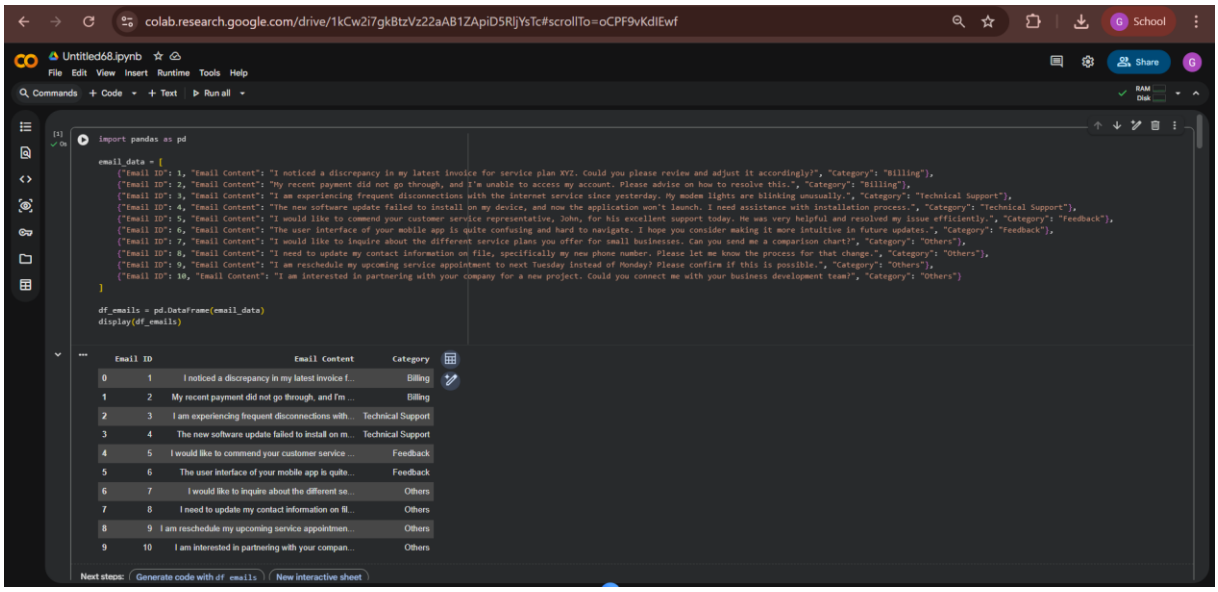
d. Few-shot Prompting:

• 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.



2. Travel Query Classification

Scenario:

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

Tasks:

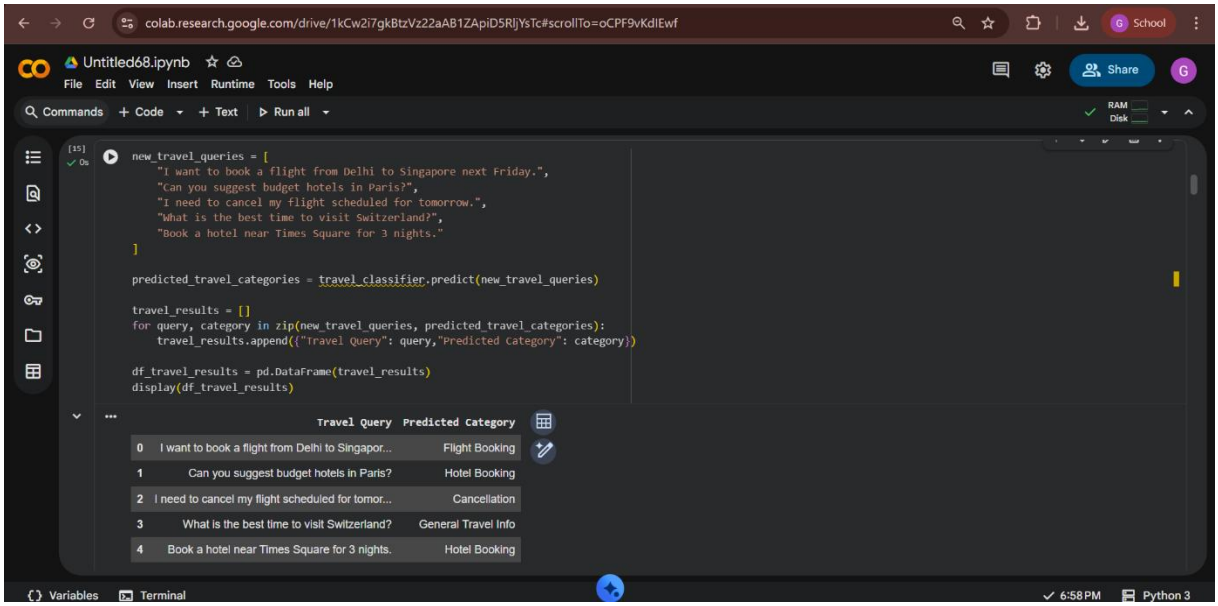
a. Prepare labeled travel queries.

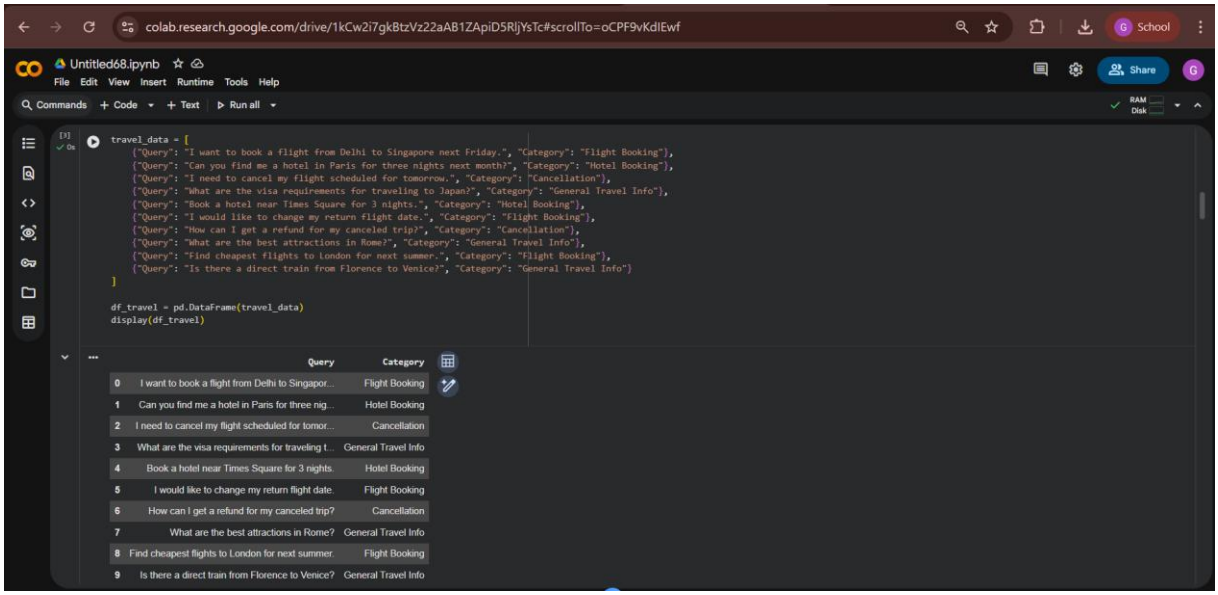
b. Apply Zero-shot prompting.

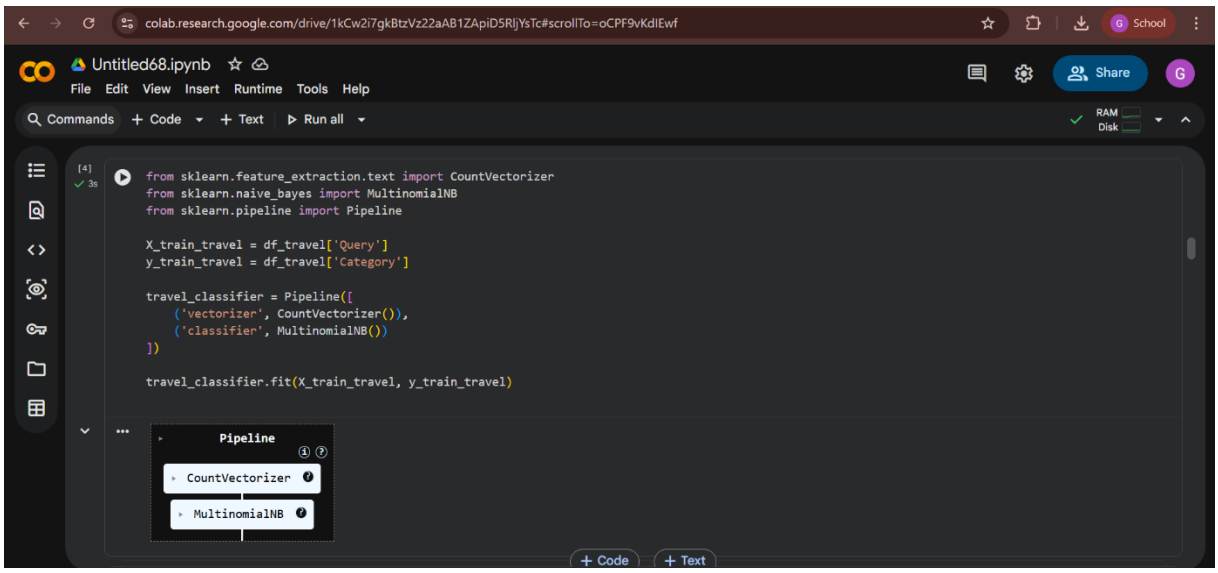
c. Apply One-shot prompting.

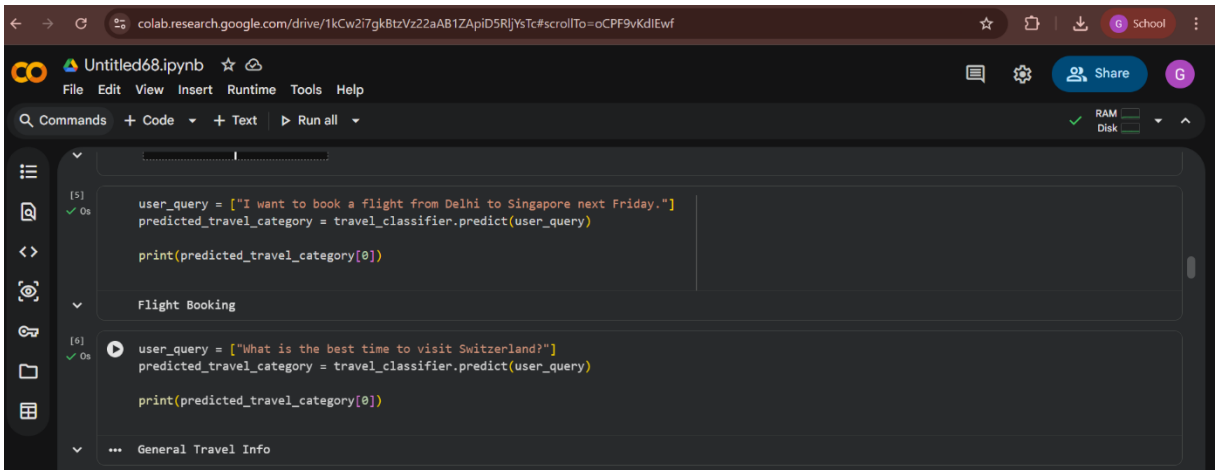
d. Apply Few-shot prompting.

e. Compare response consistency.









3. Programming Question Type Identification

Scenario:

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

Tasks:

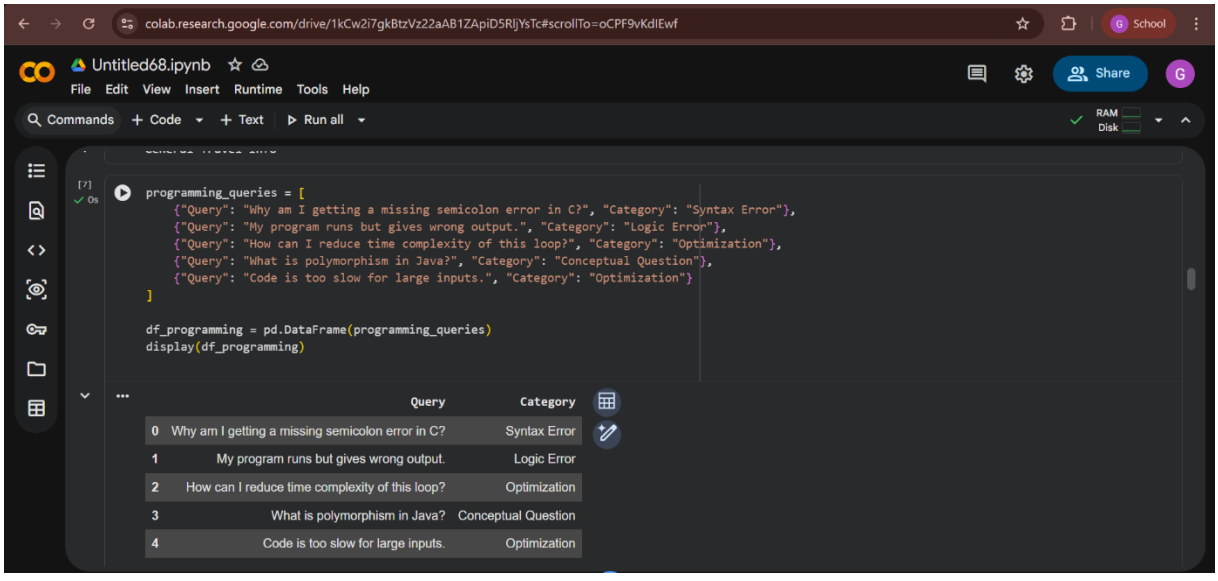
a. Prepare coding-related user queries.

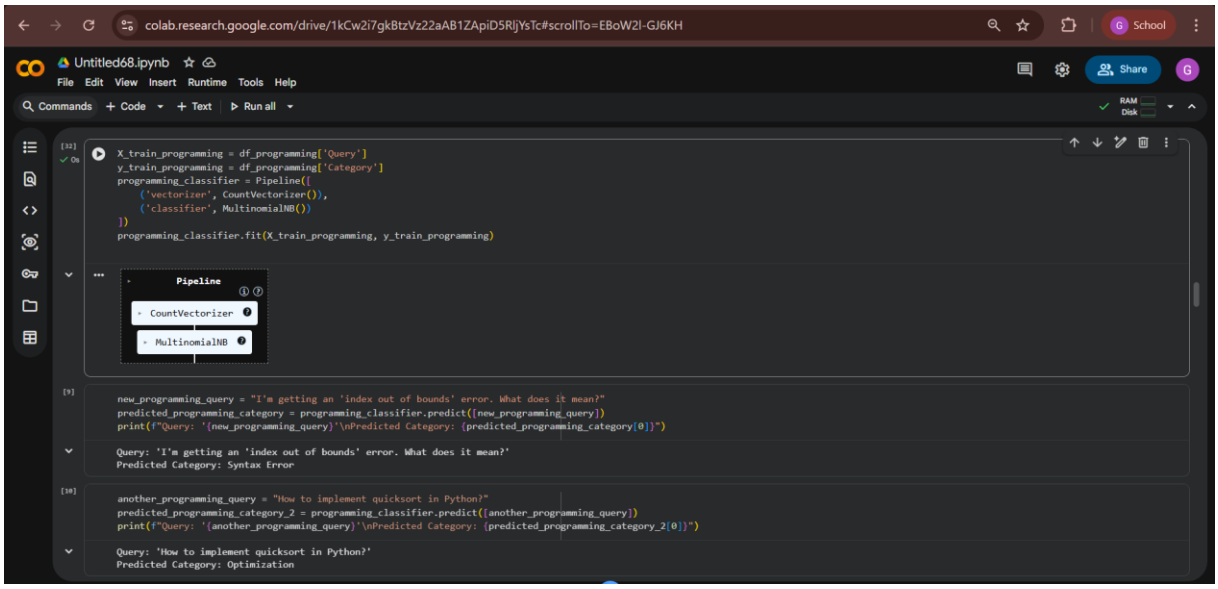
b. Perform Zero-shot classification.

