

Zero-shot vs Few-shot Prompting

1. Objective

The goal of this project is to understand the difference between **zero-shot** and **few-shot prompting** using large language models (LLMs). Specifically, we focus on **sentiment analysis** of sentences.

2. Instructions & Implementation

Step 1: Select a Task

The task chosen is **sentiment analysis**, i.e., classifying a sentence as **positive** or **negative**.

Step 2: Zero-shot Prompting

Definition: In zero-shot prompting, the model is given the task **directly without examples**.

Prompt Example:

```
csharp
```

```
Determine if the following sentence is positive or negative:  
"I love my new phone."
```

Implementation (Hugging Face Model):

- Model: valhalla/distilbart-mnli-12-1
- Approach: zero-shot-classification
- Output Example:

```
makefile
```

```
Prediction: Positive  
Confidence: 0.98
```

Step 3: Few-shot Prompting

Definition: In few-shot prompting, the model is given **2–3 examples before the query** to guide it.

Prompt Example:

vbnet

Decide whether the sentiment of the following sentence is Positive or Negative.
Give short justification.

Example 1:

Sentence: I love this movie, it was amazing!

Answer: Positive - expresses enjoyment.

Example 2:

Sentence: This food tastes terrible and I hate it.

Answer: Negative - expresses dislike.

Sentence: The movie was amazing.

Answer:

Step 4: Compare Outputs

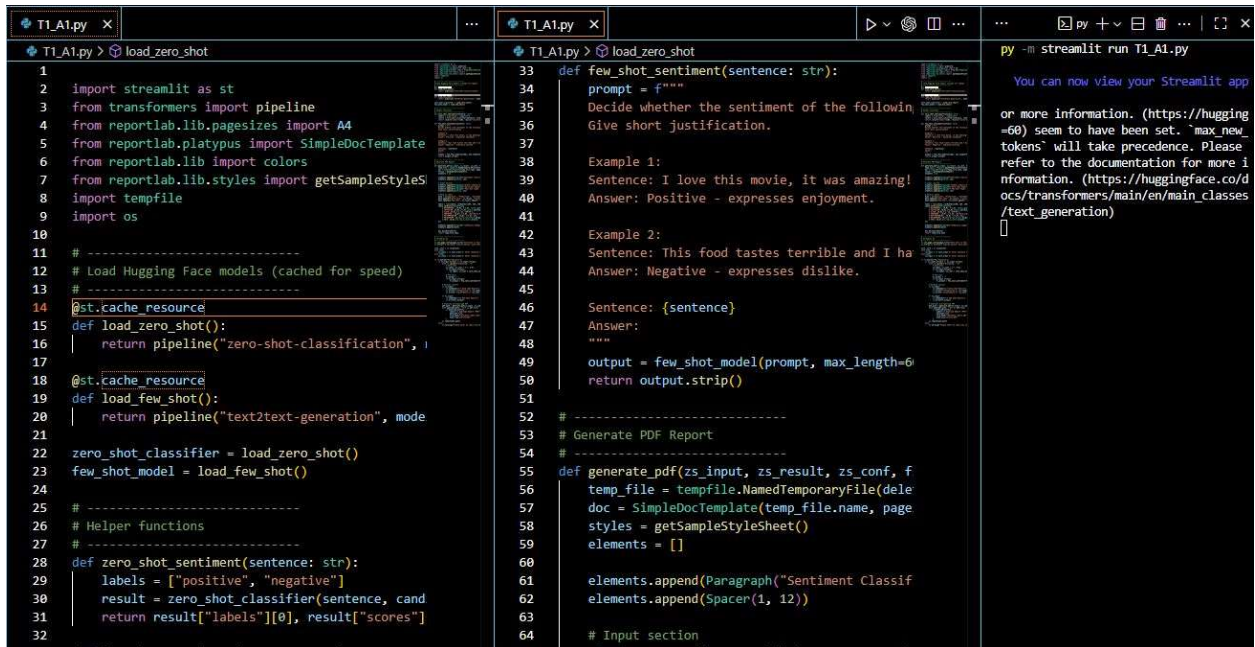
Approach	Input Sentence	Prediction	Confidence / Reasoning
Zero-shot	I love my new phone.	Positive	0.98
Few-shot	I love my new phone.	Positive	Positive - expresses enjoyment

3. Observations

- Both zero-shot and few-shot prompting correctly classified the sentiment.
- Few-shot prompting provides a **short reasoning**, which improves interpretability.
- Zero-shot is faster and simpler but may lack detailed explanation.
- Few-shot is more reliable for subtle or ambiguous inputs because it **guides the model with examples**.

