

# Policy Summarizer - Corrected Code

```
# -----
# Install required packages
# -----
!pip install PyPDF2 transformers torch gradio

# -----
# Imports
# -----
import torch
from transformers import AutoTokenizer, AutoModelForSeq2SeqLM
import PyPDF2
import gradio as gr

# -----
# Load model & tokenizer
# -----
model_name = "facebook/bart-large-cnn" # summarization model
tokenizer = AutoTokenizer.from_pretrained(model_name)
model = AutoModelForSeq2SeqLM.from_pretrained(model_name)

# -----
# Function: Summarize Policy from PDF
# -----
def summarize_policy(pdf_file, text_input):
    text = ""

    # Extract text from uploaded PDF
    if pdf_file is not None:
        pdf_reader = PyPDF2.PdfReader(pdf_file.name)
        for page in pdf_reader.pages:
            page_text = page.extract_text()
            if page_text:
                text += page_text + "\n"

    # Add manual text input (if provided)
    if text_input:
        text += "\n" + text_input

    if not text.strip():
        return "■ No text provided to summarize."

    # Tokenize and generate summary
    inputs = tokenizer(
        text[:2000], # limit length
        return_tensors="pt",
        truncation=True,
        max_length=1024
    )
    summary_ids = model.generate(
        inputs["input_ids"],
        max_length=200,
        min_length=50,
        num_beams=4,
        early_stopping=True
    )
    summary = tokenizer.decode(summary_ids[0], skip_special_tokens=True)

    return summary

# -----
# Gradio UI
# -----
pdf_input = gr.File(label="Upload Policy PDF", file_types=[".pdf"])
text_input = gr.Textbox(label="Or Paste Policy Text")
output = gr.Textbox(label="Policy Summary")

app = gr.Interface(
    fn=summarize_policy,
    inputs=[pdf_input, text_input],
    outputs=output,
    title="Policy Summarizer"
)

# -----
```

```
# Launch app
# -----
if __name__ == "__main__":
    app.launch()
```