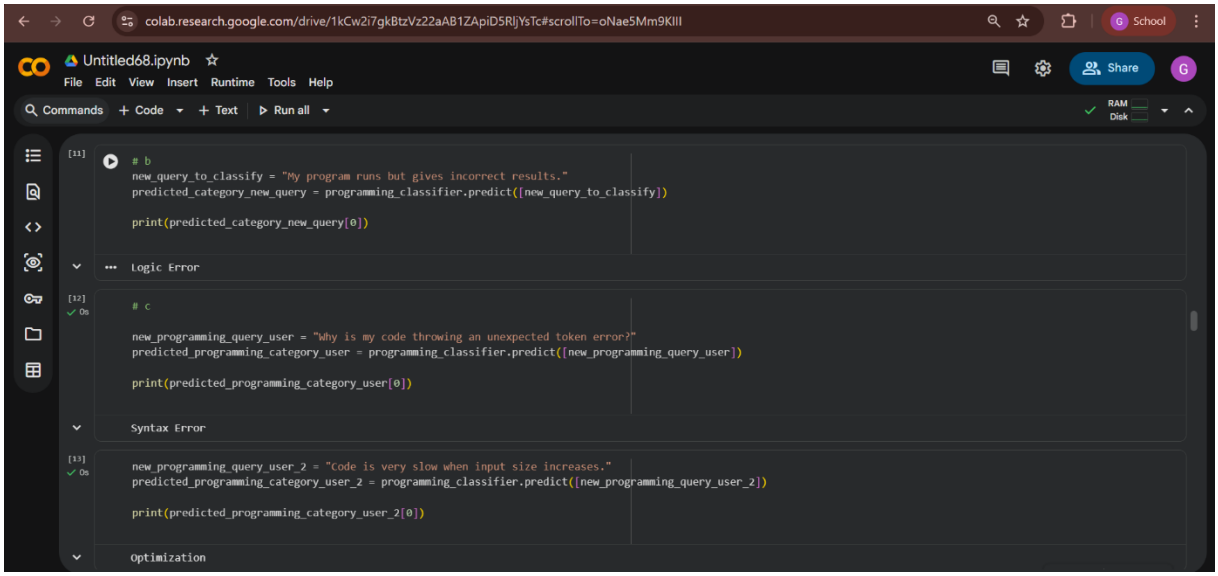
c. Perform One-shot classification.

d. Perform Few-shot classification.

e. Analyze improvements in technical accuracy







4.Social Media Post Categorization

Scenario:

A social media analytics tool must classify posts into Promotion,

Complaint, Appreciation, or Inquiry.

Tasks:

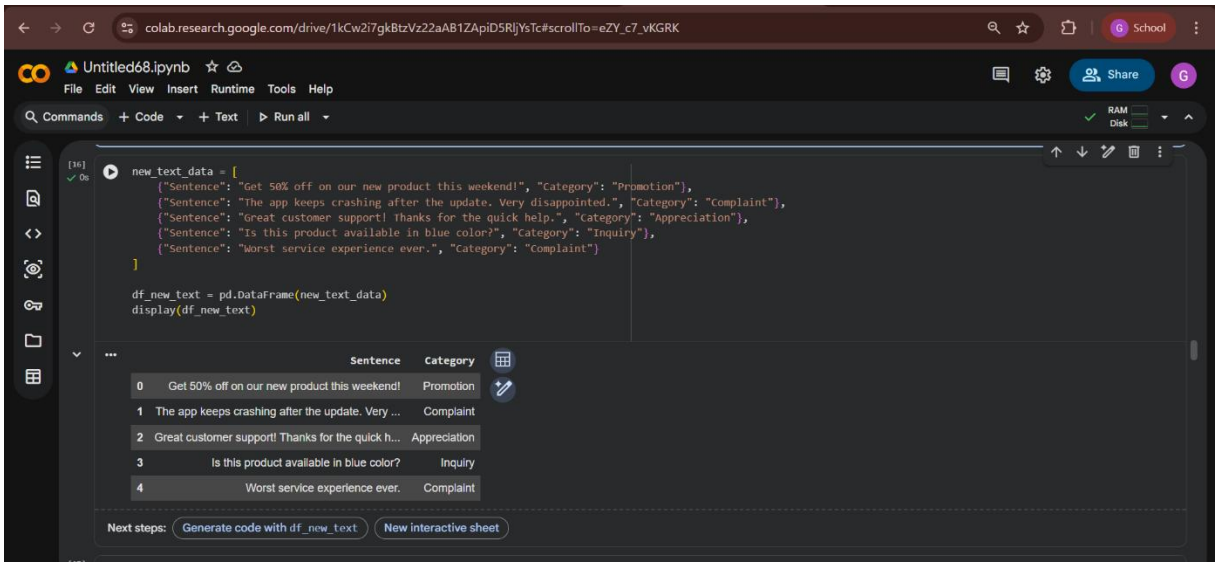
1. Prepare sample social media posts.

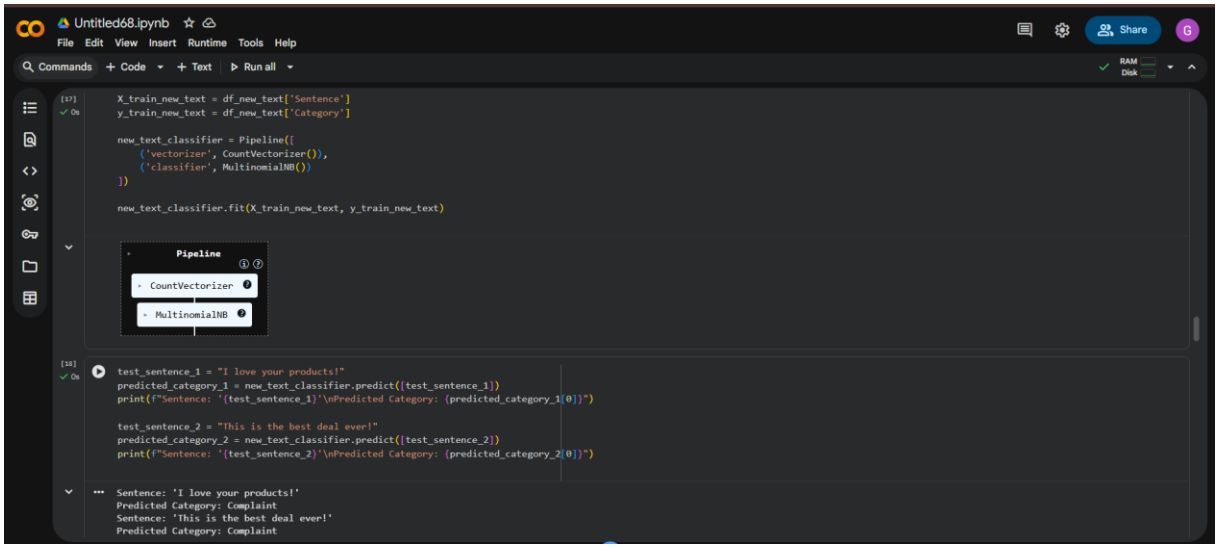
2. Use Zero-shot prompting.

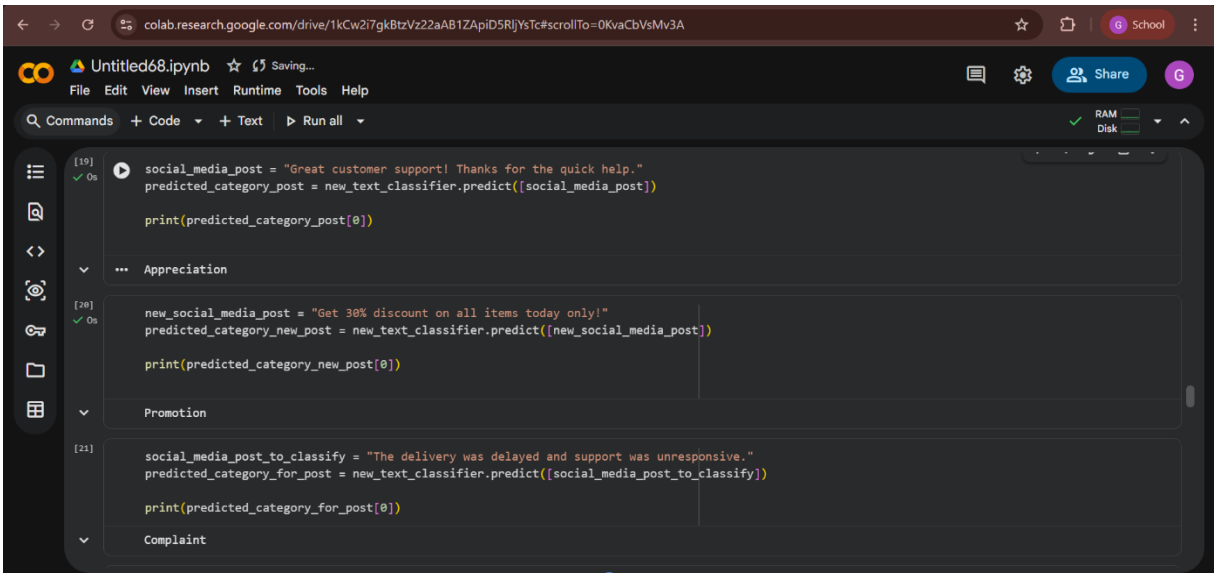
3. Use One-shot prompting.

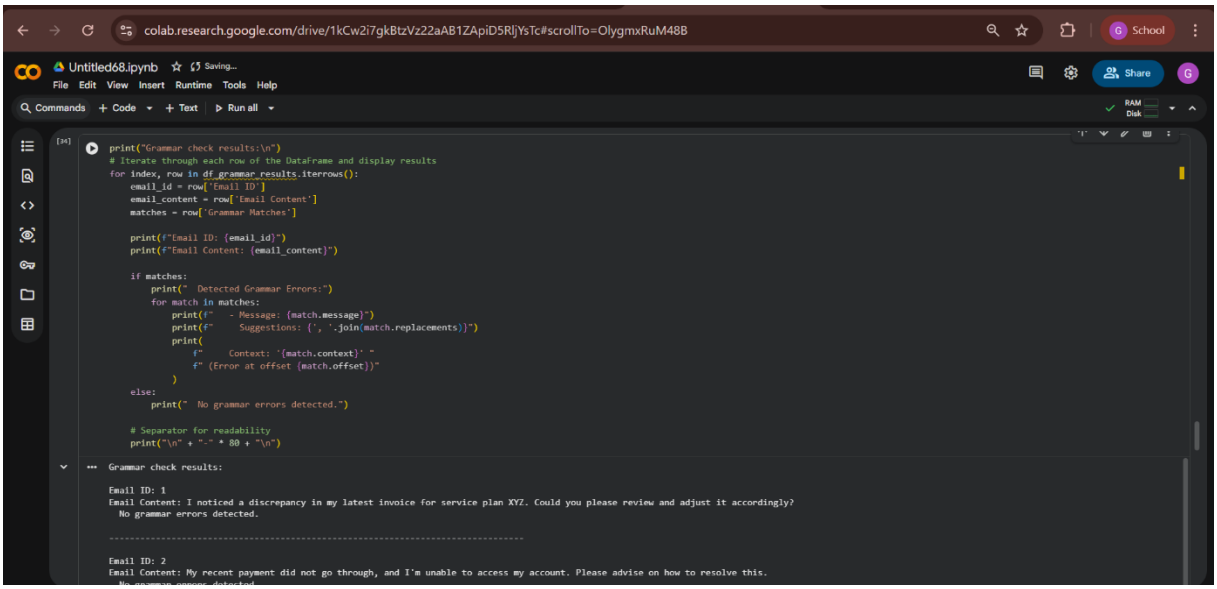
4. Use Few-shot prompting.

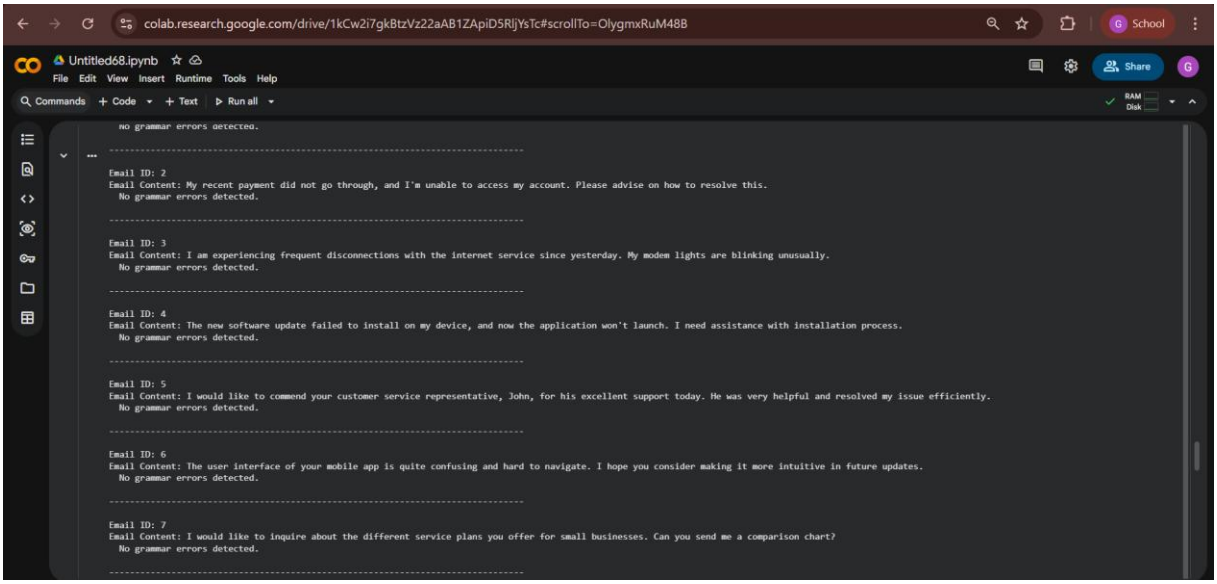
5. Analyze informal language handling

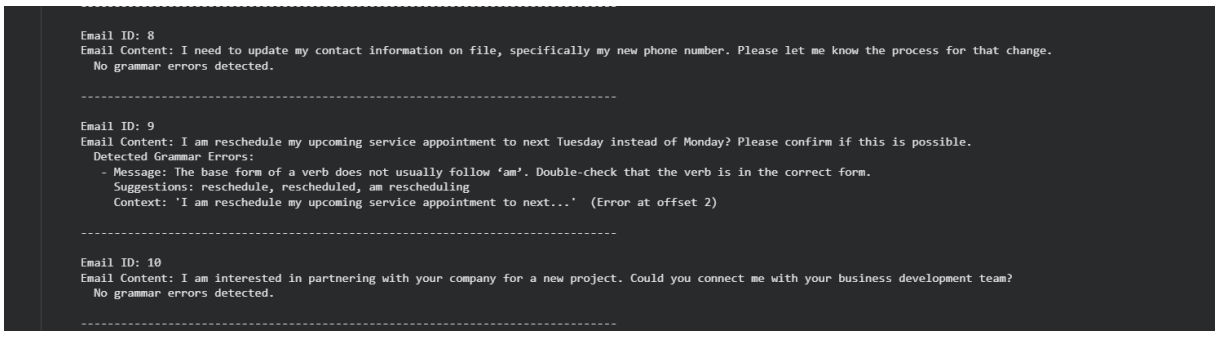

colab.research.google.com/drive/1kCw2i7gkBtzVz22aAB1ZApiD5RljYsTc#scrollTo=eZY_c7_vKGRK
Untitled68.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
Share
RAM
Disk
[16]
0s
new_text_data = [
{"Sentence": "Get 50% off on our new product this weekend!", "Category": "Promotion"},
{"Sentence": "The app keeps crashing after the update. Very disappointed.", "Category": "Complaint"},
{"Sentence": "Great customer support! Thanks for the quick help.", "Category": "Appreciation"},
{"Sentence": "Is this product available in blue color?", "Category": "Inquiry"},
{"Sentence": "Worst service experience ever.", "Category": "Complaint"}
]
df_new_text = pd.DataFrame(new_text_data)
display(df_new_text)
Sentence Category
0 Get 50% off on our new product this weekend! Promotion
1 The app keeps crashing after the update. Very ... Complaint
2 Great customer support! Thanks for the quick h... Appreciation
3 Is this product available in blue color? Inquiry
4 Worst service experience ever. Complaint
Next steps: Generate code with df_new_text New interactive sheet

