

```
pip install -U datasets fsspec
```

```
Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-packages (from datasets) (3.18.0)
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.11/dist-packages (from datasets) (2.0.2)
Requirement already satisfied: pyarrow>=15.0.0 in /usr/local/lib/python3.11/dist-packages (from datasets) (18.1.0)
Requirement already satisfied: dill<0.3.9,>=0.3.0 in /usr/local/lib/python3.11/dist-packages (from datasets) (0.3.7)
Requirement already satisfied: pandas in /usr/local/lib/python3.11/dist-packages (from datasets) (2.2.2)
Requirement already satisfied: requests>=2.32.2 in /usr/local/lib/python3.11/dist-packages (from datasets) (2.32.3)
Requirement already satisfied: tqdm>=4.66.3 in /usr/local/lib/python3.11/dist-packages (from datasets) (4.67.1)
Requirement already satisfied: xxhash in /usr/local/lib/python3.11/dist-packages (from datasets) (3.5.0)
Requirement already satisfied: multiprocessing<0.70.17 in /usr/local/lib/python3.11/dist-packages (from datasets) (0.70.15)
Downloading fsspec-2025.3.0-py3-none-any.whl.metadata (11 kB)
Requirement already satisfied: huggingface-hub>=0.24.0 in /usr/local/lib/python3.11/dist-packages (from datasets) (0.33.0)
Requirement already satisfied: packaging in /usr/local/lib/python3.11/dist-packages (from datasets) (24.2)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.11/dist-packages (from datasets) (6.0.2)
Requirement already satisfied: aiohttp!=4.0.0a0,!4.0.0a1 in /usr/local/lib/python3.11/dist-packages (from fsspec[http]<=2025.3.0)
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.24.0)
Requirement already satisfied: hf-xet<2.0.0,>=1.1.2 in /usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.24.0->datasets)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (from requests>=2.32.2->datasets)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from requests>=2.32.2->datasets) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests>=2.32.2->datasets) (2)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests>=2.32.2->datasets) (2)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packages (from pandas->datasets) (2.9.0)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas->datasets) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas->datasets) (2025.2)
Requirement already satisfied: aiohappyeyeballs>=2.3.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp!=4.0.0a0,!4.0.0a1->fsspec)
Requirement already satisfied: aiosignal>=1.1.2 in /usr/local/lib/python3.11/dist-packages (from aiohttp!=4.0.0a0,!4.0.0a1->fsspec)
Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp!=4.0.0a0,!4.0.0a1->fsspec)
Requirement already satisfied: frozenlist>=1.1.1 in /usr/local/lib/python3.11/dist-packages (from aiohttp!=4.0.0a0,!4.0.0a1->fsspec)
Requirement already satisfied: multidict<7.0,>=4.5 in /usr/local/lib/python3.11/dist-packages (from aiohttp!=4.0.0a0,!4.0.0a1->fsspec)
Requirement already satisfied: propcache>=0.2.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp!=4.0.0a0,!4.0.0a1->fsspec)
Requirement already satisfied: yarl<2.0,>=1.17.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp!=4.0.0a0,!4.0.0a1->fsspec)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.8.2->pandas->datasets)
Downloading datasets-3.6.0-py3-none-any.whl (491 kB)
491.5/491.5 kB 30.9 MB/s eta 0:00:00
Downloading fsspec-2025.3.0-py3-none-any.whl (193 kB)
193.6/193.6 kB 16.6 MB/s eta 0:00:00
Installing collected packages: fsspec, datasets
Attempting uninstall: fsspec
  Found existing installation: fsspec 2025.3.2
  Uninstalling fsspec-2025.3.2:
    Successfully uninstalled fsspec-2025.3.2
Attempting uninstall: datasets
  Found existing installation: datasets 2.14.4
  Uninstalling datasets-2.14.4:
    Successfully uninstalled datasets-2.14.4
ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behaviour is the
gcsfs 2025.3.2 requires fsspec==2025.3.2, but you have fsspec 2025.3.0 which is incompatible.
torch 2.6.0+cu124 requires nvidia-cublas-cu12==12.4.5.8; platform_system == "Linux" and platform_machine == "x86_64", but you hav
torch 2.6.0+cu124 requires nvidia-cuda-cupti-cu12==12.4.127; platform_system == "Linux" and platform_machine == "x86_64", but you
torch 2.6.0+cu124 requires nvidia-cuda-nvrtc-cu12==12.4.127; platform_system == "Linux" and platform_machine == "x86_64", but you
torch 2.6.0+cu124 requires nvidia-cuda-runtime-cu12==12.4.127; platform_system == "Linux" and platform_machine == "x86_64", but y
torch 2.6.0+cu124 requires nvidia-cudnn-cu12==9.1.0.70; platform_system == "Linux" and platform_machine == "x86_64", but you have
torch 2.6.0+cu124 requires nvidia-cufft-cu12==11.2.1.3; platform_system == "Linux" and platform_machine == "x86_64", but you have
torch 2.6.0+cu124 requires nvidia-curand-cu12==10.3.5.147; platform_system == "Linux" and platform_machine == "x86_64", but you h
torch 2.6.0+cu124 requires nvidia-cusolver-cu12==11.6.1.9; platform_system == "Linux" and platform_machine == "x86_64", but you h
torch 2.6.0+cu124 requires nvidia-cuspars-cu12==12.3.1.170; platform_system == "Linux" and platform_machine == "x86_64", but you
torch 2.6.0+cu124 requires nvidia-nvjitlink-cu12==12.4.127; platform_system == "Linux" and platform_machine == "x86_64", but you
Successfully installed datasets-3.6.0 fsspec-2025.3.0
```

```
from datasets import load_dataset

# Load the verified working dataset
dataset = load_dataset("frankdarkluo/dailydialog")

# Preview structure
print(dataset)
print(dataset["train"][0])
```

```

/usr/local/lib/python3.11/dist-packages/huggingface_hub/utils/_auth.py:94: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (https://huggingface.co/settings/tokens), set it as :
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.
warnings.warn(
README.md: 100% 31.0/31.0 [00:00<00:00, 1.29kB/s]

train.csv: 7.70M/? [00:00<00:00, 17.3MB/s]

valid.csv: 871k/? [00:00<00:00, 18.6MB/s]

test.csv: 898k/? [00:00<00:00, 14.6MB/s]

Generating train split: 60005/0 [00:00<00:00, 92825.70 examples/s]

Generating validation split: 6594/0 [00:00<00:00, 72151.27 examples/s]

Generating test split: 6955/0 [00:00<00:00, 90882.82 examples/s]

DatasetDict({
  train: Dataset({
    features: ['context', 'response'],
    num_rows: 60005
  })
  validation: Dataset({
    features: ['context', 'response'],
    num_rows: 6594
  })
  test: Dataset({
    features: ['context', 'response'],
    num_rows: 6955
  })
})
{'context': 'Waiter ! ', 'response': " I'll be with you in a second . Uh ... Yes , ma'am ? "}

```

```

# Auto-Label Responses with Sentiment
from transformers import AutoTokenizer, AutoModelForSequenceClassification
from transformers import pipeline

# Load a pre-trained Twitter sentiment model
model_name = "cardiffnlp/twitter-roberta-base-sentiment"
classifier = pipeline("sentiment-analysis", model=model_name, tokenizer=model_name)

# Annotate the first 5 responses
for i in range(5):
    text = dataset["train"][i]["response"]
    result = classifier(text)[0]
    print(f"Response: {text}")
    print(f"Predicted Sentiment: {result['label']} (score: {result['score']:.2f})")
    print("---")

```

```

config.json: 100% 747/747 [00:00<00:00, 50.0kB/s]

pytorch_model.bin: 100% 499M/499M [00:06<00:00, 111MB/s]

model.safetensors: 46% 230M/499M [00:01<00:00, 301MB/s]

vocab.json: 899k/? [00:00<00:00, 34.0MB/s]

merges.txt: 456k/? [00:00<00:00, 24.6MB/s]

special_tokens_map.json: 100% 150/150 [00:00<00:00, 3.40kB/s]

Device set to use cpu
Response: I'll be with you in a second . Uh ... Yes , ma'am ?
Predicted Sentiment: LABEL_1 (score: 0.78)
---
Response: This is not what I asked for . I'm afraid .
Predicted Sentiment: LABEL_0 (score: 0.92)
---
Response: Oh , I'm so sorry . May I ask what you ordered again ?
Predicted Sentiment: LABEL_0 (score: 0.72)
---
Response: Yes . What I ordered is roast beef , not roast beef sandwiches .
Predicted Sentiment: LABEL_1 (score: 0.68)
---
Response: Mind your own business .
Predicted Sentiment: LABEL_1 (score: 0.59)
---

```

```

label_map = {
    "LABEL_0": "Negative",
    "LABEL_1": "Neutral",

```

```

"LABEL_2": "Positive"
}

for i in range(5):
    text = dataset["train"][i]["response"]
    result = classifier(text)[0]
    sentiment = label_map[result['label']]
    print(f"Response: {text}")
    print(f"Predicted Sentiment: {sentiment} (score: {result['score']:.2f})")
    print("----")