4. Deliverables

- Streamlit App: Interactive interface to enter sentences and run both approaches.



```
1 import streamlit as st
2 from transformers import pipeline
3 from reportlab.lib.pagesizes import A4
4 from reportlab.platypus import SimpleDocTemplate
5 from reportlab.lib import colors
6 from reportlab.lib.styles import getSampleStyleSheet
7 import tempfile
8 import os
9
10 # -----
11 # Load Hugging Face models (cached for speed)
12 # -----
13 @st.cache_resource
14 def load_zero_shot():
15     return pipeline("zero-shot-classification",
16
17
18 @st.cache_resource
19 def load_few_shot():
20     return pipeline("text2text-generation", mode
21
22 zero_shot_classifier = load_zero_shot()
23 few_shot_model = load_few_shot()
24
25 # -----
26 # Helper functions
27 # -----
28 def zero_shot_sentiment(sentence: str):
29     labels = ["positive", "negative"]
30     result = zero_shot_classifier(sentence, cand
31     return result["labels"][0], result["scores"]
32
33 def few_shot_sentiment(sentence: str):
34     prompt = f"""
35     Decide whether the sentiment of the followin
36     Give short justification.
37
38     Example 1:
39     Sentence: I love this movie, it was amazing!
40     Answer: Positive - expresses enjoyment.
41
42     Example 2:
43     Sentence: This food tastes terrible and I ha
44     Answer: Negative - expresses dislike.
45
46     Sentence: {sentence}
47     Answer:
48     """
49     output = few_shot_model(prompt, max_length=6
50     return output.strip()
51
52 # -----
53 # Generate PDF Report
54 # -----
55 def generate_pdf(zs_input, zs_result, zs_conf, f
56     temp_file = tempfile.NamedTemporaryFile(dele
57     doc = SimpleDocTemplate(temp_file.name, page
58     styles = getSampleStyleSheet()
59     elements = []
60
61     elements.append(Paragraph("Sentiment Classif
62     elements.append(Spacer(1, 12))
63
64 # Input section
```

Fig1: Input Snapshot

- PDF Report: Automatically generated with:
 - User inputs
 - Zero-shot and few-shot results
 - Comparison table

Observations: Differences in reasoning and confidence between zero-shot and few-shot prompting.

5. Output

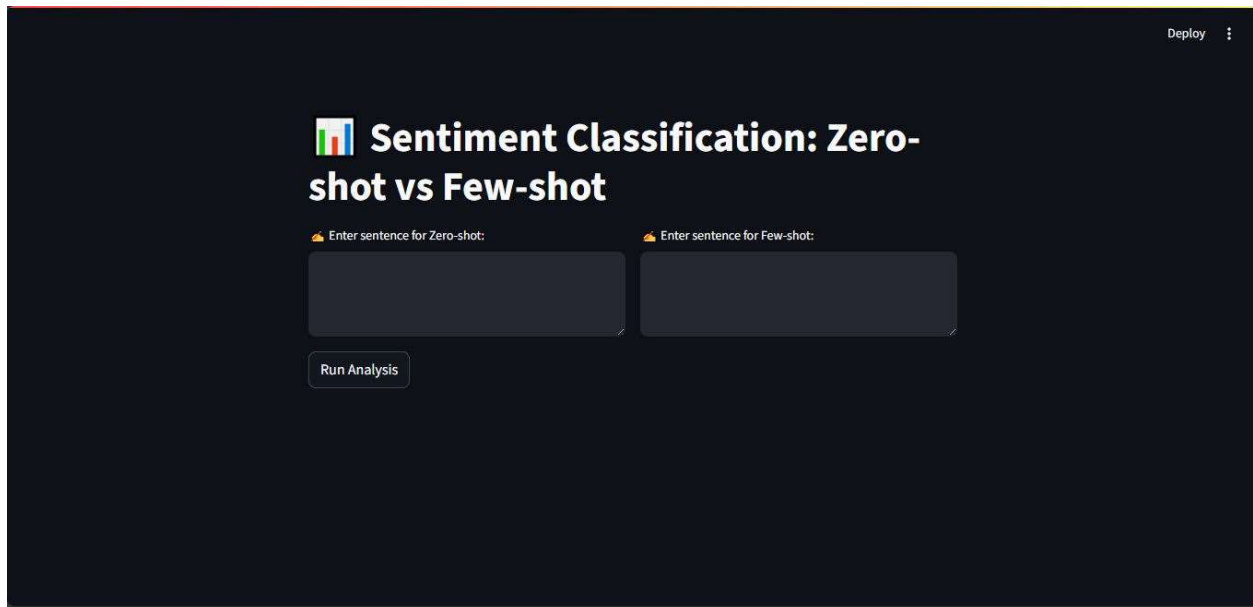


Fig2: Output Snapshot1



Fig3: Output Snapshot2