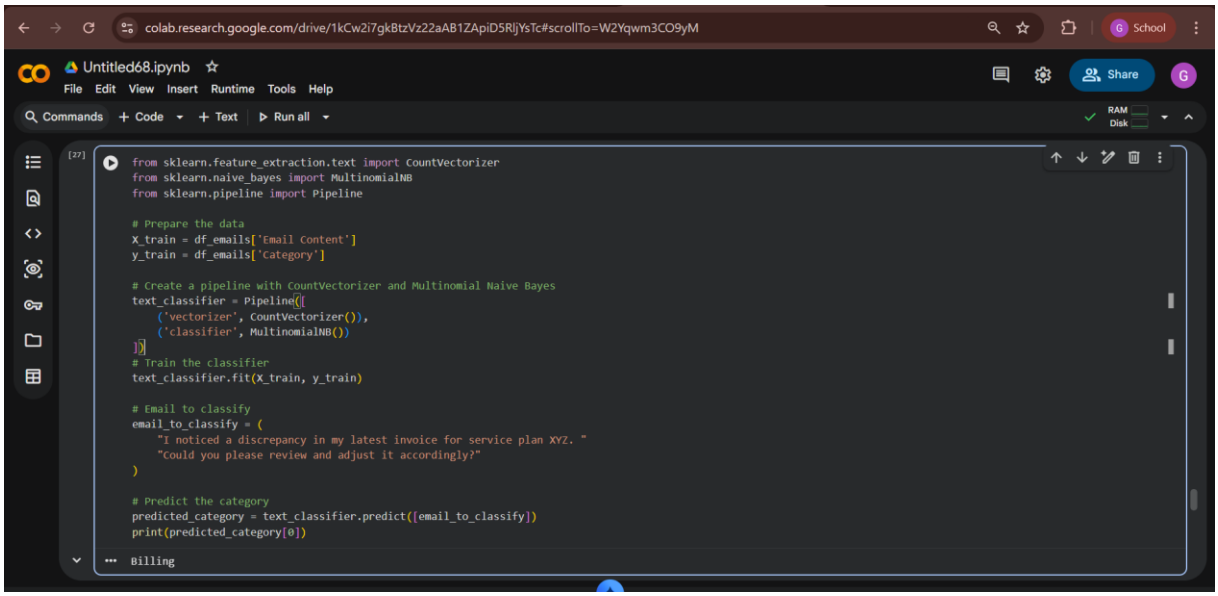

Untitled68.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
Share
RAM
Disk
[17]
0s
X_train_new_text = df_new_text['Sentence']
y_train_new_text = df_new_text['Category']
new_text_classifier = Pipeline([
('vectorizer', CountVectorizer()),
('classifier', MultinomialNB())
])
new_text_classifier.fit(X_train_new_text, y_train_new_text)
Pipeline
CountVectorizer
MultinomialNB
[18]
0s
test_sentence_1 = "I love your products!"
predicted_category_1 = new_text_classifier.predict([test_sentence_1])
print(f"Sentence: '{test_sentence_1}'\nPredicted Category: {predicted_category_1[0]}")
test_sentence_2 = "This is the best deal ever!"
predicted_category_2 = new_text_classifier.predict([test_sentence_2])
print(f"Sentence: '{test_sentence_2}'\nPredicted Category: {predicted_category_2[0]}")
Sentence: 'I love your products!'
Predicted Category: Complaint
Sentence: 'This is the best deal ever!'
Predicted Category: Complaint

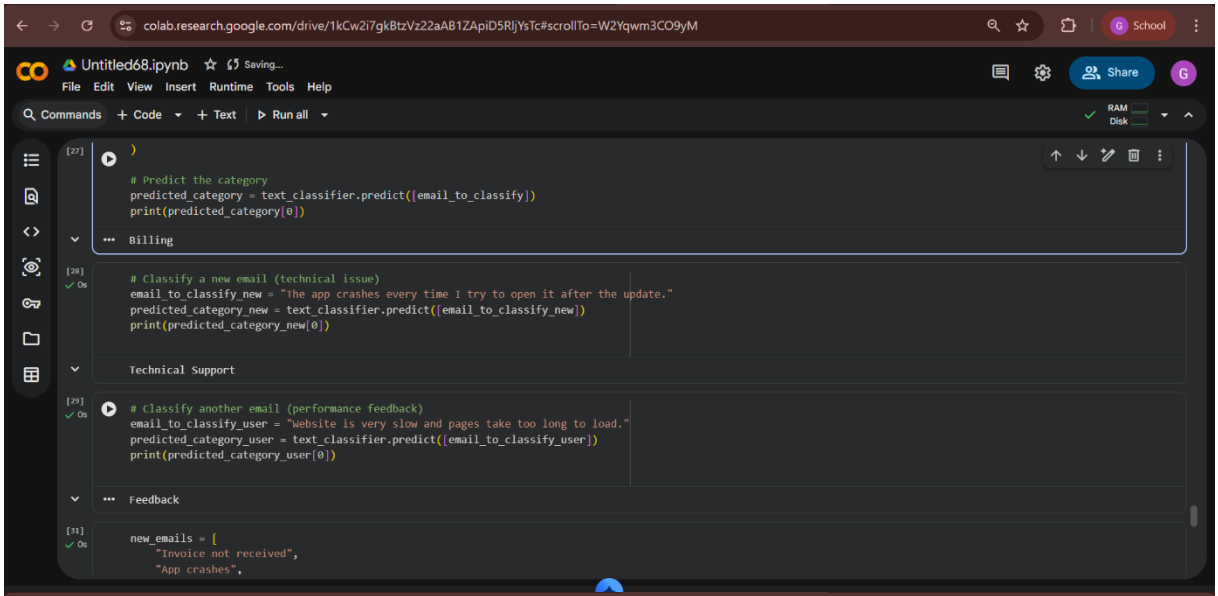

colab.research.google.com/drive/1kCw2i7gkBtzVz22aAB1ZApiD5RljYsTc#scrollTo=0KvaCbVsMv3A
Untitled68.ipynb
Saving...
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
Share
RAM
Disk
[19]
0s
social_media_post = "Great customer support! Thanks for the quick help."
predicted_category_post = new_text_classifier.predict([social_media_post])
print(predicted_category_post[0])
Appreciation
[20]
0s
new_social_media_post = "Get 30% discount on all items today only!"
predicted_category_new_post = new_text_classifier.predict([new_social_media_post])
print(predicted_category_new_post[0])
Promotion
[21]
social_media_post_to_classify = "The delivery was delayed and support was unresponsive."
predicted_category_for_post = new_text_classifier.predict([social_media_post_to_classify])
print(predicted_category_for_post[0])
Complaint

