1 - Desenvolva um prompt que utilize few-shot learning para classificar comentários como “Positivos”, “Neutros” ou “Negativos”. Inclua três exemplos de cada categoria no prompt e solicite ao LLM que classifique a frase "Este episódio é divertido, mas não tão bom quanto os antigos.". Interprete o resultado.

In [5]:
```python
# Importações necessárias
import os
from dotenv import load_dotenv
import google.generativeai as genai

# Carregar e configurar a API
load_dotenv()
genai.configure(api_key=os.getenv("GOOGLE_API_KEY"))
```

Etapa 2: Definição dos Exemplos de Treinamento

Aqui definimos três exemplos para cada categoria (positivo, neutro, negativo) que serão usados no few-shot learning

In [6]:
```python
# Exemplos de treinamento organizados por categoria
training_examples = {
    "positivos": [
        "Adorei o filme, foi incrível do início ao fim!",
        "A atuação do elenco foi excepcional, superou minhas expectativas.",
        "Uma experiência maravilhosa, recomendo fortemente."
    ],
    "neutros": [
        "O filme é ok, nada especial.",
        "Tem seus momentos bons e ruins.",
        "É um filme comum, serve para passar o tempo."
    ],
    "negativos": [
        "Não gostei nada do filme, perda de tempo.",
        "A história é confusa e mal desenvolvida.",
        "Decepcionante em todos os aspectos."
    ]
}
```

Etapa 3: Implementação do Classificador

Criação das funções para gerar o prompt e classificar o sentimento

In [7]:
```python
def create_prompt(text_to_classify):
    """
    Cria o prompt com exemplos de few-shot learning e o texto a ser classificado
    """
    prompt = "Você é um classificador de sentimentos. Classifique o comentário a

    # Adicionar exemplos de treinamento
    for category in training_examples:
        prompt += f"\nExemplos de comentários {category.title()}:\n"
        for example in training_examples[category]:
            prompt += f"Texto: '{example}'\nClassificação: {category.title()}\n"
```

```
    # Adicionar texto a ser classificado
    prompt += f"\nClassifique o seguinte texto:\n'{text_to_classify}'\nClassific

    return prompt

def classify_sentiment(text):
    """
    Classifica o sentimento do texto usando o modelo do Google
    """
    # Criar o prompt com os exemplos e o texto
    prompt = create_prompt(text)

    # Configurar o modelo
    model = genai.GenerativeModel('gemini-pro')

    # Fazer a classificação
    response = model.generate_content(prompt)

    return response.text.strip()
```

In [8]:
```
# Texto de exemplo para classificação
texto_teste = "Este episódio é divertido, mas não tão bom quanto os antigos."

print("Texto para classificação:", texto_teste)
print("\nClassificando...")

# Realizar a classificação
resultado = classify_sentiment(texto_teste)
print("\nResultado da classificação:", resultado)
```

```
Texto para classificação: Este episódio é divertido, mas não tão bom quanto os an
tigos.

Classificando...

Resultado da classificação: Neutro
```

In [13]:
```
# Exemplo que tende ao positivo
texto_teste_2 = "A nova atualização do sistema trouxe melhorias significativas n

print("Texto para classificação:", texto_teste_2)
print("\nClassificando...")
resultado_2 = classify_sentiment(texto_teste_2)
print("\nResultado da classificação:", resultado_2)

print("\n" + "="*50 + "\n")
```

```
Texto para classificação: A nova atualização do sistema trouxe melhorias signific
ativas no desempenho e corrigiu todos os bugs anteriores!

Classificando...

Resultado da classificação: Positivo

==================================================
```

In [14]:
```
# Exemplo que tende ao negativo
texto_teste_3 = "O sistema está muito lento após a atualização e vários recursos
```

```
print("Texto para classificação:", texto_teste_3)
print("\nClassificando...")
resultado_3 = classify_sentiment(texto_teste_3)
print("\nResultado da classificação:", resultado_3)
```

```
Texto para classificação: O sistema está muito lento após a atualização e vários recursos importantes pararam de funcionar completamente.

Classificando...

Resultado da classificação: Negativo
```