Não coloquei método de pagamento na OpanAI

In [ ]:
```
from skllm.config import SKLLMConfig
from skllm.datasets import get_classification_dataset
import utils_config
config_file="./config.json"
config = utils_config.load_config(config_file)
```

In [ ]:
```
SKLLMConfig.set_openai_key(config.OPENAI_SECRET_KEY)
SKLLMConfig.set_openai_org(config.OPENAI_ORG_ID)
```

Classificador Simples

In [ ]:
```
movie_reviews = [
    "This movie was absolutely wonderful. The storyline was compelling and the c
    "I really loved the film! The plot had a few unexpected twists which kept me
    "The movie was alright. Not great, but not bad either. A decent one-time wat
    "I didn't enjoy the film that much. The plot was quite predictable and the c
    "This movie was not to my taste. It felt too slow and the storyline wasn't e
    "The film was okay. It was neither impressive nor disappointing. It was just
    "I was blown away by the movie! The cinematography was excellent and the per
    "I didn't like the movie at all. The story was uninteresting and the acting
    "The movie was decent. It had its moments but was not consistently engaging.
]

movie_review_labels = [
    "positive",
    "positive",
    "neutral",
    "negative",
    "negative",
    "neutral",
    "positive",
    "negative",
    "neutral"
]

new_movie_reviews = [
    "The movie was fantastic! I was captivated by the storyline from beginning t
    "I found the film to be quite boring. The plot moved too slowly and the acti
    "The movie was okay. Not the best I've seen, but certainly not the worst."
]
```

In [ ]:
```
from skllm import ZeroShotGPTClassifier

model = ZeroShotGPTClassifier()
model.fit(X=movie_reviews, y=movie_review_labels)
movie_labels = model.predict(X=new_movie_reviews)
```

```
100%|██████████| 3/3 [00:02<00:00,  1.08it/s]
```

In [ ]:
```
for idx, review in enumerate(new_movie_reviews):
    print(f"({movie_labels[idx]}) => { review} ")
```

```
(positive) => The movie was fantastic! I was captivated by the storyline from beg
inning to end.
(negative) => I found the film to be quite boring. The plot moved too slowly and
the acting was subpar.
(neutral) => The movie was okay. Not the best I've seen, but certainly not the wo
rst.
```

Classificador Multi-Label

In [ ]:
```
restaurant_reviews = [
    "The food was delicious and the service was excellent. A wonderful dining ex
    "The restaurant was in a great location, but the food was just average.",
    "The service was very slow and the food was cold when it arrived. Not a good
    "The restaurant has a beautiful ambiance, and the food was superb.",
    "The food was great, but I found it to be a bit overpriced.",
    "The restaurant was conveniently located, but the service was poor.",
    "The food was not as expected, but the restaurant ambiance was really nice."
    "Great food and quick service. The location was also very convenient.",
    "The prices were a bit high, but the food quality and the service were excel
    "The restaurant offered a wide variety of dishes. The service was also very
]

restaurant_review_labels = [
    ["Food", "Service"],
    ["Location", "Food"],
    ["Service", "Food"],
    ["Atmosphere", "Food"],
    ["Food", "Price"],
    ["Location", "Service"],
    ["Food", "Atmosphere"],
    ["Food", "Service", "Location"],
    ["Price", "Food", "Service"],
    ["Food Variety", "Service"]
]

new_restaurant_reviews = [
    "The food was excellent and the restaurant was located in the heart of the c
    "The service was slow and the food was not worth the price.",
    "The restaurant had a wonderful ambiance, but the variety of dishes was limi
]
```

In [ ]:
```
from skllm import MultiLabelZeroShotGPTClassifier

model = MultiLabelZeroShotGPTClassifier()
model.fit(X=restaurant_reviews, y=restaurant_review_labels)
restaurant_predict_labels = model.predict(X=new_restaurant_reviews)
```

```
100%|██████████| 3/3 [00:02<00:00,  1.22it/s]
```

In [ ]:
```
for idx, review in enumerate(restaurant_predict_labels):
    print(f"({restaurant_predict_labels[idx]}) => { review} ")
```

```
(['Food', 'Location']) => ['Food', 'Location']
(['Service', 'Price']) => ['Service', 'Price']
(['Atmosphere', 'Food Variety']) => ['Atmosphere', 'Food Variety']
```

Get Classification Dataset

In [ ]:
```
X, _ = get_classification_dataset()

model_classification = ZeroShotGPTClassifier()
model_classification.fit(None, ['positive', 'negative', 'neutral'])
labels = model_classification.predict(X)
```

```
100%|██████████| 30/30 [00:22<00:00,  1.36it/s]
['positive', 'positive', 'positive', 'positive', 'positive', 'positive', 'positiv
e', 'positive', 'positive', 'positive', 'negative', 'negative', 'negative', 'nega
tive', 'negative', 'negative', 'negative', 'negative', 'negative', 'negative', 'n
egative', 'neutral', 'neutral', 'neutral', 'neutral', 'negative', 'negative', 'ne
utral', 'neutral', 'neutral']
```

In [ ]:
```
labels
```

Out[ ]:
```
['positive',
 'positive',
 'positive',
 'positive',
 'positive',
 'positive',
 'positive',
 'positive',
 'positive',
 'positive',
 'negative',
 'negative',
 'negative',
 'negative',
 'negative',
 'negative',
 'negative',
 'negative',
 'negative',
 'negative',
 'negative',
 'negative',
 'neutral',
 'neutral',
 'neutral',
 'neutral',
 'negative',
 'negative',
 'neutral',
 'neutral',
 'neutral']
```