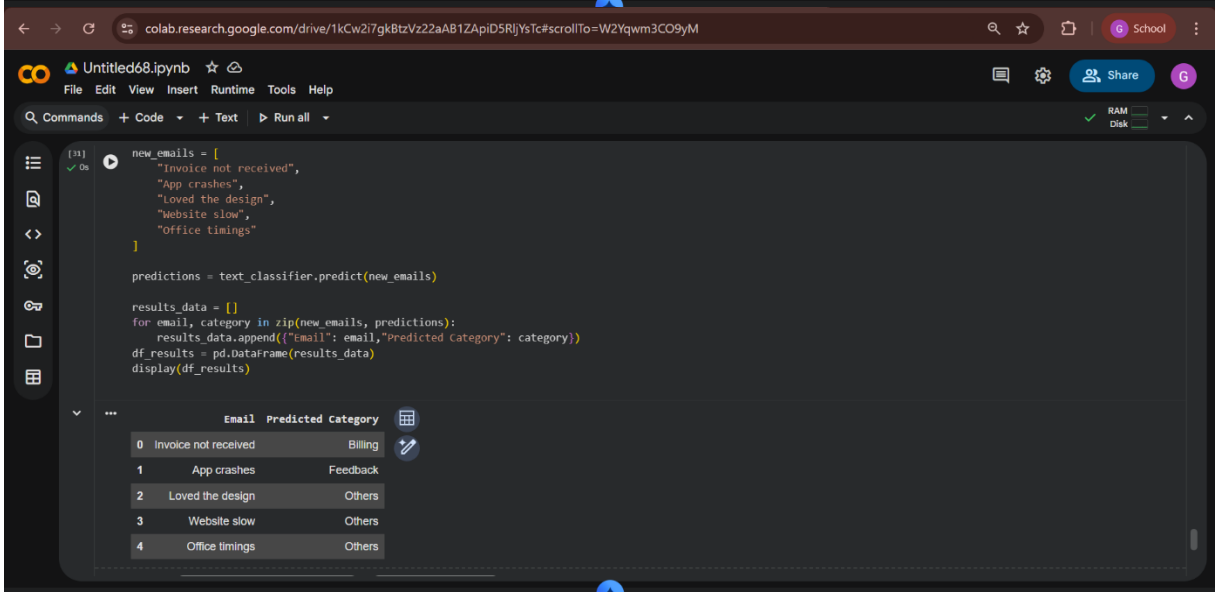

colab.research.google.com/drive/1kCw2i7gkBtzVz22aAB1ZApiD5RljYsTc#scrollTo=OlygmxRuM48B
Untitled68.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
print("Grammar check results:\n")
# Iterate through each row of the DataFrame and display results
for index, row in df_grammar_results.iterrows():
email_id = row['Email ID']
email_content = row['Email Content']
matches = row['Grammar Matches']
print(f"Email ID: {email_id}")
print(f"Email Content: {email_content}")
if matches:
print(" Detected Grammar Errors:")
for match in matches:
print(f" - Message: {match.message}")
print(f" Suggestions: {', '.join(match.replacements)}")
print(
f" Context: '{match.context}' "
f"(Error at offset {match.offset})"
)
else:
print(" No grammar errors detected.")
# Separator for readability
print("\n" + "-" * 80 + "\n")
Grammar check results:
Email ID: 1
Email Content: I noticed a discrepancy in my latest invoice for service plan XYZ. Could you please review and adjust it accordingly?
No grammar errors detected.
Email ID: 2
Email Content: My recent payment did not go through, and I'm unable to access my account. Please advise on how to resolve this.


colab.research.google.com/drive/1kCw2i7gkBtzVz22aAB1ZApiD5RljYsTc#scrollTo=OlygmxRuM48B
Untitled68.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
No grammar errors detected.
Email ID: 2
Email Content: My recent payment did not go through, and I'm unable to access my account. Please advise on how to resolve this.
No grammar errors detected.
Email ID: 3
Email Content: I am experiencing frequent disconnections with the internet service since yesterday. My modem lights are blinking unusually.
No grammar errors detected.
Email ID: 4
Email Content: The new software update failed to install on my device, and now the application won't launch. I need assistance with installation process.
No grammar errors detected.
Email ID: 5
Email Content: I would like to commend your customer service representative, John, for his excellent support today. He was very helpful and resolved my issue efficiently.
No grammar errors detected.
Email ID: 6
Email Content: The user interface of your mobile app is quite confusing and hard to navigate. I hope you consider making it more intuitive in future updates.
No grammar errors detected.
Email ID: 7
Email Content: I would like to inquire about the different service plans you offer for small businesses. Can you send me a comparison chart?
No grammar errors detected.


Email ID: 8
Email Content: I need to update my contact information on file, specifically my new phone number. Please let me know the process for that change.
No grammar errors detected.
Email ID: 9
Email Content: I am reschedule my upcoming service appointment to next Tuesday instead of Monday? Please confirm if this is possible.
Detected Grammar Errors:
- Message: The base form of a verb does not usually follow 'am'. Double-check that the verb is in the correct form.
Suggestions: reschedule, rescheduled, am rescheduling
Context: 'I am reschedule my upcoming service appointment to next...' (Error at offset 2)
Email ID: 10
Email Content: I am interested in partnering with your company for a new project. Could you connect me with your business development team?
No grammar errors detected.







	Email	Predicted Category
0	Invoice not received	Billing
1	App crashes	Feedback
2	Loved the design	Others
3	Website slow	Others
4	Office timings	Others