```

```

➡ Response: I'll be with you in a second . Uh ... Yes , ma'am ?
Predicted Sentiment: Neutral (score: 0.78)
----
Response: This is not what I asked for . I'm afraid .
Predicted Sentiment: Negative (score: 0.92)
----
Response: Oh , I'm so sorry . May I ask what you ordered again ?
Predicted Sentiment: Negative (score: 0.72)
----
Response: Yes . What I ordered is roast beef , not roast beef sandwiches .
Predicted Sentiment: Neutral (score: 0.68)
----
Response: Mind your own business .
Predicted Sentiment: Neutral (score: 0.59)
----

```

```

# Pick a sample from the dataset
sample = dataset["train"][0]

# Extract context and response
context = sample["context"]
response = sample["response"]

# Combine them into one input string
combined_input = f"{context} [SEP] {response}"

print("Combined Input:", combined_input)

```

```

➡ Combined Input: Waiter ! [SEP] I'll be with you in a second . Uh ... Yes , ma'am ?

```

```

# Load sentiment analysis pipeline
model_name = "cardiffnlp/twitter-roberta-base-sentiment"
classifier = pipeline("sentiment-analysis", model=model_name, tokenizer=model_name)

```

```

# Define label mapping
label_map = {
    "LABEL_0": "Negative",
    "LABEL_1": "Neutral",
    "LABEL_2": "Positive"
}

```

```

# Test on first 5 samples using context + response
for i in range(5):
    sample = dataset["train"][i]
    context = sample["context"]
    response = sample["response"]

    combined_input = f"{context} [SEP] {response}"

    result = classifier(combined_input)[0]
    sentiment = label_map[result["label"]]

    print(f"Context: {context}")
    print(f"Response: {response}")
    print(f"Combined: {combined_input}")
    print(f"Predicted Sentiment (with context): {sentiment} (score: {result['score']:.2f})")
    print("----")

```

```

➡ Device set to use cpu
Context: Waiter !
Response: I'll be with you in a second . Uh ... Yes , ma'am ?
Combined: Waiter ! [SEP] I'll be with you in a second . Uh ... Yes , ma'am ?
Predicted Sentiment (with context): Neutral (score: 0.83)
----
Context: I'll be with you in a second . Uh ... Yes , ma'am ?
Response: This is not what I asked for . I'm afraid .
Combined: I'll be with you in a second . Uh ... Yes , ma'am ? [SEP] This is not what I asked for . I'm afraid .
Predicted Sentiment (with context): Negative (score: 0.82)
----

```

```

Context: This is not what I asked for . I'm afraid .
Response: Oh , I'm so sorry . May I ask what you ordered again ?
Combined: This is not what I asked for . I'm afraid . [SEP] Oh , I'm so sorry . May I ask what you ordered again ?
Predicted Sentiment (with context): Negative (score: 0.92)
---
Context: Oh , I'm so sorry . May I ask what you ordered again ?
Response: Yes . What I ordered is roast beef , not roast beef sandwiches .
Combined: Oh , I'm so sorry . May I ask what you ordered again ? [SEP] Yes . What I ordered is roast beef , not roast beef sandwi
Predicted Sentiment (with context): Negative (score: 0.63)
---
Context: Why are you always staring at Melissa ? Do you like her or something ?
Response: Mind your own business .
Combined: Why are you always staring at Melissa ? Do you like her or something ? [SEP] Mind your own business .
Predicted Sentiment (with context): Negative (score: 0.57)
---

```

```

# Compare performance – with vs without context
contextual_labels = []
non_contextual_labels = []

```

```

for i in range(100): # Use 100 samples for speed
    sample = dataset["train"][i]
    context = sample["context"]
    response = sample["response"]

    # Without context
    no_context_result = classifier(response)[0]
    no_context_sentiment = label_map[no_context_result["label"]]
    non_contextual_labels.append(no_context_sentiment)

    # With context
    combined_input = f"{context} [SEP] {response}"
    context_result = classifier(combined_input)[0]
    context_sentiment = label_map[context_result["label"]]
    contextual_labels.append(context_sentiment)

    if no_context_sentiment != context_sentiment:
        print(f"--- Sample {i} ---")
        print(f"Context: {context}")
        print(f"Response: {response}")
        print(f"Without Context: {no_context_sentiment}")
        print(f"With Context: {context_sentiment}")
        print()

```

```

--- Sample 3 ---
Context: Oh , I'm so sorry . May I ask what you ordered again ?
Response: Yes . What I ordered is roast beef , not roast beef sandwiches .
Without Context: Neutral
With Context: Negative

--- Sample 4 ---
Context: Why are you always staring at Melissa ? Do you like her or something ?
Response: Mind your own business .
Without Context: Neutral
With Context: Negative

--- Sample 6 ---
Context: A month . Don't forget to return them by the due date .
Response: What if I can't finish them by then ?
Without Context: Negative
With Context: Neutral

--- Sample 13 ---
Context: yes , we can look at the computer and computer menu at the same time . And this will help us understand both better .
Response: What should I learn first ?
Without Context: Neutral
With Context: Positive

--- Sample 17 ---
Context: Since summer is coming , I think swimming is a good way for you to do .
Response: Are you sure ?
Without Context: Neutral
With Context: Positive

--- Sample 19 ---
Context: Of course ! Swimming can help you stay in shape by targeting all parts of your body .
Response: Really ? Does swimming have other advantages ?
Without Context: Neutral
With Context: Positive

--- Sample 29 ---
Context: Exactly , it seems like it's going to be loads of fun .
Response: When exactly does the party start ?
Without Context: Neutral

```

With Context: Positive

--- Sample 37 ---

Context: He is surely a hard nut , nobody likes approaching him .

Response: We'd better leave him alone .

Without Context: Neutral

With Context: Negative

--- Sample 38 ---

Context: Excuse me . I wonder if you can help me .

Response: I'll try my best .

Without Context: Positive

With Context: Neutral

--- Sample 39 ---

Context: I'll try my best .

```
# Compute statistics (optional)
```

```
# count how often sentiment changes due to context
```

```
from collections import Counter
```

```
diff_count = sum([c != nc for c, nc in zip(contextual_labels, non_contextual_labels)])
```

```
print(f"\nSentiment changed in {diff_count}/100 samples ({diff_count}%) due to context.")
```



```
Sentiment changed in 24/100 samples (24%) due to context.
```

```
# Load second sentiment model (fine-tuned BERT on SST-2)
```

```
second_model = "distilbert-base-uncased-finetuned-sst-2-english"
```

```
second_classifier = pipeline("sentiment-analysis", model=second_model)
```

```
# Label mapping for SST-2
```

```
label_map_sst = {
```

```
    "NEGATIVE": "Negative",
```

```
    "POSITIVE": "Positive"
```

```
}
```

```
# Run on first 5 samples
```

```
for i in range(5):
```

```
    context = dataset["train"][i]["context"]
```

```
    response = dataset["train"][i]["response"]
```

```
    combined_input = f"{context} [SEP] {response}"
```

```
    result_response_only = second_classifier(response)[0]
```

```
    result_combined = second_classifier(combined_input)[0]
```

```
    sentiment_response_only = label_map_sst[result_response_only["label"]]
```

```
    sentiment_combined = label_map_sst[result_combined["label"]]
```

```
    print(f"Sample {i+1}")
```

```
    print(f"Context: {context}")
```

```
    print(f"Response: {response}")
```

```
    print(f"Sentiment (Response Only): {sentiment_response_only} (score: {result_response_only['score']:.2f})")
```

```
    print(f"Sentiment (With Context): {sentiment_combined} (score: {result_combined['score']:.2f})")
```

```
    print("----")
```



config.json: 100% 629/629 [00:00<00:00, 34.5kB/s]

model.safetensors: 100% 268M/268M [00:06<00:00, 53.6MB/s]

tokenizer_config.json: 100% 48.0/48.0 [00:00<00:00, 3.86kB/s]

vocab.txt: 100% 232k/232k [00:00<00:00, 12.6MB/s]

Device set to use cpu

Sample 1

Context: Waiter !

Response: I'll be with you in a second . Uh ... Yes , ma'am ?

Sentiment (Response Only): Positive (score: 1.00)

Sentiment (With Context): Positive (score: 1.00)

Sample 2

Context: I'll be with you in a second . Uh ... Yes , ma'am ?

Response: This is not what I asked for . I'm afraid .

Sentiment (Response Only): Negative (score: 1.00)

Sentiment (With Context): Negative (score: 1.00)

Sample 3

Context: This is not what I asked for . I'm afraid .

Response: Oh , I'm so sorry . May I ask what you ordered again ?

Sentiment (Response Only): Negative (score: 1.00)

Sentiment (With Context): Negative (score: 1.00)