# Sentiment_Analysis_using_Hugging_Face_Transformers

July 27, 2025

**Sentiment Analysis using Hugging Face Transformers**

**Objective:**

Build a sentiment analysis app that detects positive or negative sentiment in user-inputted text using Hugging Face Transformers, and deploy it with a stylish Gradio UI.

**Environment Setup:**

Use Google Colab with GPU runtime for faster performance.

```
Go to: Runtime > Change runtime type > GPU
```

[ ]:
```
# !pip install transformers gradio
```

**Code – Sentiment Analysis**

[1]:
```
from transformers import pipeline
import gradio as gr
```

[2]:
```
# Load the sentiment classifier
classifier = pipeline("sentiment-analysis")
```

```
No model was supplied, defaulted to distilbert/distilbert-base-uncased-
finetuned-sst-2-english and revision 714eb0f
(https://huggingface.co/distilbert/distilbert-base-uncased-finetuned-
sst-2-english).
Using a pipeline without specifying a model name and revision in production is
not recommended.
Device set to use cpu
```

[3]:
```
# Define the function
def analyze_sentiment(text):
    result = classifier(text)[0]
    return f" Sentiment: **{result['label']}**\n\n Confidence:␣
 ↪**{round(result['score'] * 100, 2)}%**"
```

[4]:
```
# Launch Gradio app with improved UI
gr.Interface(
    fn=analyze_sentiment,
    inputs=gr.Textbox(lines=3, placeholder="Type something here...",␣
 ↪label="Your Sentence"),
```

```
    outputs=gr.Markdown(),
    title="  Sentiment Analyzer",
    description="This app uses a pretrained Hugging Face Transformer to detect␣
 ↪whether a sentence is Positive or Negative.",
    theme="default"
).launch()
```

```
* Running on local URL:  http://127.0.0.1:7860
* To create a public link, set `share=True` in `launch()`.
```

```
<IPython.core.display.HTML object>
```

[4]:

[ ]: