sentiment analysis using generative Ai

generative AI models learn patterns and relationships from large datasets and can produce novel outputs based on that learning.

In [ ]:
```
!pip install google-generativeai -q
```

In [ ]:
```
import google.generativeai as genai
```

Parameters

- Model: Which AI tool to use.
- Temperature: How creative or predictable the response should be.
- Candidate Count: How many different answers to generate.
- Top-K: Limits choices to the top 40 words.
- Top-P: Considers the most probable words until their combined chance is 95%.
- Max Output Tokens: Limits the length of the response.

In [ ]:
```
# Configure the API key
genai.configure(api_key="AIzaSyB2szmH5dgjWDNaRkMB3w5eAzdeciuo0IM")

# Default parameters for the API call
defaults = {
    'model': 'models/text-bison-001',
    'temperature': 0.5,
    'candidate_count': 1,
    'top_k': 40,
    'top_p': 0.95,
    'max_output_tokens': 1024,
}

# Prompt the user for feedback
prod_review = input("Enter your Feedback: ")

# Define the prompt with the user's feedback
prompt = f"""
What is the sentiment of the following sentence, which is delimited with tripl
Review text: ```{prod_review}```
"""

# Generate the response
response = genai.generate_text(**defaults,prompt=prompt)

# Print the result
print(response.result)
```

sentiment analysis with list

In [ ]:
```
# Configure the API key
genai.configure(api_key="AIzaSyB2szmH5dgjWDNaRkMB3w5eAzdeciuo0IM")

# Default parameters for the API call
defaults = {
    'model': 'models/text-bison-001',
    'temperature': 0.5,
    'candidate_count': 1,
    'top_k': 40,
    'top_p': 0.95,
    'max_output_tokens': 1024,
}

# List of reviews
reviews = ['I do not like this movie', 'I love this film']

# List to store the results
result = []

# Loop through each review
for review in reviews:
    # Define the prompt with the user's feedback
    prompt = f"""
    What is the sentiment of the following sentence, which is delimited with t
    Review text: ```{review}```
    """

    # Generate the response
    response = genai.generate_text(
        **defaults,
        prompt=prompt
    )

    # Append the result to the list
    result.append(response.result)

# Print the results
print(result)
```

In [ ]: